Analysis and Attacks of decentralized content curation platforms

AndrÃĺs Monteoliva Mosteiro and Orfeas Stefanos Thyfronitis Litos

University of Edinburgh
o.thyfronitis@ed.ac.uk, a.monteoliva@serious.server

Abstract. We will attack Steem.

1 Introduction

Steem is not incentive-compatible.

2 Related Work

Many people have done many similar things.

3 Model

1 Properties

Players The are $N \in \mathbb{N}^*$ players in the model. A player u is defined by it Steem Power SP, its Voting Power VP, its Likability Distribution L and its Strategy S. Let the ith player be characterized with the following tuple $u_i = (SP_i, VP_i, L_i, S_i)$ where $SP \in \mathbb{N}$, $VP \in [0, 100]$, $L \in \mathcal{D}\left([0,1]^N\right)$ and $S \in \{H, G\} \times [100] \times 2^{[N]}$. Lets describe each of the properties of a player in more detail:

- The Steem Power funds of player U_i are defined as $sp_i \in \mathbb{N}$. The vector of Steem Power funds for the N players is defined as $\mathcal{SP} = (sp_1, ..., sp_i, ...sp_n)\ \forall i \in N$.
- The Voting Power...
- A player u_i will have a "Likability Distribution" L_i which determines how likely is the jth to like a post created by u_i , defined as $L_i \in \mathcal{D}\left([0,1]^N\right)$. The Likability Distribution for all the players will be $\mathcal{L} = (L_1, ..., L_i, ..., L_n)\ \forall i \in N$

– The strategy of player u_i is defined as $s_i \in (\{H, G\}, Min, R)$, where $H \equiv honest$ and $G \equiv greedy$, $Min \in [0, 1]$ is the Minimum Voting Power the player can reach and $r_i \in \mathbb{N}$ is the size of u_i Voting Ring. If player u_i is $Honest$, her Voting Ring $r_i = \emptyset$. If player u_j is $Greedy$, her Voting Ring $r_j = p_1, ..., p_k, ..., p_r$ where $p_k \in [1, N]$, .The vector of the strategies for the N players is defined as $\mathcal{S} = (s_1, ..., s_i, ...s_n)$ $\forall i \in N$

The set of players is defined as $\mathcal{U} = (u_1, ..., u_i, ..., u_n)$ $\forall i \in [N]$.

Posts

– Each player can create one post. The post created by U_i is defined as p_i. The set of all posts will be then $\mathcal{P} = \bigcup_{i=1}^{N} p_i$.
– The likability of a post will be defined as $l_i \in [0, 1]^N$, where $l_i \sim L_i$ is retrieved at random following the "Likability Distribution" of player U_i.

2 Game Execution

Algorithm 1 Each player creates a post according to Likability Distribution

```
function PostGeneration(N, L)
    P = ∅                                          ▷ List of posts
    for i ∈ N do
        l_i ←R L_)                                 ▷ Get likability of posts
        p ← (i, l_i)
        P ← P‖p                                    ▷ Add post to list of Posts
    end for
    P ← Shuffle(P)                                 ▷ Shuffle the order of Posts
    return P
end function
```

Algorithm 2 Player casts votes according to her strategy and until she reaches her Min Voting Power

```
1: function VOTE(player, 𝒫, s, sp)
2:     switch s do
3:         case Honest
4:             for p ∈ 𝒫 do                                  ▷ Iterate over all the posts
5:                                 ▷ If user likes the post and has not reached min VPower
6:                 if l_p > 0 ∧ p.VPower > s.Min then
7:                     voteValue ← p.VPower · l_p · sp
8:                     p ← p.votes + voteValue
9:                 end if
10:            end for
11:        end case
12:        case Greedy
13:            for p ∈ 𝒫 do                                  ▷ Iterate over all the posts
14:                            ▷ If post belongs to voting ring and not reached min VPower
15:                if p ∈ s.R ∧ p.VPower > s.Min then
16:                    voteValue ← p.VPower · weight · sp
17:                    p ← p.votes + voteValue
18:                end if
19:            end for
20:        end case
21:    end switch
22:    𝒫 ← ORDER(𝒫.votes)                  ▷ Order posts by value of votes received
23:    return 𝒫
24: end function
```

Algorithm 3 Players cast votes over the maxRounds

```
1: function CURATION(N, 𝒮, 𝒫, 𝒮𝒫, maxR)
2:     round = 0
3:     while round < maxR do                          ▷ Vote while there are rounds left
4:         𝒩 ← SHUFFLE(𝒩)                  ▷ Randomize order of player participation
5:         for i ∈ 𝒩 do
6:             𝒫 ← VOTE(i, 𝒫, s, sp)                          ▷ Player i votes for posts
7:         end for
8:         round ← round + 1
9:     end while
10:    return 𝒫
11: end function
```

Algorithm 4 Main protocol of curation of posts in Steem. It represents one week of voting and each player creates only one post. At the end of the week, the list of ordered posts by value of votes received will be returned.

function PROTOCOL($N, \mathcal{S}, \mathcal{L}, \mathcal{SP}, maxR$)
    $\mathcal{P} \leftarrow$ POSTGENERATION($N, \mathcal{L}$)
    $\mathcal{P} \leftarrow$ CURATION($N, \mathcal{S}, \mathcal{P}, \mathcal{SP}, maxR$)
    ▷ Discuss payout part
end function

4 Results

Steem won't achieve high quality posts.

5 Further Work

Posts at any time

6 Conclusion

Keep inventing new decentralized content curation platforms.

7 Acknowledgements

We thank Prof. Aggelos Kiayias for constructive conversations, @serious-poster for their invaluable posts analyzing Steem and our mums for the cookies.