ISSN:2320-0790

An International Journal of Advanced Computer Technology

Stature System Protocols for Peer to Peer Networks: A Survey

Ankita Thadani[1], Vinit Gupta[2], IndraJeet Rajput[3]

[1, 2,3]Department of Computer Engineering, Hasmukh Goswami College of Engineering, Vehlal

Abstract: There are various websites presently used by us so the situation arises where people transact with unknown agents and take decision for these agents for by considering the stature score. Central idea of this paper is to compare online stature reporting systems that are particularly suitable for the peer to peer network but uses different approaches for calculating the stature of an entity. This paper describes the working of these stature systems, their properties and various parameters advantages and disadvantages. Finally, it concludes by comparison of all these stature system protocols.

Keywords*:* peer to peer network, privacy preserving, stature system

I. INTRODUCTION

Stature the word itself relates with the status i.e. what is one's status and by relying on that we do our activities for a person community or organization. major weakness of electronic markets is the raised level of risk associated with the loss of the notions of trust and stature .A unified view of trust by the source for which the stature is being calculated to the entities provides the stature of source entities for the other entities who needs to deal with the source entity for which the stature is going to be computed, Privacy is related to the feedback providers stature systems because anonymity deals with genuine feedback. The stature is strength and polarity of an opinion, for a context where the opinion is evaluated, merchants pricing power is affected by feedback it can incline or suppress negative ratings considering a person or an organization. To encourage resource sharing among peers and combat malicious peer stature score plays an important role for peer's trustworthiness so that one can deal with an entity which is more reputable one.

Stature system calculates the global stature score of a peer by taking in account the feedback values given by all other peers who have transacted with a particular pear. After completing a transaction, which can include a file download. The score calculated should be made publically available to the peers are, so that they can take informed decisions for the fact that which peers they can to trust in one context.

The transaction with an agent done in past, can be considered as the reliable source of information for that agent's stature. But simply depending directly on the past experience cannot be reliable as an. A single agent cannot transact with much number of other agents in the network.

II. DEFINATION AND TERMINOLOGIES RELATED TO STATURE

Reputation: perception that an agent creates Through past actions about its intentions and norms.[1] Reputation is a social quantity calculated based on actions by a given agent a_i and observations made by others in an "embedded social network"

Stature is what is generally said or believed about a person's or thing's character or standing [2]. This definition has the view that quantity derived from the underlying social network which is globally visible to all members of the network.

Trust and stature can be differentiated by normal and plausible statements:

(1) "I trust you because of your good stature."[2]

(2) "I trust you despite your bad stature." issue is how[2].

A. *Centralized Stature Systems:*

In Centralized stature system the feedback value is collected from agents in the community. The central authority collects all the values and computes a stature score on the basis of collected value from the agents, and publicly avails it. Agents can use these scores, by deciding whether or not to transact with a particular agent.

After each transaction, the agents can give score about the performance in the transaction.

The two fundamental aspects of centralized stature systems are:

1. Centralized communication protocols that allow participants to provide ratings about transaction partners to the central authority, as well as to obtain stature scores of potential transaction partners from the central authority.
2. A stature computation engine used by the central authority to derive stature scores for each participant, based on received ratings, and possibly also on other information.

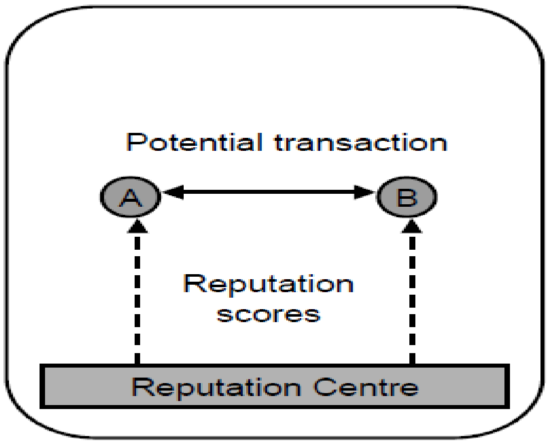

Figure. 1 Centralized reputation system [2]

B. Distributed Stature Systems:

A distributed stature system is without any centralized Functions. Instead of Central location for submitting feedback values stature scores of distributed authorities are present for submitting the feedback value or each participant records the opinion about each transaction with other parties, and gives information on request from trusted agents.

The stature score is computed based on the received ratings. Trusted agent should have had direct experience with the agent party.

Every node plays the role of both client and server, and is therefore sometimes called a servent

The purpose of a stature system in P2P networks is:

1. To compute which servents are most trusted.
2. To determine which servents provide the most reliable information with regard to (1).

it is often impossible or too costly to obtain ratings resulting from all interactions with a given agent.

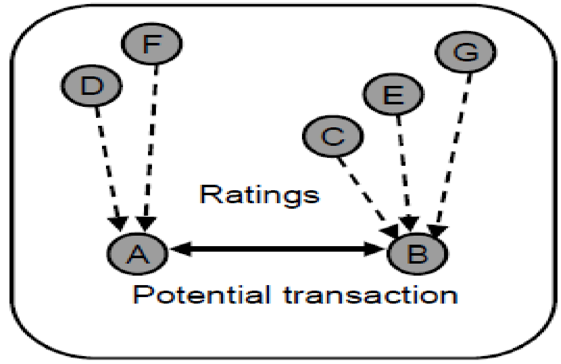

Figure. 2 Decentralized stature system [2]

C. The Identification of the general way can be classified by the three dimensions as being fundamental to any reputation system: [3]

Formulation: Before starting any protocol the ideal mathematical underpinnings of the reputation metric and the sources of input to that formulation. It can also be network formation For example, a system may accept Positive and negative feedback information weighted as +1 and −1 and defines an identity's reputation to be the summation of all of its corresponding feedback.

Calculation:

The algorithm to calculate the mathematical formulation for a given set of constraints physical distribution of participants, type of communication substrate, For example, the algorithm to calculate the formulation could Specify that a random set of peers is queried and the feedback received for each identity tallied. This mainly deals with the aggregation. Way of aggregation how the feedback values for the reputation is aggregated.

Dissemination:

The mechanism that allows system participants to obtain the reputation metrics resultant from the calculation is called dissemination. Such a mechanism may involve storing the values and disseminating them to the participants.

III. SOME STATURE BASED PROTOCOL

A. Jøsanget. et al. The Beta Reputation System[4]

Reputation system is based on the beta probability density function used to represent probability distributions of binary events so it can only handle the ratings positive, negative and neutral. The posteriori probability estimates of binary events can be :

The beta distribution $f(\rho|\alpha,\beta)$ by gamma functioncan be expressed using the gamma function as:

$f(\rho|\alpha,\beta) = \frac{\tau(\alpha+\beta)}{\tau(\alpha)\tau(\beta)}\rho^{\alpha-1}(1-\rho)^{(\beta-1)}$ Where $0\leq,\rho\leq 1$, $\alpha>0$, $\beta>0$.

With the restriction that the probability variable $\rho \neq 0$ if $\alpha < 1$ and $\rho \neq 1$ if $\beta < 1$. The probability expectation value of the beta distribution is given by
$E(\rho) = \alpha / (\alpha + \beta)$.
Feedback score of a transaction basically differs from the statistical observations of binary event, as known that an agent's satisfaction after a transaction is not binary.

This leads to the definition of the reputation function which is subjective that if agent provides feedback about target agent , then the reputation function resulting from that feedback represents the reputation as seen by feedback providing agent and not to be considered for representing target agents reputation from an objective viewpoint, because no such thing exists.

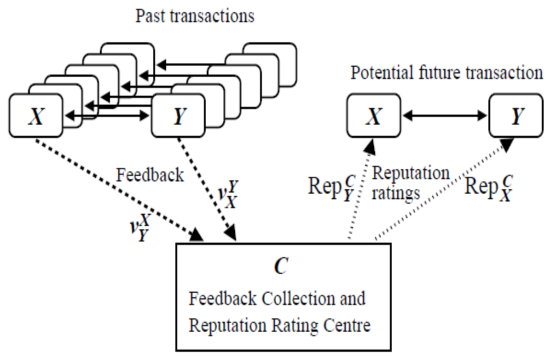

Figure. 3 Framework for collecting feedback and providing reputation ratings [4]

An engine calculates reputation score by the various feedback providers
Propagation which lets the agent to obtain reputation values when required. There are two available approaches for user reputation propagation.

In the centralized approach reputation values are stored in a central server, and whenever there is a need, users forward their query to the central server for the reputation value.

Following is the algorithm of the Beta Reputation System

The Reputation Function:
When dealing with the binary values the possible outcomes are $\{x, \bar{x}\}$. Initial step takes the integer number of past observations of x and $\bar{x}$ for estimating the probability of x,to predict the expected relative frequency with what will happen in the future in simple words for prediction.

The Reputation Rating
This step is ideal for mathematical manipulation, and less for reputation computation rating to human users simpler representation is needed the notion of a probability value is opted $E(\rho)$
reputation rating in the range [0,1] where 0.5 would be neutral rating.

Combining Feedback
By accumulating all the received parameters from the feedback provider the score is calculated. Assume two agents X_and Y providing feedback for target agent $T\varphi(\rho, r_T^X s_T^X)$ and $T\varphi(\rho, r_T^Y s_T^Y)$.The reputation function $\varphi(\rho, r_T^{X,Y} s_T^{X,Y})$ can be expressed as:
$r_T^{X,Y} = r_T^X + r_T^Y$
$s_T^{X,Y} = s_T^X + s_T^Y$
T's combined reputation function by X and Y

$$\varphi(\rho, r_T^{X,Y} s_T^{X,Y}) = \varphi(\rho, r_T^X s_T^X) \oplus \varphi(\rho, r_T^Y s_T^Y)$$

Discounting
Feedback value from high reputed agents carries more weight compared to feedback from agents with low reputation rating. So discounting the feedback is function of the agent providing the feedback. a metric called opinion to describe beliefs about the truth of statements

Forgetting
The old feedback value is not always be relevant for the actual reputation rating, as the agent may changes over time. The old feedback is given less weight than more recent feedback. A forgetting factor which can be adjusted according to the expected rapidity of change in the observed entity.

B. *Chris Clifton et at Securesum [5]*

Minimum three peers should be there and they should not collude with each other. It is multiparty computation [6]. The end of the computation, no peer knows anything except its own input and the final result.

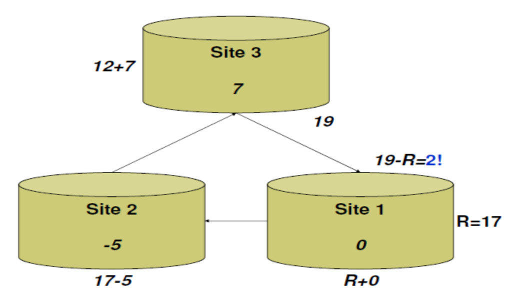

Figure. 4 Secure sum computation[5]

An **assumption** is at least three peers without collusion.
Assume that the value $v = \sum_{l=1}^{s} v1$ in the range of [0,..,n]
A site the master site, numbered 1 others from 2..s.

Initiation Each Site adds its feedback value v1, and sends the sum R + v1 mod n to site neighbored site. Value R is encrypt with a randomly chosen key

chosen uniformly from [1..n], the number R + v1 mod n is also distributed uniformly across this hence other site gets no idea about the actual value of v1.
For other sites the values can be computed V= R +$\sum_{j=1}^{l-1} vj$ mod n For site i R + $\sum_{j=1}^{l-1} vj$mod n= (vj + V) mod npasses it to site l + 1.
Sending result to master site: Site 1, then the master site subtracts R to get the actual result. site 1 can also determine the result by $\sum_{l=2}^{s} vj$subtractingv1. From the global result irrespective of the fact how it is calculated, so the master site does not get any information from it but faces problem if sites collude. As they can compare their values with each other Sites to determine the exact value for vl.

C. Gupta et al. DebitCredit Reputation Computation[7]

This reputation system is for p2p network to reliably calculate reputation score as a basis for an incentive system and suitable for multimedia upload and download.
There tunable system parameters
there in this protocol: File size factor f, f ∈integer, this parameters measures the level of MBytes data depending on increasing the reputation score. Bandwidth factor b, b ∈ real, identifies nodes for bandwidth Time factor in hours t, t ∈ integer.Period for the peer cooperation by sharing and staying online is rewarded

The reputation is computed by the agent called reputation computation agent to periodically update to the feedback providing agent's reputation, and to ensure that feedback value provided by them is kept locally so that it can be retrieved quickly. Reputation computation agent does not play any role while searching and retrieving so that it does become bottleneck for the normal operation of the P2P system:

Query-Response Credit (QRC) Agents initially need to register then they receive credit for providing their feedback to the system processing the query-response messages. key pair i.e. public and private key are generated on the registration. The agent chooses to send these proof of m process to the RCA(Reputation computing agent) for receiving the credits.
Upload Credit (UC): Each agent gets credit for providing any content related to multimedia and gets credit, (public, private) key pair is denoted here $\{PK_r, SK_r\}$ and sender peers by $\{PK_s, SK_s\}$. At the time of the file download For downloading {requester identity, file_name, file size, time stamp}and encrypt it with its private key and send to the up loader/sender agnates. On receiving the information from the above step and decrypting it by using the requester's public key and then encrypts the receipt of the transaction by its private key.
Download Debit (DD) While downloading a file an agent needs to debit for downloading the file. For negative reputation value, the RCA retains the negative scores in the form of debit state with itself until those peers send some credits for processing.
Sharing Credit (SC): Registered agents gets credit to be shared for staying online, based on the number of files they are sharing It can be achieved in two ways. First way deals with transaction state being recorded by RCA to check the time period for which particular agent was online and total amount of data shared by an agent. Second one periodic monitoring of the shared directories of agents by the RCA. But this method is more inaccurate Because the credit depends on the monitoring frequency.
Expiration and Consolidation of Reputation Scores: The time stamp is not important for it as the debit is there in the reputation scores. The peers can periodically send their reputation scores to the RCA for consolidation and get one encrypted score back.

D. Zhou et. al. The PowerTrust System Concept[8]

The Power Trust system is inspired by the power-law use Bayesian method to generate local trust scores where few power nodes are dynamically selected based on stature by using a distributed ranking mechanism is implemented by Distributed Hash Table (DHT) such as Chord [9] globally. Good stature for the power nodes is gathered by the running history of the system.

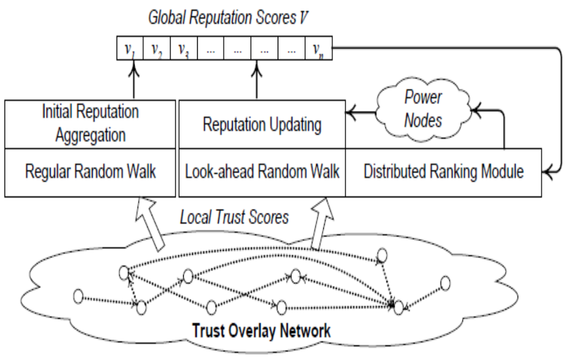

Figure . 5 the control flow pattern in local trust score collection and global stature aggregation [8]

A trust overlay network abbreviated as TON is built for all peers a P2P system. All peers evaluate each other, whenever a transaction takes place between a peer pair.
All global scores form a stature vector, V = (v1, v2, v3, …..,v_n), which is the output of the PowerTrust system. All global scores are the normalized.
The regular random walk module is initial stature aggregation. The look-ahead random walk is used to update the stature score, periodically works with a distributed ranking module to identify the power nodes.
Feedback frequency f_d is the number of nodes with feedback amount d.
The ranking index θ_d indicates the order of d in a decreasing list of feedback amounts.

Selection of top-m peers (Power nodes)
global statures stored among score managers are input
for each node i score manager j calculates, hash stature value $H(v_i)$ using locality preserving hash and insert the (v_i, i, j) to the successor node of $H(v_i)$ stored in the ascending order of their stature values in the DHT hash space due to the property of LPH.
initialize node x = successor node of the maximum hash value

Global Stature Aggregation
Local trust scores stored in the nodes are given as input to this step
for each node i& node j,the out-degree neighbor of node i is feed with score message (r_{ij}, i) to the score manager of node j
temporary variable pre=0 is initialized; the error threshold ε and global stature v_k of node k
For all received score pair (r_{jk}, j), where j is an in-degree neighbor of node k
Receive the global stature v_j from the score manger of node j
$v_k = v_k + v_j r_{jk}$
Compute $\delta = | v_k - pre|$ until $\delta<\varepsilon$output is Global stature for every node

Global Stature Updating Procedure:
The score managers collaborate with each other to find the power nodes by step 1.
If node x stores the triplet (i,v_i, j) and finds i as a power node, node x will notify to node j.
Local trust scores stored among nodes is the input to this step for each i& all node j Aggregate local trust scores from node j Send the score message (r_{ij}, i) to the score manager of node j
temporary variable pre=0; error threshold εglobalstaturev_k of node k
Initialize pre= v_k; v_k =0
For all received score pair (r_{jk}, j), where j is an in-degree neighbor of node k do
Receive node j global stature v_j from score manager of node j
For node k be a power node,
$v_k=(1-\alpha)\Sigma (v_j \times r_{jk}) +\alpha/m$
else $v_k=(1-\alpha)\Sigma (v_j \times r_{jk})$
$\delta = | v_k - pre|$, until $\delta<\varepsilon$
Global stature scores for all nodes for use by score managers collaboratively to find
the m most reputable nodes using is the output here.

E. Androulaki et al. A Reputation System for Anonymous Networks[10]

In this reputation system A peer agent is represented by a pseudonym and interact with each other by discarding pseudonyms such that their identity is not revealed to each other. These pseudonyms are unlikable the individual and the peers they share the same reputation score. The values of the reputation to each peer sum up to create that peer's reputation value which are publically made available,
anonymous credential systems, e-cash, and blind signatures. Reputation is exchanged in the form of e-coins called repcoins. The higher the amount of repcoins received from other users, the higher is the reputation of the user. A centralized entity bank, maintains the three data bases first the repcoin quota database which gives repcoin one peer can give to another

the reputation database: amount of repcoin earned by other peers and the history database to prevent for single time utilization of the points

Pseudonyms Generation

Each peer generates pseudonyms without registering with Bank. It just gives the random string for proving Ownership of the pseudonym.
P = f(r)
where f be one-way function, with zero-knowledge proof
p be the pseudonym and r be random string.
Digital signature is used where for signing and the pseudonym is for verification.

RepCoin Withdrawal.
Let B be the Bank. The U is peer and EC[6] be the e cash. First message is from user to bank, then bank verifies and then replies to the user in accordance to validity. A wallet W of n repcoins has been withdrawn. Repcoins are used to provide anonymity. And unique spending of the coins

Reputation Award
Can be simply stated reputation providing as Two pseudonyms are there in this step, it does not involves actual identities rather two pseudonyms are involved as no direct interaction but the pseudonym are used so no information of identities are revealed.

Reputation Update.
Takes place when a peer wants to increase reputation having the repcoins received presenting itself to Bank
And other peers as a pseudonym. But this cannot be simple as peer U wants to deposit a received repcoin as pseudonym everyone is unaware except U the owner of PU. So other peer may try to deposit the repcoin by to Bank as U. if peer's identity kwon then anonymity is not preserved. So peer contacts Bank gets blind permission been deposited, then deposits that blind permission.

Reputation Demonstration
For demonstrating ones reputation to other peer, both peers interacts using pseudonyms. For group G based on certain reputation levels, managed by Bank. For a peer to demonstrate reputation to peer verifier V, the bank holds the group and registers in the group G.
Peer contacts a Group and registers to the group by giving master public key the public key of group and a zero knowledge proof of knowledge that master secret key belongs to it has been created correctly and he is the owner.
Group checks that peer's reputation actually belongs to that group or higher, and then access Grant for credential.
Peer interacts with the verifier P under his pseudonym PU proves by executing Verify Credit having credential from group G. Specifically, PU proves that its owner has registered under a group of membership

F. Zhou et al Gossiptrust for fast reputation aggregation [12]

Gossiptrust deals with the fast aggregation of global stature scores. It deals with two steps within i.e. local score aggregation and global score dissemination are Performed.
Mathematically, for stature calculation we need to compute the weighted sum of all local scores s_{ij} score given by I for node j foreach peer j= 1, 2, …,n , where the values of the feedback score normalized global scores and weights are applied.

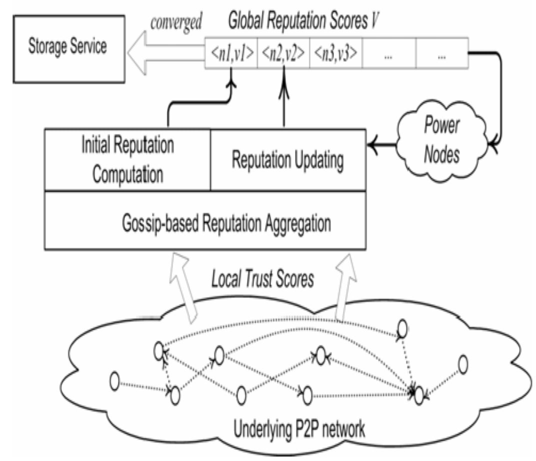

Figure. 6 Working of gossip group protocol [13]

Consider it for node N, here each node keeps a row vector of trust matrix S based on its outbound local trust scores.
At each node the global reputation vector V (t) is which has {node_id,score} pair.

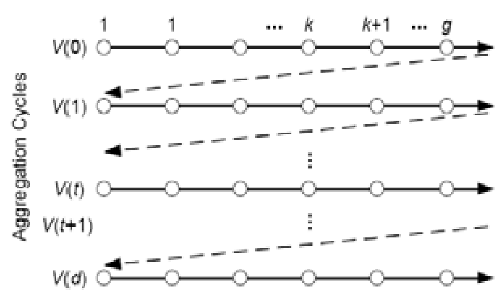

Figure. 7 Working of Gossip trust reputation aggregation cycle[12]

Vector initialization
Initially the global reputation vector is V(0) .

Recursive matrix vector calculation

Then matrix vector is calculated by aggregation process recursively,
$V(t+1) = S^T \times V(t)$
t is the iterative cycle. S is global score and T is trust parameter.

Exchange of global reputation:
Vectors are exchanged from every node to other, which are combined with current reputation vector, and the updated score is sent to a random node in the network.

Gossip aggregation of reputation:
local score s_{ij}, global score $v_i(t-1)$
for $i = 1,2,\ldots,n$ and gossip threshold ε
$xi \leftarrow s_{ij} \times v_i(t-1)$ weighted score x_i is initialized
if (i == j), set wi ← 1, else wi = 0 consensus factor wi
$k \leftarrow 0$ k is gossip step
$u \leftarrow x_i/w_i$ is previous score $\{(x_r, w_r)\}$ is gossip pair sent to i in previous step
$x_i \leftarrow \Sigma_r x_r$, $w_i \leftarrow \Sigma_r w_r$ Update the score and weight updated score is sent to a random node in the network($\frac{1}{2} x_i$, $\frac{1}{2} w_i$) to node it and itself
$k \leftarrow k+1$ Next gossip step
until $|x_i/w_i - u| \le \varepsilon$
$v_j(t) \leftarrow x_i/w_i$

Storage of global reputation
For achieving the memory efficiency on each node, Bloom-filter scheme for storage and retrieval of ranked globalscores is used . A Bloom filter is a space-efficient data structure for membership queries. They store the global scores. Each Bloom filter requires m bits tohold multiple hashed encodings into the same class.

G. *Kerschbaumet al The coercion-free stature System [14]*

This system provides complete privacy of the ratings, i.e. neither the feedback provider nor the stature system will learn the value of the rating. Here both cryptographic as well as a non-cryptographic approaches.

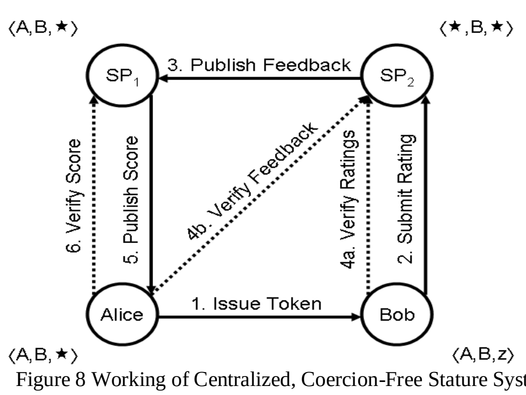

Figure 8 Working of Centralized, Coercion-Free Stature System [13]

An overview of this stature system is depicted in Figure above and its steps proceed as follows.

1. Alice (A) and Bob (B) two entities engage in a transaction. Alice issues Bob a token, that to give feedback. Token should be issued before the result of the transaction is known. else it should be refused, if the result was negative and prevent Bob from leaving negative feedback. No transaction should be engaged without having token for feedback first.
2. Bob leaves his feedback rating with SP2.
3. SP2 collects feedback from several raters and publishes all feedback on a public bulletin board.
4. All the feedback providers verify the published feedback that no feedback for them is present for which they did not issue a token. All raters verify in the published feedback that each rating is as they left it.
5. SP1 computes the aggregate stature score for each ratee and publishes it in the same bulletin board.
6. All the feedback providers verify that SP1 has computed their score and according to the left feedback.

Assumptions

An unique identity, e.g. through a public key infrastructure Denoted by SX() a signature using the private key of party X.

Parties can rate as well as be rated. binary ratings $z \in \{0, 1\}$ in this section where 1 denotes a positive rating and 0 a negative rating.

Registration:
An entity initially needs to register randomly chooses two secret keys $s \in Zp$ and $t \in Zp$. Sends public keys gs to SP1 and gt to SP2. SP1 publishes a list with all public keys gsX and their identities X or alternatively issues a certificate. In the same manner SP2 does for gt

Token Issue:
For a transaction between Alice and Bob .Alice issue a token. Alice chooses a random number r $\in Zp$ sends to Bob $\alpha = gr$, $\beta = grs$, $\gamma = grt$,
Bob verifies that $e(\alpha, gs) = e(\beta, g)$ and that $e(\alpha, gt) = e(\gamma, g)$. Alice keeps a copy of r & record of the transaction. re-randomization is done to make token unlinkable for SP2 and rely on Alice identifying any feedback forged by Bob.

Feedback Submission:
Bob gives his feedback z and encrypts by homomorphic encryption ESP1(z).chooses two random numbers l and m $\in Zp$. sends $\delta = gr$, gl,ESP1(grl), grsAl, $\varrho = gm$, $\zeta = grm$, $\eta = grtm$, $\theta =$

ESP1 (z), SB(gr, gl,ESP1 (grl), grsl, gm, grm, grtm,ESP1(z)) to the second service provider SP2. SP2 verifies with token generated.
Feedback Publication SP2 publishes collected feedback values and publish ι = gr, κ = gl, κ = ESP1 (grl), μ = grsl, ν = gm, ξ = grtm, ο = ESP1 (z) Alice scans and checks whether e(ι, κ)s = e(μ, g) true, Alice will conclude it will be used for stature computation record r matching gr = ι verifies that e(ν, gt)r = e(ξ, g) if fails, she claims that the feedback is forged and initiates an investigation Bob could similarly scan all feedback and check whether e(ι, ν)t = e(ξ, g), but he performs an inverse check by comparing δ = ι that his rating is unchanged ο = θ = ESP1(z). If any check fails, he similarly claims a forged feedback.
Stature Score Computation: SP1 decrypts DSP1 (λ) = π = grl Checks e(ι, κ)s = e(π, g) For all gsX checks whether e(π, gsX) = e(μ, g) If true, SP1 it should use this feedback& decrypts z = DSP1 (ο), computes stature score, publishes that score along with Alice's identity. SP1 claims a forged feedback, if any corresponding gsX and cannot use it in any score computation. SP1 must produce and publish a zero knowledge proof (ZKP) for the correct computation of the score from the ciphertexts ο.
Dispute Resolution: If any party claims that any feedback has been forged, a trusted third party D is called upon. Each party presents as evidence the published feedback and the judge D decides which party is at fault If a party can prove its innocence the next party will be accused. SP2's proof is the signature SB(gr, gl,ESP1(g_rl), gr_{sl}, gm, grm, grtm,ESP1(z)) submitted by Bob. D verifies the equality of each entry in the signature with the published feedback and if all checks succeed it accepts the proof. Bob's proof is the signature SA(g_r, g_{rs}, g_{rt}) received with the token. D verifies that gr = ι, e(κ, g_{rs}) =e(μ, g) and that e(ν, g_{rt}) = e(ξ, g). If all checks succeed, itaccepts the proof. false claims of a forged feedback in will be erased.
Leaving Self Feedback
This is case for forged positive feedback, No party alone can decide whether Bob has left feedback for himself. By increasing the service provider SP2's view to include the ratee, sendg_{rl} instead of its ciphertext ESP1(g_{rl}), but there exists a more privacy-preserving solution. Bob publish g_{st} can be verified by checking e(g_s, g_t) = e(g_{st}, g). Bob submits another value gr2lm with his feedback. SP2 then checks if e(g_{rls}, g_{rmt}) and e(g_{r2lm}, g_{st}) differ. without SP1 being able to link the feedback to gt either, which revealing grm would do. SP2 does so by choosing a random number n ∈Zp and publishing grmn and gr2lmn along with the feedback.

there are two service providers SP1 be the first service provider for stature and SP2 be the second stature service provider. X is the set of all ratees and raters.

H. Hasan, et al decentralized privacy preserving reputation protocol [15]

Each source agent s relies on at most k agents to preserve its privacy. On its own knowledge of their trustworthiness in the context of preserving privacy and sends each of them an additive shares of his private feedback value.

Initiation & Select Trustworthy Agents Is done by querying agent for computation of the reputation of a target agent .Source agent gets the feedback providers in a context.(advogato trust metric[16] is used here for this purpose). Each agent can selects up to k other agents with the probability that the selected agents will break agent's privacy is low
Prepare Shares At a time the source agent makes the k other feedback providing agents the number one decides is stated as K [17]. Agent prepares k + 1 share for secret feedback the k shares are random numbers uniformly distributed over a large interval. But the last k+1 share (Fat-∑ individual feedback) mod M.M is publically known Fat be feedback of a source agent a about a target agent t
Encrypt Shares: the list of all shares is implemented by agents own public key so that only agent can open it also each k th share is encrypted by public key of the feedback agent so that only one can have access to its own share by

once private key
Generate Zero-Knowledge Proofs Agent a computes: for an agent the zp(zero knowledge proof) zp=(E(1) x…xE(k+1)) mod n2 public rsa modulus [18]. The output of this product is then further encrypted sum of agents shares, Ea (additive homomorphic property). Two zero knowledge proof are there non-interactive set membership zero-knowledge proof: its non interactive as interaction is not needed and proves to a that the ciphertext has an encrypted value that lies in that is the ciphertext contains feedback value within range. Non-interactive plaintext equality zero-knowledge proofs. here the two ciphertexts, encrypted with the public key of feedback provider and other encrypted with the public key of whole list, contain the same plaintext. Assuring that agent a has prepared the shares such that they add up to a correct feedback value and are trustworthy agents correspond to those correct shares.
Send Encrypted Shares and Proofs All encrypted shares & zero-knowledge proofs are sent simply for feedback providing based on trusted agents.
Verify the Proofs. Each agent computes zp and verifies proofs received from agent that shares are prepared correctly.
Relay the Encrypted Shares. Agent relays to each agent a, the encrypted shares received for it from trustworthy agents. Where, each encrypted share is combined, any agent who drops a message would be detected without learning any of the shares.
Compute Sum of the Shares. Each agent receives the encrypted shares of trustworthy feedback providers. Agent computes as the product of those encrypted shares along with the ciphertext of its own k + 1th share by additive homomorphic property. Agent decrypts to obtain the plaintext sum and by adding the ka + 1'th share provides security
Encrypt the Sum. Agent a then encrypts the sum with k+1 from previous step the sum of the shares correctly And Compute Reputation
Generate Zero-Knowledge Proof. Agent generates a non interactive plaintext equality zero-knowledge proof, assures proof has the correct sum of the shares.
Send Encrypted Sum and Proof. Agent a sends the encrypted sum and the zero-knowledge proof to query agent
Verify the Proof. Query agent computes a and verifies the zero-knowledge proof received from each agent a. which assure agent has computed

TABLE I COMPARISION OF STATURE SYSTEMS

Sr No.	System/ Protocol	Architecture	Pros	Cons	Suitable for
1.	Jøsang et al. [4] The Beta Reputation System	Centralized	flexible and simple to implement	Immunity against agents changing identities. Can only be used for binary values	supporting electronic contracts and for building trust between players in e-commerce
2.	Chris Clifton et al [5] Securesum	Decentralized	Everyone knows only its own feedback value	Minimum 3 peers required and with no collusion	Honest multiparty omputation
3.	Gupta et al[7] DebitCredit Reputation Computation	Decentralized	Short term misuse of reputation	Less secure for the receipt off the message	incentive system and can guide peers in their decision making (e.g., who to download a file from
4.	Zhou et al[8]The PowerTrust System Concept	Decentralized	Low overhead in using locality-preserving hashing to locate power nodes. robust with dynamic peer join and leave and malicious peers	Complicated local and global computation	Malicious peer network
5.	Androulaki et al. [10] A Reputation	Decentralized	represented by a pseudonym	bank, which is a centralized entity.	P2p malicious adversary

	System for Anonymous Networks			no negative feedback	
6.	Zhou et al [11] Gossiptrust for fast reputation aggregation	Decentralized	Not requires secure hashing or fast lookup mechanism	Bloom filter makes it complicated	fully distributed p2p network, ranking systems
7.	Kerschbaum et al [12]The coercion-free stature System	Centralized	Ratings kept private from ratee and reputation system. does not require a central registry of transactions enabling it to be used in an open community	no one colludes with any of the service providers SP1 and SP2, including themselves	Centralized token issuing system, business transactions
8.	Hasan, et al [15] decentralized privacy preserving reputation protocol for the malicious adversarial	Decentralized	Zero knowledge transferred Secure ,robust	Can't prevent slandering	malicious adversarial, reputation systems.

IV. CONCLUSION

This paper has surveyed the literatures on reputation models across diverse disciplines. The centralized as well as decentralized different aggregation methods for peer to peer network. Disadvantage of each of the protocol has been pointed out. We have attempted to integrate our understanding across the surveyed literatures any tried to find out the one system proving the privacy and with strong cryptography building blocks.

ACKNOWLEDGMENT

I acknowledge here my debt to those who have contributed significantly in this survey paper. I indebted to my internal guide Mr. Vinit Gupta,Department of Computer Engineering, Hasmukh Goswami College of Engineering, Vehlal,Gujarat Technological University for helping me and his experience is very helpful to me.

REFERENCES

[1] Mui, Lik, MojdehMohtashemi, and Ari Halbersta dt.,*A computational model of trust and reputation.*,System Sciences, 2002. HICSS. Proceedings of the 35th Annual Hawaii International Conference on. IEEE, 2002.

[2] Jøsang, Audun, Roslan Ismail, and Colin Boyd. *A survey of trust and reputation systems for online service provision*, Decision support systems 43.2 (2007): 618-644.

[3] Hoffman, Kevin, David Zage, and Cristina NitaRotaru. ,*A survey of attack and defense techniques for reputation system*s. ACM Computing Surveys (CSUR) 42.1 (2009): 1.

[4] Jøsang, Audun, and Roslan Ismail,*The beta reputation system*, Proceedings of the 15th bled electronic commerce conference. 2002.

[5] Clifton, Chris, et al,*Tools for privacy preserving distributed data mining*.ACM SIGKDD Explorations Newsletter 4.2 (2002): 28-34.

[6] B. Schneier, *Applied Cryptography*. John Wiley & Sons,2nd edition, 1995.

[7] Gupta, Minaxi, Paul Judge, and MostafaAmmar., *A reputation system for peer-to-peer networks*, Proceedings of the 13th international workshop on Network and operating systems support for digital audio and video. ACM, 2003.

[8] Zhou, Runfang, and Kai Hwang, *Powertrust: A robust and scalable reputation system for trusted peer-to-peer computing*, Parallel and Distributed Systems, IEEE Transactions on18.4 ,2007.

[9] I. Stoica, R. Morris, D. Nowell, D.Karger, M. Kaashoek, F. Dabek, and H. Balakrishnan, *Chord: A Scalable Peer-to-Peer Lookup Protocol for Internet applications*", Proceedings ofACM SIGCOMM, San Diego, Aug. 2001.

[10] Androulaki, Elli, et al, *Reputation systems for anonymous networks*, Privacy Enhancing Technologies. Springer Berlin Heidelberg, 2008.

[11]Camenisch, Jan, Susan Hohenberger, and Anna Lysyanskaya, *Compact e-cash*. Advances in Cryptology–EUROCRYPT 2005. Springer Berlin Heidelberg, 2005. 302-321.

[12] Zhou, Runfang, Kai Hwang, and Min Cai. , *Gossiptrust for fast reputation aggregation I peer-to-peer networks*. Knowledge and Data Engineering, IEEE Transactions on 20.9 (2008):1282-1295..

[13] Zhou, Runfang, and Kai Hwang, *Gossip-based reputation aggregation for unstructured peer-to- peer networks*. Parallel and Distributed Processing Symposium, 2007. IPDPS 2007. IEEE International. IEEE, 2007.

[14] Kerschbaum, Florian, *A verifiable, centralized, Coercion-free stature system*. Proceedings of the 8th ACM workshop on Privacy in the electronic society. ACM, 2009.

[15] Hasan, Omar, et al., *A decentralized privacy preserving reputation protocol for the malicious adversarial model.*, Information Forensics and
Security, IEEE Transactions on 8.6 (2013): 949-962.

[16] Levien, R. , *from Advogato Website*, Advogato Trust Metric, URL: http://www. advogato.org/trust-metric. html.

[17] Hasan, Omar, Lionel Brunie, and Elisa Bertino. "k-shares: A privacy preserving reputation protocol for decentralized environments." Security and Privacy–Silver Linings in the Cloud. Springer Berlin Heidelberg, 2010. 253-264.

[18] Boneh, Dan. "Twenty years of attacks on the RSA cryptosystem." Notices of the AMS 46.2 (1999): 203-213.