CS765 Autumn 2021 Semester, Project Part-2

Simulating a selfish mining attack using the P2P Cryptocurrency Network developed in the last assignment

Due Date: 23:59 hrs, October 12, 2021

Note: Marks will be awarded for the points mentioned within the text. The marks have been indicated in square brackets (e.g. [1]).

In this assignment you will have to simulate different attacks on top the *discrete-event* simulator for a P2P cryptocurrency network, you built in the last assignment. This assignment can be done in groups consisting of **at most 3** persons.

You can use any programming language of your choice. If you include code from a publicly available source, then state the source in the comments of the code. **Not more than 15% of the code should be taken from such sources.**

Selfish Mining Attack This is the same attack which was proposed by Eyal and Sirer in the paper "Majority is not Enough" (4th link mentioned in "Useful links"). The key idea behind this strategy is for a selfish miner (Adversary) to keep its discovered blocks private, thereby intentionally forking the chain. The honest nodes continue to mine on the public chain, while the adversary mines on its own private branch. If the adversary mines more blocks, it develops a longer lead over the public chain and continues to keep these new blocks private. When the public branch approaches the adversary's private branch in length, it reveals blocks from its private chain to the public. Steps that the adversary follows are given below. These are slightly different to those discussed in class, but follow the paper by Eyal more closely.

1. If the lead of the selfish miner is 1 block and the honest miners discover a block, then the selfish miner immediately broadcasts the block he has mined secretly. A competition then follows, where $0 \leq \gamma \leq 1$ fraction of honest nodes mines on the selfish miners block while $1 - \gamma$ fraction of honest miners mines on the honest block. At the same time, the selfish miner also mines on top of his own block. **note that γ is not a parameter here**. All the honest miner mines on top of the block (honest or adversary block) they first receive. We mentioned the γ just to explain how the attack works.

2. If the lead of the selfish miner is 2 blocks and the honest miners discover a block, then the selfish miner broadcasts immediately all the blocks he has mined secretly.

3. If the advance of the selfish miner is greater than 2, as soon as the honest miners discovers one block, then the selfish miner makes public one more block, releasing a subchain that ends with that block which enters into competition with the new honest block. The selfish miner keeps mining on top of his secret chain.

4. Except in (1), the selfish miner keeps on mining secretly on top of his fork.

Note that the attacker does not forward block generated by other nodes. You can find a detailed explanation in the paper "Majority is not Enough" by Gun Sirer et al (link given below in the "Useful links").
Marks:
Correct implementation of each step listed above. [15, 7, 8]

Stubborn Mining Attack: In selfish mining, when $lead = 2$ and if the honest node finds the next block and closes the gap by 1, the selfish miner would immediately reveal her private chain to guarantee that the network chooses her private chain over the honest (public) chain. Therefore, the state transitions to lead = 0. In stubborn mining, instead of revealing her entire private chain, the adversary reveals the next block on her private chain only to match the length of the public chain. In this case, γ fraction of honest node will mine on adversary private chain, and $1 - \gamma$ fraction mines on honest (public) fork; and the state transitions to $lead = 1$. This has pros and cons for the attacker: if the $(1 - \gamma)$ fraction of honest node succeeds in advancing public chain, an adversary may risk losing her private chain. However, if the adversary or the γ fraction of honest node advances adversary's private fork, then adversary has successfully diverted a part of honest, $(1 - \gamma)$ fraction to

do useless work. (Note that parameter γ is used in the paper for analysis purposes. In your simulator, what fraction of honest nodes mine on the attacker's block will depend on the network topology and which nodes are fast or slow.)

When adversary wins a race in state $lead = 0'$, the selfish-miner would hurry to reveal her new block to the public, transitioning to $lead = 0$. By contrast, the stubborn miner would conceal her new block and continue mining on it privately, thus proceeding to state $lead = 1$. For more detail on Stubborn mining, refer to the useful links (at end of the document).

Marks:
Correct implementation of all the functionality mentioned above. [30]

Simulate latencies L_{ij} between pairs of peers i and j connected by a link. Latency is the time between which a message m was transmitted from sender i and received by another node j. Choose the latency to be of the form $\rho_{ij} + |m|/c_{ij} + d_{ij}$, where ρ_{ij} is a positive minimum value corresponding to speed of light propagation delay, $|m|$ denotes the length of the message in bits, c_{ij} is the link speed between i and j in bits per second, and d_{ij} is the queuing delay at node i to forward the message to node j. d_{ij} is randomly chosen from an exponential distribution with some mean $96kbits/c_{i,j}$. Note that $d_{i,j}$ must be randomly chosen for *each message* transmitted from i to j. ρ_{ij} can be chosen from a uniform distribution between 10ms and 500ms at the start of the simulation. c_{ij} is set to 100 Mbps if both i and j are fast, and 5 Mbps if either of the nodes is slow. **Note that 50% of the honest nodes in the network are slow while an adversary will always be fast. Also, you have to vary the fraction of honest nodes ζ, an adversary is connected to**.

Experiment with choosing different values for different parameters (n, T_{tx}, mean of T_k, adversary mining power, ζ). **For simplicity you can consider** $n = 100$ **and** $T_{tx} = 10$ **ms,** $\zeta = 25\%, 50\%, 75\%$. At the demo time, you are expected to run the experiment for a specific parameter configuration. By running the experiment, find the following ratios(1, 2) for both the aforementioned mining strategies and compare them. Furthermore, find the effect of adversary mining power on these ratios. Also, you should compare the fraction of attacker blocks in the main chain obtained experimentally with the theoretically obtained one in Eyal and Sirer paper.

$$MPU_{node_{adv}} = \frac{\text{Number of block mined by an adversary in main chain}}{\text{Total number of blocks mined by an adversary}} \tag{1}$$

$$MPU_{node_{overall}} = \frac{\text{Number of block in the main chain}}{\text{Total number of blocks generated across all the nodes}} \tag{2}$$

Use an appropriate visualization tool to study the blockchain tree (suitable choices can be gnuplot, matlab, or other visualization tools). We expect you to show the blockchain tree at an adversary and one honest node using a visualization tool. You do not need to include the blockchain tree in the report, but TA will ask you to show the tree at the demo time while running the experiment.

Marks:
Proper study using different parameters and use of a particular visualization tool to show the blockchain tree. [12]
Insight and critique of the observed values. [8]

In your submission on Moodle, submit a single zip file (filename format: RollNo1_RollNo2_RollNo3.zip) containing:
1. Source code for simulator. You need not submit any code for the visualization tool.
2. README file with instructions for compiling and running.
4. A report detailing your findings along with pictures of typical blockchain trees and appropriate insight.

Marks: Proper commenting of code [6], Design document [8], README file [6]

Useful links:
`https://people.orie.cornell.edu/mru8/orie3120/lec/lec10.pdf`
`http://cs.baylor.edu/~maurer/aida/desauto/chapter3.pdf`
`https://www.cs.cmu.edu/~music/cmsip/readings/intro-discrete-event-sim.html`
Majority is not enough - `https://arxiv.org/abs/1311.0243`
Stubborn Mining - `https://ieeexplore.ieee.org/abstract/document/7467362`