We illustrate the process overview of BEQA in Fig. 1. As illustrated in interaction 1 (upper portion of figure), to submit feedback $i = (e, v_i, pub)$ about an entity e, a user with a public-key pub first signs the feedback with its own private key PK. Thus, each user has a randomly generated key pair (not that of the Blockchain), which makes BEQA robust against *repudiation attacks*. Next, the user submits the signed feedback to Blockchain using a wallet (which is a software used in Blockchain to perform transactions). This will incur the user a transaction fee f_i; when the transaction is confirmed on Blockchain, the feedback is assigned a time-stamp t_i.

As illustrated in interaction 2 (lower portion of Fig. 1), when a user enquires about the trustworthiness of some entities $e_1, e_2, ..., e_n$, BEQA fetches from a database all the existing feedbacks for those entities as well as the corresponding fees and timestamps. This database is synchronized with the Blockchain (only on transactions relevant to the BEQA framework). Examples of database technologies, which can perform targeted crawling and indexing of Blockchain, can be found in [1, 2]. Next, BEQA verifies the user signatures of these fetched feedbacks. This prevents *identity attacks* in which an attacker associates its malicious feedback with a legitimate user's identity. Finally, BEQA's trust model, which is explained in the next section, uses Blockchain transaction fees to quantify the cost of whitewashing and Sybil attacks and provide a robust assessment of the trustworthiness of entities $e_1, e_2, ..., e_n$.

1.2 Details of TRM

Weighting feedbacks. BEQA assigns a weight of zero to feedbacks from publicity influencers, and hence filters out their opinions; but BEQA uses their total spend on transaction fees as a whitewashing deposit. BEQA weights the remaining of the feedbacks proportionately to the transaction costs incurred by the submitter. This costs are grown over time using an exponential growth function with an unknown growth rate. This way, BEQA quantifies the weight of a feedback as a function of this growth rate.

Formally, let f_j be the transaction fee of feedback j, A_e be the set of all feedbacks about entity e, and B_i be the set of all feedbacks submitted not later than feedback i by the submitter of i about all entities including but not limited to e. We obtain the publicity of an entity e from (1) where μ_x, σ_x^2, $\mathcal{S}_x$, and $\mathcal{K}_x$ are the first four statistical moments of the set $\{x_i = \sum_{j \in B_i} f_j \mid i \in A_e\}$ as follows:

$$P_e = \left\{ i \in A_e \,\middle|\, x_i \geq \mu_x + \sigma_x (1 - \frac{1}{\mathcal{K}_x}) \left(\frac{2}{\pi} \tan^{-1}(-|\mathcal{S}_x|) + 3 \right) \right\} \quad (1)$$

We define the weight of a feedback i about an entity e as w_i:

$$w_i = \begin{cases} \sum_{j \in B_i} f_j (1 + r_e)^{t_i - t_j} & i \notin P_e \\ 0 & i \in P_e \end{cases} \quad (2)$$

where t_i and t_j are the timestamps of feedback i and feedback j respectively, P_e is the set containing the *publicity* of entity e, and r_e in (2) is a growth rate that BEQA calculates for each entity dynamically.

Computing the growth rate r_e. Formally, given a set of entities E and publicity set P_e, where $e \in E$, we calculate r_e from (4) where R_0^{max} is obtained using (3).

$$R_0^{max} = \min \left\{ \operatorname*{argmax}_{r_e} \left(\sum_{i \in A_e} w_i < \sum_{i \in P_e} \sum_{j \in B_i} f_j \right) \,\middle|\, e \in E \right\} \quad (3)$$

$$r_e = \frac{\sum_{i \in P_e} \sum_{j \in B_i} f_j}{\max \left\{ \sum_{i \in P_{e'}} \sum_{j \in B_i} f_j \,\middle|\, e' \in E \right\}} \times R_0^{max} \quad (4)$$

Trust computation. Having obtained r_e for $e \in E$ from (4), we can compute the trustworthiness of each entity using (5).

$$T_e = \sum_{i \in A_e} v_i \cdot w_i \Big/ \sum_{i \in A_e} w_i \quad (5)$$

2 EVALUATION

We simulate a TRM scenario with three groups of entities. The only difference between the groups is their publicity expenditure which we define as the total spend of their publicity influencers. That is, the number of the total transactions of the influencers of an entity is a random variable with a mixture of uniform distributions with different means of 100, 200, and 400 and equal mixture weights. At each time-step of the simulation, each entity interacts with 3 percent of randomly chosen honest users. Then, these users each submits a feedback which incurs the user a transaction fee that we treat as the cost unit. We say a Sybil attack is successful if the feedbacks of the attacker about the target entity outweigh that of the honest users.

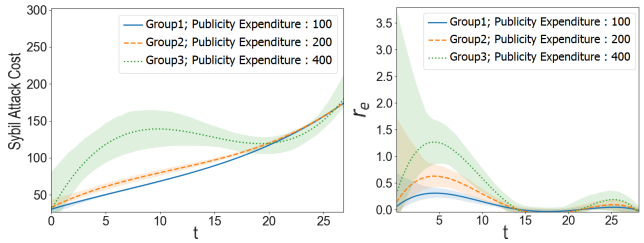

Figure 2: Simulation results: (a) Cost of a successful Sybil attack for entities with different whitewashing deposits at each time step; (b) r_e for entities with different whitewashing deposits at each time step.

We present the experiment results in Fig. 2(a)-(b). The results show that in general a higher publicity expenditure leads to a higher Sybil attack cost. This is a desired property that discourages Sybil attacks, and also encourages higher publicity expenditure as a deposit against white washing.

REFERENCES

[1] Planaria corp. 2019. Planaria: crawl and index the Bitcoin blockchain. Retrieved November 7, 2019 from https://github.com/interplanaria/planaria
[2] Matashard. 2018. Easily access data stored on Blockchain. https://metashard.com/. (Accessed on 6 Feb 2021).
[3] Satoshi Nakamoto. 2019. *Bitcoin: A peer-to-peer electronic cash system.* Technical Report. Manubot.
[4] Michael Bedford Taylor. 2017. The evolution of bitcoin hardware. *Computer* 50, 9 (2017), 58–66.
[5] Craig S Wright. 2008. Bitcoin: A peer-to-peer electronic cash system. *Available at SSRN* 3440802 (2008).