Development of best response policies in multi-state attacks with Monte Carlo simulations

Computational Decision Making & Risk Analysis – CS 5376

Jonathan Avila & Oscar Galindo

Final Project

12/05/2020

**Introduction**

For this work we propose to perform a simulation of the possible scenarios that an organization may face under attack after discovering a zero-day vulnerability. To do this, first we propose a model, or a structured series of decisions, that an organization must follow depending on the possible outcomes and the outcomes’ probabilities as the attacker proceeds to make decisions. Then, we propose the use of Monte Carlo simulations to determine the best response policy for the organization to take at any step of the decision-making process. Finally, to make the decision-making process more interesting we present different scenarios where the outcomes vary depending on the collaboration degree of both parties (e.g., If the attacker attacks the organization with the intention of negotiating and the organization wants to negotiate as well).

**Scenario**

For this work we take an organization that has a private network that allows the company to keep secure the contents of their business. The organization has put into place a system to help with defend their network, but at the time an attack occurs we assume the security system has been overridden. Furthermore, this is a company with a lot of employees, and we assume that the vulnerability was obtained by a user unsuspectedly allowing a vulnerability to be introduced to the network. This vulnerability only affects the main hardware of the organization and becomes a ransomware that takes control over access to the files in the servers. The kind of organization we talk about is a medium size organization probably with a revenue of less than a million dollars (or around 1,000 units of value in our context) and we assume it to be big enough to require a significant level work when an issue happens with the infrastructure, hence the costs of replacing hardware, launching an investigation, etc. Which will be used throughout this work.

**Model**

For this exercise we make a couple of assumptions. We assume that the attacker has nothing to lose and everything to win by asking for ransoms. We also assume that the organization has found the vulnerability without knowing if an attack happened. We also model our decision-making scenarios in such a way that the next state is determined by the decision of the next player. For example, at the beginning of our scenario (i.e., at the top of the tree) an organization finds a vulnerability. If the attacker decides to perform an attack, then we immediately move to the left of the tree, otherwise we assume that there is time to make a different set of decisions, so we consider the options on the right side of the tree.

We further increase the realism of the simulation by including some cost for letting time pass by, since the organization is losing access to data and cannot do business while their servers are hijacked. Furthermore, if the organization decides to perform an investigation, the organization can decide to wait and observe if the “issue” is resolved before reacting to it. Also, we include the costs of killing the service or basically stopping the affected hardware from running, as well as the differences of paying a negotiated ransom from the perspective of being the party that asks for the negotiation or being the party that receives the negotiation offer.

A picture containing diagram

Description automatically generatedChart, radar chart

Description automatically generated

**Simulation Values**

To decide on the hypothetical values for the simulation we start by setting the value of the company. From the company’s value we derive fractional costs of operation, or how much money is obtained by unit of time operated to assess the cost of losing the ability of operation, which we concluded was approximately 0.1% per unit of time. We assess the cost of launching an investigation to be around 1% of the company’s value since the entire network architecture would have to be analyzed.

We also assess the cost of getting all new hardware – which normally is prohibited in our view – and so we conclude the cost of performing this directive to be 5% of the company’s value. We set the cost of paying the ransom asked by the attacker to be between 1-5% of the company’s value since this value is “small” enough that most companies would be willing to pay it. Also, we assess the cost of preparing the company for an attack to be based on the cost of the obtainable ransom. For example, if the attacker attempts to obtain up to 5% of the company value as a ransom the cost of preparing should be at least less than what the cost of paying the full ransom is. So, we set the value of this situation to be around 1-2% of the company’s value. On the other hand, if the company decides to not prepare this decision does not mean to just ignore a vulnerability, instead it means that the organization decides to “cover” itself by paying a premium to obtain an insurance, we assess the cost of this premium to be around 2.5% of the company’s value.

In our considerations we take into the account the “emotional distress” that the whole company’s team suffers when the attack is known, as such we provide a value of 0.5% when there is an attack. Furthermore, we take into account the cost of getting attacked after an investigation to be almost negligeable but not quite, and so we decide to give a cost for that situation of around 0.1% of the company’s value. In the case when we the company is attacked by surprise, we model the further distress that is caused by the company choosing to “not pay” with of around 0.3% of the company’s value, when the attacked is not a surprise because the investigation was launched the emotional distress gets reduced since the company is probably already having a negative outlook of the outcome of the investigation. In general, whenever we present a cost of paying the ransom, we literally show either 25%, 50%, and 75% depending on the collaboration degree of the attacker and the defender. For example, after an investigation is launched, and the attack was beginning by the attacker the defender could choose to wait, and the attacker might decide to offer a “cheap” exit by lowering the ransom to 25% of the original cost. In contrast, in situations where the defender decides not to agree to pay and the attacker has a negative response to that determination, the attacker could decide to expose sensitive data, at which point neither of the two players has a high degree of willingness to collaborate but recognize the value in negotiating so the attacker might decide to lower the ransom to 50% or 75% of the original ransom.

In some general situations the outcome might be to pay the full amount like when there is an absolute lack of willingness to continue “playing” the game, in such cases due to the attacker being fearful/uncertain about the future the defender might decide to simply pay the full amount, like when the attack occurs, and the defender simply pays off right away. Lastly, when the investigation is ongoing and the organization decides not to take any action but instead “wait” the organization has a slim, but true chance of getting to solve the issue and getting the attacker off the network. In that case we assume that the cost of doing so is not existent an action only has to be executed.

**Diagram with Simulation Values**

* **Comment: Value of the company set to 1000, and the loss/moment of time = 1, ransom set to 0.3%:**

A picture containing diagram

Description automatically generatedChart, radar chart

Description automatically generated

**Experimental Results**

**Comment:** We wrote the code for the simulations below (attached as *simulate.py*). Following the magnitudes of the payoffs, we found the learning rate of 0.1 to be appropriate. A learning rate of 0.1 places less weight on new experiences and more weight on previous experience during the learning process.

* **Running for 10,000 episodes**

Baseline expected payoff = -37.02235

N\_EPISODES=10000, LEARNING\_RATE=0.1, START\_STATE="d0: vulnerability found", STEP\_COST=-1

[state\_id: state\_name] = value (terminal states indicated with >)

TERMINAL STATES

[d39: pay 75%] = -44.543

[d38: pay 50%] = -38.651

[d34: pay full amount] = -56.637

[d33: kill service] = -46.668

[d32: pay 75%] = -39.328

[d31: pay 50%] = -28.897

[d30: pay 50%] = -39.996

[d29: pay 25%] = -32.495

[d28: pay 75%] = -69.5

[d27: pay 50%] = -62.0

[d23: lost control] = -21.873

[d20: prepare] = -29.0

[d19: not prepare] = -39.0

[d17: pay full amount] = -45.0

[d14: pay full amount] = -75.0

[d13: kill service] = -65.0

[d12: pay 75%] = -61.5

[d11: pay 50%] = -54.0

[d5: replace hardware] = -52.0

[d4: pay full amount] = -37.0

\*\*\*\*\* NON-TERMINAL STATES

[d37: ask 75%] = -44.543

[d36: ask 50%] = -38.651

[d35: negotiate] = -47.078

[d26: expose sensitive data not expecting negotiation] = -51.896

[d25: negotiate] = -37.433

[d24: negotiate] = -36.062

[d22: ask 75%] = -69.5

[d21: ask 50%] = -62.0

[d18: disagree] = -48.683

[d16: wait] = -35.816

[d15: negotiate] = -67.99

[d10: not attacked] = -34.489

[d9: attacked] = -42.751

[d8: expose sensitive data expecting negotiation] = -69.772

[d7: negotiate] = -57.368

[d6: launch investigation] = -38.876

[d3: not pay] = -67.507

[d2: attack did not happen] = -44.379

[d1: attack happened] = -54.542

[d0: vulnerability found] = -54.08

* **Running 100,000 episodes**

Baseline expected payoff = -36.95546

N\_EPISODES=100000, LEARNING\_RATE=0.1, START\_STATE="d0: vulnerability found", STEP\_COST=-1

[state\_id: state\_name] = value (terminal states indicated with >)

TERMINAL STATES

[d39: pay 75%] = -51.5

[d38: pay 50%] = -44.0

[d34: pay full amount] = -57.0

[d33: kill service] = -47.0

[d32: pay 75%] = -41.5

[d31: pay 50%] = -34.0

[d30: pay 50%] = -40.0

[d29: pay 25%] = -32.5

[d28: pay 75%] = -69.5

[d27: pay 50%] = -62.0

[d23: lost control] = -24.0

[d20: prepare] = -29.0

[d19: not prepare] = -39.0

[d17: pay full amount] = -45.0

[d14: pay full amount] = -75.0

[d13: kill service] = -65.0

[d12: pay 75%] = -61.5

[d11: pay 50%] = -54.0

[d5: replace hardware] = -52.0

[d4: pay full amount] = -37.0

\*\*\*\*\* NOT TERMINAL STATES

[d37: ask 75%] = -51.5

[d36: ask 50%] = -44.0

[d35: negotiate] = -47.819

[d26: expose sensitive data not expecting negotiation] = -51.385

[d25: negotiate] = -38.189

[d24: negotiate] = -36.221

[d22: ask 75%] = -69.5

[d21: ask 50%] = -62.0

[d18: disagree] = -48.794

[d16: wait] = -34.681

[d15: negotiate] = -69.259

[d10: not attacked] = -35.397

[d9: attacked] = -43.656

[d8: expose sensitive data expecting negotiation] = -67.941

[d7: negotiate] = -57.03

[d6: launch investigation] = -41.954

[d3: not pay] = -63.996

[d2: attack did not happen] = -49.256

[d1: attack happened] = -54.909

[d0: vulnerability found] = -54.351

* **Running 1,000,000**

Baseline expected payoff = -36.950895

N\_EPISODES=1000000, LEARNING\_RATE=0.1, START\_STATE="d0: vulnerability found", STEP\_COST=-1

[state\_id: state\_name] = value (terminal states indicated with >)

TERMINAL STATES

[d39: pay 75%] = -51.5

[d38: pay 50%] = -44.0

[d34: pay full amount] = -57.0

[d33: kill service] = -47.0

[d32: pay 75%] = -41.5

[d31: pay 50%] = -34.0

[d30: pay 50%] = -40.0

[d29: pay 25%] = -32.5

[d28: pay 75%] = -69.5

[d27: pay 50%] = -62.0

[d23: lost control] = -24.0

[d20: prepare] = -29.0

[d19: not prepare] = -39.0

[d17: pay full amount] = -45.0

[d14: pay full amount] = -75.0

[d13: kill service] = -65.0

[d12: pay 75%] = -61.5

[d11: pay 50%] = -54.0

[d5: replace hardware] = -52.0

[d4: pay full amount] = -37.0

\*\*\*\* NON-TERMINAL STATES

[d37: ask 75%] = -51.5

[d36: ask 50%] = -44.0

[d35: negotiate] = -46.643

[d26: expose sensitive data not expecting negotiation] = -49.886

[d25: negotiate] = -37.28

[d24: negotiate] = -36.997

[d22: ask 75%] = -69.5

[d21: ask 50%] = -62.0

[d18: disagree] = -45.554

[d16: wait] = -33.513

[d15: negotiate] = -67.669

[d10: not attacked] = -32.892

[d9: attacked] = -40.692

[d8: expose sensitive data expecting negotiation] = -67.937

[d7: negotiate] = -57.352

[d6: launch investigation] = -37.339

[d3: not pay] = -62.387

[d2: attack did not happen] = -41.89

[d1: attack happened] = -42.805

[d0: vulnerability found] = -43.447

* **Running 10,000,000**

Baseline expected payoff = -36.9503154

N\_EPISODES=10000000, LEARNING\_RATE=0.1, START\_STATE="d0: vulnerability found", STEP\_COST=-1

[state\_id: state\_name] = value (terminal states indicated with >)

TERMINAL STATES

[d39: pay 75%] = -51.5

[d38: pay 50%] = -44.0

[d34: pay full amount] = -57.0

[d33: kill service] = -47.0

[d32: pay 75%] = -41.5

[d31: pay 50%] = -34.0

[d30: pay 50%] = -40.0

[d29: pay 25%] = -32.5

[d28: pay 75%] = -69.5

[d27: pay 50%] = -62.0

[d23: lost control] = -24.0

[d20: prepare] = -29.0

[d19: not prepare] = -39.0

[d17: pay full amount] = -45.0

[d14: pay full amount] = -75.0

[d13: kill service] = -65.0

[d12: pay 75%] = -61.5

[d11: pay 50%] = -54.0

[d5: replace hardware] = -52.0

[d4: pay full amount] = -37.0

\*\*\*\* NON-TERMINAL STATES

[d37: ask 75%] = -51.5

[d36: ask 50%] = -44.0

[d35: negotiate] = -46.384

[d26: expose sensitive data not expecting negotiation] = -49.984

[d25: negotiate] = -37.516

[d24: negotiate] = -34.47

[d22: ask 75%] = -69.5

[d18: disagree] = -43.465

[d16: wait] = -33.215

[d15: negotiate] = -68.634

[d10: not attacked] = -33.463

[d9: attacked] = -39.931

[d8: expose sensitive data expecting negotiation] = -68.668

[d7: negotiate] = -58.232

[d6: launch investigation] = -36.055

[d3: not pay] = -67.448

[d2: attack did not happen] = -44.641

[d1: attack happened] = -54.232

[d0: vulnerability found] = -52.711

* **Running 100,000,000**

Baseline expected payoff = -36.9505791

N\_EPISODES=100000000, LEARNING\_RATE=0.1, START\_STATE="d0: vulnerability found", STEP\_COST=-1

[state\_id: state\_name] = value (terminal states indicated with >)

TERMINAL STATES

[d39: pay 75%] = -51.5

[d38: pay 50%] = -44.0

[d34: pay full amount] = -57.0

[d33: kill service] = -47.0

[d32: pay 75%] = -41.5

[d31: pay 50%] = -34.0

[d30: pay 50%] = -40.0

[d29: pay 25%] = -32.5

[d28: pay 75%] = -69.5

[d27: pay 50%] = -62.0

[d23: lost control] = -24.0

[d20: prepare] = -29.0

[d19: not prepare] = -39.0

[d17: pay full amount] = -45.0

[d14: pay full amount] = -75.0

[d13: kill service] = -65.0

[d12: pay 75%] = -61.5

[d11: pay 50%] = -54.0

[d5: replace hardware] = -52.0

[d4: pay full amount] = -37.0

\*\*\*\* NON-TERMINAL STATES

[d37: ask 75%] = -51.5

[d36: ask 50%] = -44.0

[d35: negotiate] = -48.057

[d26: expose sensitive data not expecting negotiation] = -50.614

[d25: negotiate] = -36.962

[d24: negotiate] = -34.604

[d22: ask 75%] = -69.5

[d21: ask 50%] = -62.0

[d18: disagree] = -47.683

[d16: wait] = -31.873

[d15: negotiate] = -67.9

[d10: not attacked] = -34.361

[d9: attacked] = -37.073

[d8: expose sensitive data expecting negotiation] = -67.149

[d7: negotiate] = -56.934

[d6: launch investigation] = -34.829

[d3: not pay] = -65.215

[d2: attack did not happen] = -42.915

[d1: attack happened] = -51.177

[d0: vulnerability found] = -50.374

**Baseline**

The baseline is a greedy algorithm; the attacker chooses actions that follow our estimated probability distributions while the defender chooses actions that results in the largest immediate payoff. Since actions have an associated cost (sometimes zero cost), the defender is always choosing the “least bad” option. The defender follows this algorithm for the same number of episodes as our Monte Carlo simulations and we record the average payoff

**Discussion**

In our simulations we observe that the baseline predicts that going down the “least bad” path might be just as expensive as paying the entire ransom (-36.95 units compared to -30 units). Knowing this, we might be led to believe it is worth the chance of making the attacker lose control through an investigation or to pay only a portion of the ransom via negotiation. Our experiments, after 1,000,000 simulations, contradict this belief. In fact, we can expect a cost of -43 units from the very beginning of the tree, contradicting the optimistic view of the baseline. Furthermore, there are states like d27, d28 which allow to pay a reduced ransom but in reality, it can be expected for those states to cost as much as “killing the service/containing the threat to solve it” (d13) which might be a “safer” solution because it is unknown if the attacker will comply with its word.

While it seems interesting that the baseline predicts that the average payoff by traveling the tree should cost a little more than just paying the full ransom, it also seems unrealistic. Since, for example, we do not account for how different the likeliness of every event is – maybe bad scenarios are most likely to happen than good scenarios – and we also can see through our simulations a disparity of around 22% more cost based on the simulations than in the simple baseline analysis for 1 million simulations, which is also the case were the least disparity exists. With the information provided to the organization through Monte Carlo simulations, the organization/company might decide to pay right away the ransom if the opportunity arises or maybe take a greater cost to simply replace the hardware – since a cost of 45 is close to a cost of 52 to replace the hardware. Or, as the results after simulating 100 million episodes suggest, it might be desirable to change the hardware instead of taking any other risk by launching an investigation.

Based on our results we can confirm that the organization/company is statistically better off in cases that seem at a glance really bad in terms of payoff. This means, that our analysis provides the organization will valuable results that indicate that some not “common sense” options are indeed valuable if taken. Hence, our results offer great insight for the organization.

Table

Description automatically generated



Chart, line chart

Description automatically generated



**Time Complexity Analysis**

The baseline algorithm presented in this work makes quick decisions for the defender. However, like the Monte Carlo simulations, the baseline’s runtime depends on the number of episodes (a linear relationship). The algorithm needs to determine the attacker’s choice at each of his nodes determined by a probability distribution. For larger number of episodes, simulations required several minutes to complete on modest hardware. The code may benefit from a stop condition instead of relying on a set number of episodes; additional episodes are unnecessary once we have found that the value converges. Both algorithms have a similar runtime, but the baseline algorithm will give a worse approximation compared to the Monte Carlo simulations.

**Conclusions**

We showed that there is value in running these simulations and that following a naïve algorithm like our baseline is not always the best solution when faced with uncertainty. Modeling our problem required estimating costs and probabilities following our assumptions. We cannot model the problem perfectly and there is still uncertainty, albeit a lot less uncertainty. These simulations do not consider real-world “secret” reactions like stealing data for further blackmailing or other unique scenarios, but overall, the approximations simile the reality of how an exchange between attacker and defender will go down like. In a future work more scenarios with repeating patterns, like not negotiating for a little amount of time and then actually negotiating, could be modeled which may add complexity but would make the conclusions even more realistic. In addition, we think that there are some repetitive processes that could be modeled in the system. For example, when the user is negotiating with the attacker there could be an inclusion of more options to respond like attacking again/exposing data, etc. By adding this non-cooperative actions we could add even more realism to our work and hence more value for the organization in our analysis.