Limiting Fake Accounts in Large-Scale Distributed Systems through Adaptive Identity Management

Weverton Luis da Costa Cordeiro[1,2], Luciano Paschoal Gaspary[1]

[1]Institute of Informatics – Federal University of Rio Grande do Sul
Av. Bento Gonçalves, 9500 – 91.501-970 – Porto Alegre, RS, Brazil
{weverton.cordeiro,paschoal}@inf.ufrgs.br

[2]Federal Institute of Pará @ Itaituba
Rua Universitário, SN – 68.180-000 – Itaituba, PA, Brazil
weverton.cordeiro@ifpa.edu.br

***Abstract*—Various online, networked systems offer a lightweight process for obtaining identities (e.g., confirming a valid e-mail address), so that users can easily join them. Such convenience comes with a price, however: with minimum effort, an attacker can subvert the identity management scheme in place, obtain a multitude of fake accounts, and use them for malicious purposes. In this work, we approach the issue of fake accounts in large-scale, distributed systems, by proposing a framework for adaptive identity management. Instead of relying on users' personal information as a requirement for granting identities (unlike existing proposals), our key idea is to estimate a trust score for identity requests, and price them accordingly using a proof of work strategy. The research agenda that guided the development of this framework comprised three main items: (*i*) investigation of a candidate trust score function, based on an analysis of users' identity request patterns, (*ii*) combination of trust scores and proof of work strategies (e.g. cryptograhic puzzles) for adaptively *pricing* identity requests, and (*iii*) reshaping of traditional proof of work strategies, in order to make them more resource-efficient, without compromising their effectiveness (in stopping attackers).**

***Keywords*—Identity management, peer-to-peer, fake accounts, collusion attacks, proof of work, and sybil attack.**

I. Introduction

Identity and access management infrastructures play an important role in the digital era, enabling networked systems to determine who has access to them, what permissions one has to managed resources, how these resources can be accessed, etc. [1]. These infrastructures support a variety of functions, such as identity lifecycle management (e.g., creation, update, and revocation of identities), authentication, and access control [2]. In our research, we focused on identity management.

In an ideal scenario, the identity management scheme in place should allow only one identity per individual in a given system. Depending on the system nature (e.g., peer-to-peer and collaborative intrusion detection systems), this relationship could be read as "one device, one identity". The reality is very far from that, however. Online systems such as Facebook, Twitter, Digg, Skype, and BitTorrent (to mention a few) offer a lightweight process for creating identities[1] (e.g., confirming a valid e-mail address), so that users can easily join them. Such convenience comes with a price, however: with minimum effort, an attacker can easily subvert the identity management scheme in place and obtain a multitude of *fake accounts*[2] (Sybil attack [3]). These accounts can then be used to either perform malicious activities (that might harm legitimate users) or obtain unfair/illegal benefits. The corruptive power of sybils is widely known, being the object of several investigations [4], [5], [6].

It is extremely challenging (if not impossible) to devise a *one-size-fits-all* solution for identity management. As a consequence, the research community has focused on the design of system-specific solutions, in scenarios having a well-defined set of purposes, requirements, and constraints. In the thesis [7], we approached the issue of fake accounts in large-scale, distributed systems. More specifically, we targeted those based on the peer-to-peer paradigm and that can accomodate lightweight, long-term identity management schemes [8] (e.g. file sharing and live streaming networks, collaborative intrusion detection systems, among others); *lightweight* because users should obtain identities without being required to provide "proof of identity" (e.g., personal documents) and/or pay taxes; and *long-term* because users should be able to maintain their identities (e.g., through renewal) indefinitely.

In the scope of these systems, strategies such as certification authorities [9], [10], [11], trusted computing [12], blacklisting [13], [14], reputation [15], [16], social networks [4], [17], [6], [18], and proof of work (e.g., computational puzzles) [19], [20], [21], [22] have been suggested as promising directions to tackle fake accounts or stop malice in general. In spite of the potentialities, important questions remain (please refer to the thesis [7] for an in-depth discussion of the merits and drawbacks of existing classes of solutions). For example, a number of investigations [23], [24] have shown that some of the key assumptions on which social network-based schemes rely (e.g., sybils form tight-knit communities) are invalid. More importantly, social network-based identity verification might violate user's privacy, This is an extremely sensitive issue, specially because of the growing concern and discussion about privacy issues in social networks [25], [26], [27], [28].

Puzzle-based identity management schemes inherently preserve users' privacy (since puzzle-solving does not require any personal information), and thus represent an interesting approach to stop sybils. Existing schemes focus on the users' computing power, and use cryptographic puzzles of fixed

[1]The terms "account" and "identity" are used interchangeably to refer to an informational abstraction capable of distinguishing users in a given system.

[2]We use the terms "fake account", "sybil", and "counterfeit identity" interchangeably to refer to those identities created and controlled by an attacker, and which are used with malicious/illegal purposes.

978-1-4799-0913-1/14/$31.00 © 2014 IEEE

complexity to hinder attackers [19], [20]. However, puzzle-solving incurs considerable energy consumption, which increases proportionally to the system popularity. Furthermore, users waste computing resources when solving puzzles.

The main objective of the thesis was to propose a framework for adaptively pricing identity requests (using proof of work), as an approach to limit the spread sybils in large scale distributed systems. We based our framework on the hypothesis that "**one can separate presumably legitimate identity requests from those potentially malicious by observing their source of origin and users' identity request patterns**[3]". Based on this hypothesis, our key idea is therefore to estimate a trust score of the source from which identity requests depart, calculated as a proportion of the number of identities already granted to (the) user(s) associated to that source, in regard to the average of identities granted to users associated to other sources. The higher the frequency (the) user(s) associated to a source obtain(s) identities, the lower the trust score of that source and, consequently, the higher the price that must be paid per identity requested.

The research agenda that led to the development of our framework was consituted of three main items: 1) An investigation of a candidate trust score function for measuring the likeliness they were part of an ongoing sybil attack. The investigation involved a throughout analysis of users' identity request patterns in a large scale distributed system; 2) A combination of trust scores with proof of work, to propose the notion of adaptive puzzles. The goal is to adaptively adjust the complexity of puzzles based on the measured trust score. Therefore, those requests likely to be malicious will be assigned puzzles of higher complexity, whereas presumably legitimate will be assigned less complex ones; and 3) An analysis of resource-efficiency of existing proof of work strategies. We proposed a direction for reshaping traditional puzzles, in order to make them *green* (in terms of energy required to solve them) and useful (by recycling the processing cycles dedicated to puzzle-solving). In the end, our research led to the proposal of a novel design for lightweight, long-term identity management based on *green* and *useful* computational puzzles.

The research items mentioned above are approached in Sections II, III, and IV, respectively. We then close the paper in Section V with lessons learned and concluding remarks.

II. Trust Score Model

The trust score model forms the basis of our framework for adaptive identity management. Next we provide an overview of the model. Then, we present a brief evaluation considering traces of identity requests collected from BitTorrent.

A. From Users' Recurrence Patterns to Trust Scores

In the scope of the thesis, we defined *trust score* [30], [31] as an index that establishes the likeliness that some identity request, originated from a certain source, is presumably legitimate or potentially part of an ongoing attack.

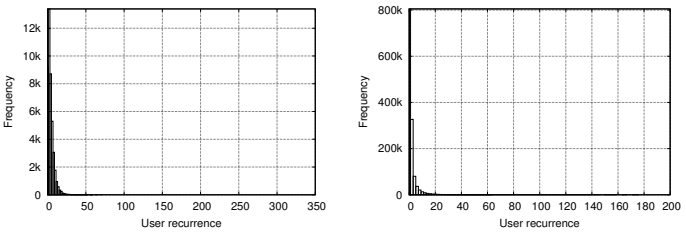
[3]In the context the thesis, "a *source* requests identities" means in fact "user(s), from a certain *source*, request(s) identities". *Source* may refer to a user's workstation, a local network, etc. (identified by an IP address or prefix). In substitution or as a complement, *source* may be a network coordinate provided by a system such as Veracity [29].

1) Analysis of Users' Recurrence Patterns: Our research for a trust score function was driven by the idea that a good candidate should consider the dynamics of users' recurrence patterns in large scale distributed systems, and ensure that they receive good reputation. To this end, it is important to characterize users' recurrence in these systems, and assess a baseline regarded as "regular behavior".

Our characterization basically consisted in evaluating aspects such as users' time and frequency of arrivals. From the files of users' participation in torrent swarms analyzed, we extracted traces that rebuild users' identity request events[4].

The analysis of collected traces revealed some important aspects for our research. First, users' arrival has shown to be consistent over the week, with a few access peaks, and was characterized by an increase in the number of joins during daytime (UTC), and subsequent descrease overnight. Second, various users left and re-joined the system within relatively short time intervals. These two aspects indicated that our candidate design for a trust score function should accomodate sazonal changes in users' behavior (within a window of hours, days, or even weeks), and enable users to rejoin the system with a certain frequency without being much penalized.

The most important aspect for our research was revealed with the analysis of users' recurrence. In one of the traces, the average number of identities assigned per IP address was 2.5 (in a period of one week); the 9th and 10th deciles were 5 and 180, respectively.Figure 1 provides an overview of the results achieved. In the traces studied, the majority of users joined the system very few times a week; the distribution of recurrences had the shape of either a power-law or exponential.



(a) Trace A (b) Trace B

Fig. 1. Distribution of users' recurrences.

In summary, the trace analysis has shown that a vast majority of users tend to access online systems with a relatively low frequency, during a given period. Attackers on the other hand shall present a discrepant behavior, by requesting a higher number of identities from a limited number of sources. Therefore, keeping track of sources' recurrence becomes a promising approach for the design of our trust score function.

2) Trust Score Function: The main input for computing trust scores is the number of identities already granted to the users associated to a given source. This number is defined as $\phi_i(t)$ for the *i-th* source, at instant t (with $\phi_i(t) \in \mathbb{N}$). Based on this information, we formally define the *source recurrence* ($\Delta\phi_i(t)$) and *network recurrence* ($\Phi(t)$, with $\Phi(t) \in \mathbb{R}$ and $\Phi(t) \geq 1$) metrics. The former, given by $\Delta\phi_i(t) = \phi_i(t) - \phi_i(t-\Delta t)$, represents the number of identities granted to users

[4]For ethical reasons, the traces collected have been fully anonymized *a priori*, in order to make it impossible any sort of user identification/tracking. We also assume that the traces contain legitimate identity request activity only, since no evaluation of peer identity vs. address was carried out.

associated to some specific source i, in the last Δt units of time. The latter corresponds to the average number of identities that sources have obtained in the same period.

The network recurrence metric $\Phi(t)$ is computed using the simple mean of the values of sources' recurrence, according to Equation 1. In this equation, n is the number of currently active sources, i.e., those that have obtained at least one identity within the interval Δt. Note that when $\Delta\phi_k(t) = 0$ for some source k, users associated to that source have not obtained any identity (during Δt); such a source can be safely ignored.

$$\Phi(t) = \begin{cases} 1 & \text{, if } n = 0 \\ \frac{1}{n} \times \sum_{i=1}^{n} \Delta\phi_i(t) & \text{, if } n \geq 1 \end{cases} \quad (1)$$

Observe that Δt serves as a bound for the portion of identity grants considered when computing the sources' (and the network) recurrence metrics, thus functioning as a "sliding window" that addresses the seasonality of users' access patterns. As the window slides forward, older identity grants are gradually discarded, thus allowing room to newer ones which are more representative of the current state of the system.

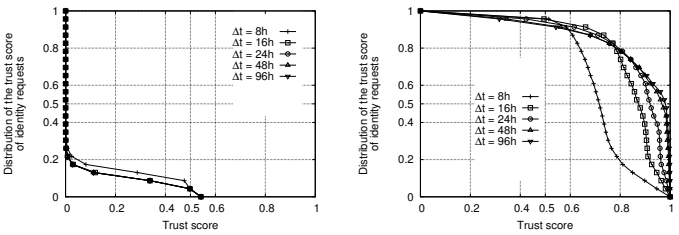
Recall from the trace analysis that a large fraction of users presented a similar, consistent behavior (as evidenced by users' recurrence metrics and distributions). For this reason, we use the average behavior of the network as baseline for "normal behavior". In this context, by comparing the behavior of a given source i (inferred from $\Delta\phi_i(t)$) and the network behavior (inferred from $\Phi(t)$), we calculate the *relationship between source and network recurrences* ($\rho_i(t)$, with $\rho_i(t) \in \mathbb{R}$). When negative, $\rho_i(t)$ indicates how many times the recurrence of the *i-th* source is lower than the recurrence of the network. Equation 2 provides the value of $\rho_i(t)$.

$$\rho_i(t) = \begin{cases} 1 - \frac{\Phi(t)}{\Delta\phi_i(t)} & \text{, if } \Delta\phi_i(t) \leq \Phi(t) \\ \frac{\Delta\phi_i(t)}{\Phi(t)} - 1 & \text{, if } \Delta\phi_i(t) > \Phi(t) \end{cases} \quad (2)$$

The relationship index $\rho_i(t)$ serves then as input for computing the trust score of the *i-th* source ($\theta_i(t)$). It is calculated at instant t according to Equation 3, and assumes values in the interval $(0, 1)$: on one extreme, values close to 1 denote a high trust on the legitimacy of (the) user(s) associated to the *i-th* source; on the other, values close to 0 indicate high distrust, i.e., a high probability that source hosts an attacker.

$$\theta_i(t) = 0.5 - \frac{\arctan(\Phi(t) \times \rho_i(t)^3)}{\pi} \quad (3)$$

In addition to addressing the seasonality of users' access patterns, abrupt but momentarily changes in their behavior should also be taken into account. For this reason, we compute the *smoothed trust score*. Defined as $\theta'_i(t)$ for the *i-th* source in instant t, it is calculated as shown in Equation 4. The smoothing factor β determines the weight of present behavior in the calculation of the smoothed trust score, assuming values in the interval $(0, 1]$. Thus, values of β close to 0 assign a high weight to the historical behavior of the source under consideration, and vice-versa. In Equation 4, $\theta'_i(t')$ refers to the last computed value of smoothed trust score.



(a) malicious (b) legitimate

Fig. 2. CCDF of trust score of requests, under different sizes for Δt.

$$\theta'_i(t) = \begin{cases} \theta_i(t) & \text{, if } t = 0 \\ \beta \times \theta_i(t) + (1 - \beta) \times \theta'_i(t') & \text{, otherwise} \end{cases} \quad (4)$$

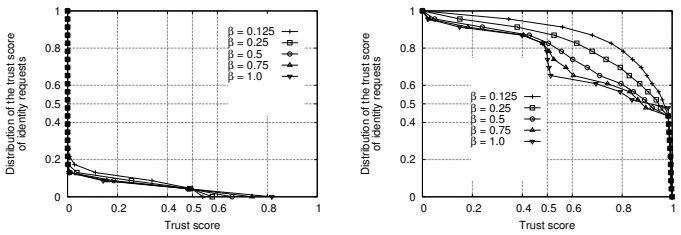
B. Evaluation

For evaluating the concept of trust scores, we implemented a bootstrap entity for use in a simulation environment; it aggregates the functionalities of management of identity requests from users interested in joining some system. We considered in our evaluation various traces of identity requests from BitTorrent file sharing communities. In summary, we had the goal of assessing the efficacy and effectivenes of our solution regardless of number of active users, amount of identity requested, users' behavior and seasonal effects, among others. We refer the reader to the thesis [7] for a detailed discussion of the results achieved. Due to space constraints, next we focus on Trace A; it contains 203,060 requests from 44,066 distinct sources, during a period of one week.

We also considered scenarios with and without attack. In the scenarios with attack, we considered two situations: one in which the attacker controlled 1% ($M_u = 1\%$) of sources of identity requests, and another in which she controlled 10% ($M_u = 10\%$) of sources (proportional to the number of sources seen in the traces used as input). The number of malicious requests was also defined proportionally to the number of legitimate ones seen in the trace, 1/3. This factor was chosen since it exceeds the proportion of fake accounts that sybil-resilient solutions tolerate [32], [33]. It is important to emphasize that these proportions represent extreme scenarios, favorable to the attacker. Otherwise, should she control a higher proportion of sources, she would have already outnumbered legitimate users, and launching a sybil attack would be purposeless.

In Figure 2(a) we present an evaluation considering different sizes of Δt: 8, 16, 24, 48, and 96 hours. Observe that malicious requests were significantly affected, being assigned extremely low values of trust scores for a majority of them, regardless of the values for Δt considered. For example, when using $\Delta t = 8$ hours, only 20.63% of malicious requests were assigned a value of trust score higher or equal to 0.01; in other words, over than 79.37% of requests were assigned extremely poor values of trust scores. Only 13.38% of malicious requests received values of trust score higher or equal to 0.1, and less than 3.79% were assigned values of trust scores of 0.5 or higher. No malicious requests were assigned a score higher than 0.6. Such low trust scores may be explained by the recurrence of each malicious source, which is comparatively higher than the average recurrence of the network.

Observe also from Figure 2(a) that the higher the duration of Δt, the more restrictive our solution becomes for the



(a) malicious (b) legitimate

Fig. 3. Trust score of requests, under different values for β.

attacker. For example, an increasing from $\Delta t = 8$ to $\Delta t = 16$ decreases from 11.7% to 7.75% the proportion of requests that were assigned values of trust score of 0.5 or higher. The reason is that a larger sliding window makes a higher fraction of the history of sources to be considered when calculating their recurrences $\Delta\phi(t)$, and thus those sources involved in an attack become more evident. With regard to legitimate requests, Figure 2(b) shows that the measured values were overall significantly high, regardless of Δt. The majority of identity requests from legitimate sources received a value of trust score higher or equal to 0.5 (above 92% in all cases).

In Figure 3 we show the effect that varying values for the smoothing factor β causes to the identity requests of these sources. We concentrate on the following values of β: 0.125, 0.25, 0.5, 0.75, and 1. For this set of experiments, we used $\Delta t = 48$. Focusing on Figure 3(a), observe that increased values of β makes a larger fraction of malicious requests be assigned lower values of trust score. For example, less than 3.8% of malicious requests received a value of trust score higher or equal to 0.5, when using $\beta = 0.125$; this proportion decreased to 2.2% when using $\beta = 1.0$. With regard to legitimate users, Figure 3(b) evidences that lower values for β decreases the overhead caused to legitimate requests. In spite of this, the proportion of requests assigned with a trust score higher or equal to 0.5 was above 76% in all scenarios, and above 92% in the particular case of $\beta = 0.125$.

Finding an appropriate setting for Δt and β is subjective, and basically depends on the system nature (more specifically, on how may identities are expected to be obtained per source in a given period) and the desired conservativeness with those sources potentially involved with an ongoing attack. In the evaluation scenarios described in the remainder of this paper, we chose a value of $\Delta t = 48$ and $\beta = 0.125$; these parameter settings make the concept of trust scores more robust to sources that continuously request more identities than the network average, whereas allows a legitimate user to request more identities during a transient failure (e.g., unstable network connectivity), without being (much) penalized for that.

III. Adaptive Identity Management

Proof of work has been long used in identity management [9], [19], [20], and also for stopping abusing behavior such as spamming [21], [22]. Although effective against sybils, existing defense schemes do not distinguish between identity requests from (presumably) legitimate users and attackers, requiring both to afford the same price per request. Next we discuss how we bridge this gap with adaptive puzzles [34].

A. Adaptive Pricing of Identity Requests

The mapping from trust score to puzzle complexity is given by an abstract function $\gamma : \Theta \rightarrow \mathbb{N}^*$. An actual instantiation of this function depends essentially on the nature of the adopted puzzle; for being effective, the puzzle must belong to the complexity class NP-complete. In the abstract function shown, the trust score $\theta_i(t) \in \Theta$ (of the i-th source) is mapped to a puzzle having exponential complexity, equivalent to $O(2^{\gamma_i(t)})$. An example of mapping function is given in Equation 5; note that the puzzle complexity is defined based on a maximum possible complexity Γ. In this equation, the constant 1 defines the minimum possible puzzle complexity.

$$\gamma_i(t) = \lfloor \Gamma \cdot (1 - \theta_i(t)) \rfloor + 1 \tag{5}$$

To illustrate an actual instantiation of the function above, consider the computational puzzle presented by Douceur in [3]: given a sufficiently high random number y, find two numbers x and z such that the concatenation $x|y|z$, after processed by a secure hash function, leads to a number whose γ least significant bits are 0. The time required to solve the proposed puzzle is proportional to $2^{\gamma-1}$, and the time to assert the validity of the solution is constant. Any puzzle having similar characteristics can be employed with our solution. There are other effective examples in the literature, such as [19], [20].

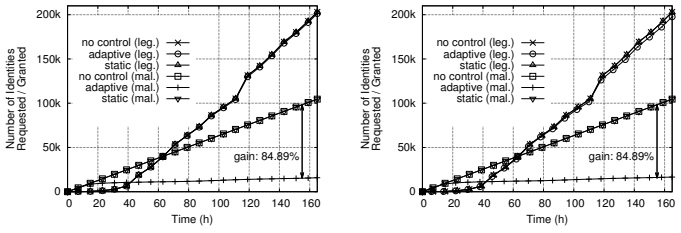
B. Evaluation

To evaluate the feasibility of using adaptive puzzles in limiting the spread of fake accounts, we extended the bootstrap entity and the simulation environment for implementing the dynamics of puzzle assignment and resolution. In summary, the extended entity aggregates the functionalities of managing identity requests from users interested in joining some generic system, assignment of puzzles for each identity request, validation of puzzle solutions received, and granting (or denial) of requests (according to the correctness of received solutions).

In order to model the delay caused by solving computational puzzles, we considered the puzzle presented by Douceur in [3] and described earlier. Further, for the sake of simplicity, we considered that a puzzle having complexity $\gamma_i(t_k)$ takes $2^6 + 2^{\gamma_i(t_k)-1}$ seconds to be solved in a standard device with normalized computing power of 1, for reference; a device twice faster takes half of this time to solve the same puzzle. For the computing power of legitimate users, we analyze cases in which it follows an Exponential ($\lambda_{cp} \approx 0.003$) and Gaussian ($\mu_{cp} = 1.2$ and $\sigma_{cp} \approx 0.4$). In both cases, it ranges from 0.1 to 2.5 times the computing power of the reference hardware. We consider such settings since they reflect the distribution of users' computing power as reported in the literature [35], [36], [37]. For the sake of comparison, we also included in our evaluation an identity management scheme based on static puzzles [19]; the complexity was defined as $\Gamma = 9$, which takes between 5 minutes and 2 hours (approximately) to solve.

Figure 4 shows the results obtained for each solution, for Trace A. For the sake of clarity and space constraints, only the results obtained for attack scenario $M_u = 1\%$ are shown. Observe that our solution outperforms both static puzzles [19] and the absence of control in limiting the creation of fake accounts. The proposed solution (curve "adaptive (mal.)") reduced in 84.9% the number of counterfeit identities granted, in comparison to the "no control" scenario (curve "no control (mal.)"). This represented an effective gain of 84.89% over the scenario where static puzzles were used (curve "static

(mal.)"); the use of such puzzles reduced marginally the number of granted counterfeit identities (0.05% only). Such a performance of the static puzzles is because the time required to solve them (five minutes, in the case of the attacker) was overall smaller than the interval between identity requests. In regard to the overhead caused to legitimate users, observe that the curve "adaptive (leg.)" (and also "static (leg.)") overlaps with the curve "no control (leg.)", thus indicating that the imposed overhead was negligible.



(a) Identities granted (Exponential) (b) Identities granted (Gaussian)

Fig. 4. Identities granted for each of the scenarios evaluated with Trace A.

Observe also that the results are marginally affected by the distribution of users' computing power considered. Comparing for example the number of identities granted to legitimate users and the attacker (Figures 4(a) and 4(b)), the difference observed is marginal. These results evidence that our solution is able to perform satisfactorily regardless of the environment settings (e.g. distribution of users' computing power) in place.

IV. Green and Useful Identity Management

Next we summarize our two-tiered solution for reshaping puzzles and making them green and useful [38], [39].

A. Towards Green Puzzles

The effectiveness of cryptographic puzzles comes from the fact that puzzle-solving is time-consuming. Puzzles that take one second to solve will barely stop attackers; to keep them away, it is important to assign puzzles that take longer to be solved. The more time users spend solving puzzles, the more energy is consumed, however. In a context of growing concern with rational usage of resources, the investigation for "green puzzles" becomes imperative.

In our proposal, we reduce the average puzzle complexity and complement them with "wait time". To illustrate, suppose that an effective puzzle should take five minutes to solve, on average. In our approach, instead of assigning such puzzle, we assign one that takes one minute; as a complement, once the user solves the puzzle, we "ask" him to wait four more minutes to obtain an identity. The strategy we envisage for defining the puzzle complexity depends on the process currently taking place, which can be either an *identity request* or *renewal*.

1) **Identity request process.** We use as input the smoothed trust score $\theta'_i(t)$ to estimate the puzzle complexity. The mapping function, defined abstractly as $\gamma : \Theta \rightarrow \mathbb{N}^*$, depends essentially on the nature of the adopted puzzle. In this function, $\theta'_i(t) \in \Theta$ is mapped to a computational puzzle having complexity equivalent to $O(2^\gamma)$. In an identity request process, the value of $\gamma_i(t)$ is estimated considering a differentiated, higher value of maximum complexity, $\Gamma = \Gamma_{req}$. The value of $\theta'_i(t)$ is saved in the identity ($I\langle\theta\rangle$), for later use during its renewal.

2) **Identity renewal process.** The value of $\theta'_i(t)$ used to estimate the puzzle complexity is computed based on $I\langle\theta\rangle$, according to Equation 7. Once the renewal process is complete, the bootstrap must save $\theta'_i(t)$ in the identity, for use in future renewal processes.

$$\theta'_i(t) = \beta \cdot 1 + (1 - \beta) \cdot I\langle\theta\rangle \quad (6)$$

To renew an identity, the user must solve a puzzle considering a lower value of maximum puzzle complexity, $\Gamma = \Gamma_{renew}$. If the identity has expired by the time the user renews it, the bootstrap entity must use another value of maximum puzzle complexity, $\Gamma = \Gamma_{reval}$. Note that making $\Gamma_{renew} < \Gamma_{reval} < \Gamma_{req}$ encourages users to renew their identities before expiration.

Similarly to the puzzle complexity, the waiting time should increase exponentially (e.g., proportionally to 2^ω, where ω is a wait factor), and be defined as a function of $\theta'_i(t)$. The design we consider for computing ω is given in Equation 7. In this function, Ω represents the maximum waiting time factor.

$$\omega_i(t) = \Omega \cdot (1 - \theta'_i(t)) \quad (7)$$

B. Towards Useful Puzzles

There are several proposals of cryptographic puzzles in the literature that can be used with our design to establish a cost for the identity renewal process [3], [19], [20]. An important characteristic of such puzzles is that their processing does not result in actual useful information. Here we propose a different type of puzzle, which takes advantage of the users' processing cycles to compute useful information.

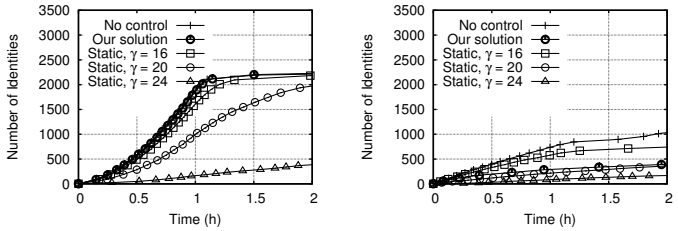
To assign a puzzle to be solved, the bootstrap entity replies to any identity request or renew messages (*i*) an URL that contains a piece of software that implements the puzzle (which can be a *useful* puzzle or a cryptographic one) and (*ii*) a set $\mathcal{J}$ of jobs (where each job is comprised of a number of input arguments to the downloaded piece of software). The puzzle complexity is given by $|\mathcal{J}|$.

An example of puzzle is a software that runs a simulation and generates the results using plain text. In this context, $\mathcal{J}$ contains a number of seeds, chosen by the bootstrap, that must be used as input to the simulation. Supposing that $\gamma_i(t_k) = 4$ (as computed from Equation 5), then $|\mathcal{J}| = 2^4 = 16$.

C. Evaluation

Our evaluation was carried out using the BitTornado framework on the PlanetLab environment. It had the goal of assessing the technical feasibility of green and useful adaptive puzzles, and also compare it with other approaches.

The parameter setting adopted in this evaluation attempted to replicate, in a smaller scale, the scenarios considered in our simulations. In summary, we considered an environment having 240 legitimate sources and 20 malicious ones. The legitimate users request 2,400 identities during one hour. The first request of each user is uniformly distributed during this period; their recurrence follows an exponential distribution, varying from 1 to 15 minutes. The interval between arrivals is also exponentially distributed, between 1 and 10 minutes. The attacker requests 1,200 identities ($1/3$ of the requests of legitimate users), making an average of 60 identities per malicious source; their recurrence follows a fixed rate of one request per minute. Our evaluation was defined observing the



(a) Legitimate (b) Malicious

Fig. 5. Results achieved with the PlanetLab environment.

technical constraints imposed by the PlanetLab environment (e.g., limited computing power, scarce and unstable nodes, and volatile network connectivity); due to these constraints, the identity renewal aspect of our solution could not be evaluated.

To make puzzles useful in our design, we used a software that emulates a small simulation experiment; it receives a list of random number generator seeds, and generates a single text file containing the results (for all seeds informed). The puzzle complexity is determined by the number of seeds informed, which in turn is proportional to $2^{\gamma_i(t)-1}$. For the mechanism based on static puzzles, we considered the one proposed by Douceur [3] (discussed earlier).

The other parameters were defined as follows. For our solution, $\Delta t = 48$ hours, $\beta = 0.125$, $\Gamma_{req} = 22$, and $\Omega = 10$. For static puzzles, we considered three scenarios: $\gamma_1 = 16$, $\gamma_2 = 20$, and $\gamma_3 = 24$. It is important to mention that the difference in the puzzle complexity, comparing the simulation model with the evaluation presented next, was necessary to adapt the puzzle-based mechanisms to the computing power constraints present in the PlanetLab environment.

Figure 5 shows that the dynamic of identity assignments to legitimate users with our solution (curve "Our solution") is similar to the no control scenario ("No control"). In contrast, it evidences the overhead/ineffectiveness of using static puzzles for identity management. Focusing on the attacker, our solution reduced significantly the number of fake accounts created (compared to the no control scenario).

The energy consumption estimates obtained also indicate the efficacy of our solution. While static puzzles with $\gamma_1 = 16$, $\gamma_2 = 20$, and $\gamma_3 = 24$ consumed 58.70 KJ, 533.85 KJ, and 803.92 KJ (respectively), our solution led to a consumption of 13.39 KJ only. It represents 22.81%, 2.41%, and 1.66% of the estimated consumption with static puzzles.

Our PlanetLab experiments also confirmed the effectiveness of using data processing jobs in replacement of cryptographic puzzles, as an strategy to make puzzles useful. In summary, the experiments carried out in the PlanetLab environment evidenced the technical feasibility of using adaptive puzzles, waiting time, and massive distributed computing for *green* and *useful* identity management.

V. Final Considerations

In this paper, we provided a brief overview of the research work carried out to develop a framework for adaptively pricing identity requests as an approach to limit the spread of fake accounts. Our framework explores the fact that sources in hands of attackers launch a significantly higher number of identity requests than those associated to presumably legitimate users. Based on this *request-to-source* ratio observation, we formulated a model that derives values of trust scores based on the frequency that sources of identity requests obtain identities, in comparison to the average number of identities already granted to other sources in the network. The model supports the concept of adaptive puzzles – a proof of work strategy that undermines the attacker's ability of controlling a large fraction of fake accounts in the network. In our research, we also reshaped traditional cryptographic puzzles in order to propose a lightweight design for *green* and *useful* identity management. An analytical evaluation of our framework [7] provided evidence that it is scalable and suitable for busy environments composed of millions of users.

A. Lessons Learned

In conclusion, the overall work presented in the thesis [7] and summarized in this paper underscores the importance of controlling the dissemination of fake accounts, and thus mitigating malicious behavior in networked, distributed systems. In the absence of controlling mechanism, attackers can easily control a large number of fake accounts, even outnumbering legitimate users in some cases. Existing approaches that achieve reasonable performance in stopping malice obligate users to provide personal information and/or pay taxes to create identities, which might not be desirable in many situations.

The combination of trust scores and adaptive pricing through proof of work has shown effective to hinder attackers, bounding the number of fake identities created, and increasing the (monetary) cost per fake identity. In this context, our research pushes the state of the art, by (*i*) providing a closer look at the problem of fake accounts, (*ii*) characterizing identity request patterns in the network, and (*iii*) introducing an identity management solution that both preserves users' privacy and makes it more expensive to engage on successful sybil attacks.

B. Thesis Deliverables and Final Remarks

The full thesis can be downloaded from http://hdl.handle.net/10183/90442. A subset of the thesis deliverables were published at renowned conferences and journal, namely, IEEE IM 2013 [39], Elsevier COMNET (2012) [34], IEEE CNSM 2011 [30], and SBRC 2014 (proceedings published in the IEEE Digital Library) [40]. There were also other publications in Brazilian conferences [31], [38], one of them awarded a best paper. Finally, there are deliverables in process of submission: one involving an in-depth analysis of identity request patterns (see Section 3.3 of the thesis [7]), plus an analytical evaluation of our framework (Section 6.4 [7]); and another involving an strategy for improving the quality of traces of identity requests (Sections 3.1 and 3.2 of the thesis [7]).

Our research was awarded a Microsoft Research Ph.D. Fellowship [41], and also a Microsoft Azure Research Grant (2014). A NOMS 2010 best student paper award was also received, for a publication done with a Ph.D. colleague [42].

Acknowledgements

The authors are grateful to Flávio Santos and Marinho Barcellos (UFRGS, Brazil) for the countless contributions made to our research. We also thank Francisco Brasileiro (UFCG, Brazil), Burkhard Stiller (University of Zurich, Switzerland), Alberto Montresor (University of Trento, Italy), and Jacob R. Lorch and John Douceur (Microsoft Research, USA), for the valuable discussions that helped improving our work. Our research was partially covered by a grant from CNPq (Project 560226/2010-1, Edital MCT/CNPq no. 09/2010 PDI), and a Microsoft Research Ph.D. Fellowship (2011).

REFERENCES

[1] K. Tracy, "Identity management systems," *Potentials, IEEE*, vol. 27, no. 6, pp. 34–37, 2008.

[2] M. Hansen, A. Schwartz, and A. Cooper, "Privacy and identity management," *Security Privacy, IEEE*, vol. 6, no. 2, pp. 38–45, 2008.

[3] J. R. Douceur, "The sybil attack," in *1st International Workshop on Peer-to-Peer Systems (IPTPS 2002)*, 2002, pp. 251–260.

[4] O. Jetter, J. Dinger, and H. Hartenstein, "Quantitative analysis of the sybil attack and effective sybil resistance in peer-to-peer systems," in *2010 International Communications Conference (ICC 2010)*, Cape Town, South Africa, may 2010, pp. 1 –6.

[5] H. Yu, "Sybil defenses via social networks: A tutorial and survey," *SIGACT News*, vol. 42, no. 3, pp. 80–101, Oct. 2011. [Online]. Available: http://doi.acm.org/10.1145/2034575.2034593

[6] L. Alvisi, A. Clement, A. Epasto, S. Lattanzi, and A. Panconesi, "Sok: The evolution of sybil defense via social networks," in *Security and Privacy (SP), 2013 IEEE Symposium on*, May 2013, pp. 382–396.

[7] W. Cordeiro, "Limiting fake accounts in large-scale distributed systems through adaptive identity management," Ph.D. dissertation, Federal University of Rio Grande do Sul, January 2014. [Online]. Available: http://hdl.handle.net/10183/90442

[8] G. Danezis and P. Mittal, "Sybilinfer: Detecting sybil nodes using social networks," in *2009 Network and Distributed System Security Symposium (NDSS 2009)*. San Diego, California, USA: The Internet Society, 2009.

[9] M. Castro, P. Drushel, A. Ganesh, A. Rowstron, and D. S. Wallach, "Secure routing for structured peer-to-peer overlay networks," in *5th Usenix Symposium on Operating Systems Design and Implementation (OSDI 2002)*, 2002, pp. 299–314.

[10] K. Aberer, A. Datta, and M. Hauswirth, "A decentralized public key infrastructure for customer-to customer e-commerce," in *Intl. Journal of Business Process Integration and Management*, 2005, pp. 26–33.

[11] R. Morselli, B. Bhattacharjee, J. Katz, and M. A. Marsh, "Keychains: A decentralized public-key infrastructure." 2006, http://hdl.handle.net/1903/3332.

[12] D. Levin, J. R. Douceur, J. R. Lorch, and T. Moscibroda, "Trinc: Small trusted hardware for large distributed systems," in *Proceedings of the 6th USENIX Symposium on Networked Systems Design and Implementation (NSDI)*, april 2009.

[13] J. Liang, N. Naoumov, and K. W. Ross, "The index poisoning attack in p2p file-sharing systems," in *25th IEEE International Conference on Computer Communications (INFOCOM 2006)*, Barcelona, Catalunya, Spain, 2006, pp. 1–12.

[14] SpamHaus, "The SpamHaus Project," 2013. [Online]. Available: http://www.spamhaus.org/

[15] A. Jøsang, R. Ismail, and C. Boyd, "A survey of trust and reputation systems for online service provision," *Decision Support Systems*, vol. 43, no. 2, pp. 618 – 644, 2007, emerging Issues in Collaborative Commerce.

[16] A. Jøsang, "Robustness of trust and reputation systems: Does it matter?" in *Trust Management VI - 6th IFIP WG 11.11 International Conference (IFIPTM 2012)*, ser. IFIP Advances in Information and Communication Technology, vol. 374. Springer, 2012, pp. 253–262.

[17] Q. Cao, M. Sirivianos, X. Yang, and T. Pregueiro, "Aiding the detection of fake accounts in large scale social online services," in *9th USENIX conference on Networked Systems Design and Implementation (NSDI 2012)*. Berkeley, CA, USA: USENIX Association, 2012, pp. 15–15.

[18] P. Liu, X. Wang, X. Che, Z. Chen, and Y. Gu, "Defense against sybil attacks in directed social networks," in *Digital Signal Processing (DSP), 2014 19th International Conference on*, Aug 2014, pp. 239–243.

[19] N. Borisov, "Computational puzzles as sybil defenses," in *6th IEEE International Conference on Peer-to-Peer Computing (P2P 2006)*, September 2006, pp. 171–176.

[20] H. Rowaihy, W. Enck, P. McDaniel, and T. La Porta, "Limiting sybil attacks in structured p2p networks," in *26th IEEE International Conference on Computer Communications (INFOCOM 2007)*, Anchorage, Alaska , USA, May 2007, pp. 2596–2600.

[21] B. Groza and B. Warinschi, "Cryptographic puzzles and dos resilience, revisited," *Designs, Codes and Cryptography*, vol. 73, no. 1, pp. 177–207, 2014. [Online]. Available: http://dx.doi.org/10.1007/s10623-013-9816-5

[22] T. L. Q. Bui, "Using spammers' computing resources for volunteer computing," Master's thesis, Portland State University, March 2014. [Online]. Available: http://archives.pdx.edu/ds/psu/11031

[23] A. Mohaisen, A. Yun, and Y. Kim, "Measuring the mixing time of social graphs," in *Proceedings of the 10th annual Conference on Internet Measurement*. New York, NY, USA: ACM, 2010, pp. 383–389.

[24] Z. Yang, C. Wilson, X. Wang, T. Gao, B. Y. Zhao, and Y. Dai, "Uncovering social network sybils in the wild," in *ACM SIGCOMM Conference on Internet Measurement Conference (IMC'11)*. New York, NY, USA: ACM, 2011, pp. 259–268.

[25] J. Angwin and J. Singer-Vine, "Selling you on facebook," *The Wall Street Journal*, 2012. [Online]. Available: http://online.wsj.com/article/SB10001424052702303302504577327744009046230.html

[26] B. Fung, "Whisper: The anonymous messaging app that reportedly tracks your location and shares data with the pentagon," 2014, [Online]. Available: http://www.washingtonpost.com/blogs/the-switch/wp/2014/10/16/whisper-the-anonymous-messaging-app-that-reportedly-tracks-your-location-and-shares-data-with-the-pentagon/.

[27] S. Jayson, "Social media research raises privacy and ethics issues," *USA Today*, March 2014. [Online]. Available: http://www.usatoday.com/story/news/nation/2014/03/08/data-online-behavior-research/5781447/

[28] G. Coleman, "Why the world needs anonymous," *MIT Technology Review*, November 2014. [Online]. Available: http://www.technologyreview.com/view/532381/why-the-world-needs-anonymous/

[29] M. Sherr, M. Blaze, and B. T. Loo, "Veracity: Practical secure network coordinates via vote-based agreements," in *USENIX Annual Conference (USENIX '09)*, June 2009.

[30] W. Cordeiro, F. R. Santos, G. H. Mauch, L. P. Gaspary, and M. P. Barcellos, "Securing p2p systems from sybil attacks through adaptive identity management," in *Mini-conference Proceedings of the 7th International Conference on Network and Service Management (CNSM 2011)*, October 2011, pp. 1–6.

[31] G. H. Mauch, F. R. Santos, W. Cordeiro, L. P. Gaspary, and M. P. Barcellos, "Dois pesos, duas medidas: Gerenciamento de identidades orientado a desafios adaptativos para contenção de sybils," in *28o Simpósio Brasileiro de Redes de Computadores e de Sistemas Distribuídos (SBRC 2010)*, Maio 2010, pp. 17–30.

[32] H. Yu, M. Kaminsky, P. B. Gibbons, and A. Flaxman, "Sybilguard: Defending against sybil attacks via social networks," in *2006 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM '06)*. New York, NY, USA: ACM Press, 2006, pp. 267–278.

[33] H. Yu, P. B. Gibbons, M. Kaminsky, and F. Xiao, "Sybillimit: A near-optimal social network defense against sybil attacks," in *IEEE Symposium on Security and Privacy*. IEEE Computer Society, 2008.

[34] W. Cordeiro, F. R. Santos, G. H. Mauch, M. P. Barcelos, and L. P. Gaspary, "Identity management based on adaptive puzzles to protect p2p systems from sybil attacks," *Computer Networks*, vol. 56, no. 11, pp. 2569 – 2589, 2012.

[35] NET MARKET SHARE, "Operating system market share," 2013. [Online]. Available: http://www.netmarketshare.com/operating-system-market-share.aspx

[36] CPU BENCHMARK, "Passmark cpu benchmarks - common cpu's," 2013. [Online]. Available: http://www.cpubenchmark.net/common_cpus.html

[37] ENIGMA @ HOME, "Cpu / os stats sorted by total credits granted," 2013. [Online]. Available: http://www.enigmaathome.net/cpu_os.php

[38] W. Cordeiro, P. Cemim, F. R. Santos, M. P. Barcellos, and L. P. Gaspary, "Segurança Verde: Usando Desafios com Espera Adaptativa para Conter Sybils em Redes Par-a-Par," in *30o Simpósio Brasileiro de Redes de Computadores e de Sistemas Distribuídos (SBRC 2012)*, Maio 2012, pp. 17–30.

[39] W. Cordeiro, F. R. Santos, M. P. Barcellos, and L. P. Gaspary, "Make it green and useful: Reshaping puzzles for identity management in large-scale distributed systems," in *IFIP/IEEE International Symposium on Integrated Network Management (IM 2013)*, april 2013, pp. 400 –407.

[40] W. Cordeiro, R. Mansilha, F. R. Santos, L. P. Gaspary, and M. P. Barcellos, "Were you there? bridging the gap to unveil users' online sessions in networked, distributed systems," in *32nd Brazilian Symposium on Computer Networks and Distributed Systems (SBRC 2014)*, May 2014, pp. 239 – 248.

[41] M. Research, "Microsoft research fellows in latin america," 2014. [Online]. Available: http://research.microsoft.com/en-us/collaboration/global/latam/fellows-latam.aspx#2011

[42] F. R. Santos, W. Cordeiro, L. P. Gaspary, and M. P. Barcellos, "Choking polluters in bittorrent file sharing communities," in *12th IFIP/IEEE Network Operations and Management Symposium (NOMS 2010)*, April 2010, pp. 559 – 566.