Reputation Protocol for Diversity Network: A Survey

Neha Chauhan[1] Falguni Patel[2]
[1,2]Department of Computer Engineering
[1,2]SVIT,Vasad Gujarat Technological University

Abstract— Now a day, several websites presently used by us but the situation arises when the people in contact with unidentified agents and take the decision regarding them by considering the reputation score. core idea of this paper is to compare the reputation system for variety of network but it uses different approaches for computing the reputation of an entity. This paper describes the working of these reputation systems, their properties and various parameters advantages and disadvantages. Finally, it concludes by comparison of all these reputation system protocols.

Key words: privacy preserving, reputation system, information security

I. Introduction

Reputation systems represent a key technology for securing distributed applications from misuse by untrusted entities. Examples of some reputation systems may be found in several application domains: E-commerce websites such as eBay (ebay.com) and Amazon (amazon.com) use their reputation systems to dispirit fake activities. The EigenTrust reputation system which enable peer-to-peer file sharing systems to clean out peers who provide inauthentic content. The web-based community of Advogato.org uses a reputation system for spam filtering.

A reputation system calculates the reputation score of an entity as the aggregate of the feedback provided by the other entities. Reputation score which help recognize the entities that are display unwanted behavior The reputation score of a target entity is a function of the feedback values provided by other entities. Thus an accurate reputation score is possible only if the feedback is accurate. authority collects all the values and computes a reputation score on the basis of collected value from the agents, and publicly avails it. Agents can use these scores, by decide that to transact with a particular agent or not. [1]

Reputation can be defind as perception that an agent creates through the past actions about with its intents and norms. Reputation is a social quantity calculated based on actions by a given agent ai and observations made by others in an "embedded social network" reputation is what is generally said or believed about a person's or thing's character. This definition has the view that quantity which is 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) "I trust you despite your bad stature

Trust and reputation mechanisms have been proposed in various fields such as distributed computing, agent technology, grid computing, economics and evolutionary biology [2]

A. Centralized Reputation Systems:

In Centralized stature system the feedback value is collected from all agents in the community. The central authority collects all the values and Computes a stature score on the basis of collected Value publically avail it. The two fundamental features of centralized stature systems are:

1) Centralized communication protocols which allow participants to give ratings about transaction partners to the central authority, as well as to get reputation scores of transaction partners from the central authority.
2) A reputation computation engine used by the central authority to obtain reputation scores for all participant which based on received ratings, and maybe also on other information

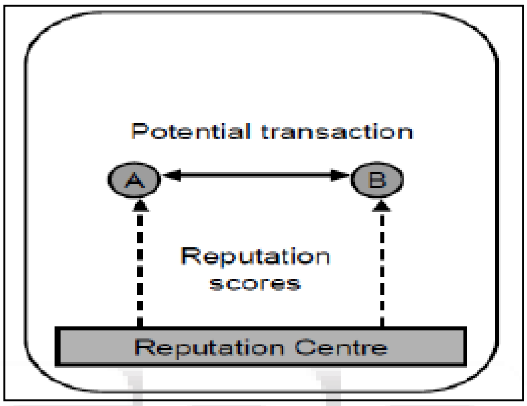

Fig. 1: Centralized reputation system [3]

B. Distributed Reputation Systems:

A distributed reputation system is without any centralized Functions. Instead of Central location for submitting feedback values reputation scores of distributed authorities are present for submitting the feedback value or all participant list the opinion about each transaction with other parties, and gives information on request from trusted agents. The reputation score is computed based on the received ratings. Every node plays the role of both client and server, and is therefore sometimes called a servent The purpose of a reputation 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.

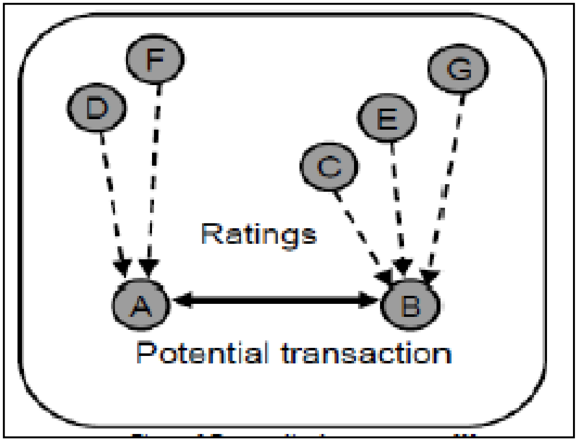

Fig. 2: Decentralized stature system [3]

II. Some Reputation Based Protocol

A. *Jøsanget. Et Al. The Beta Reputation System[4]:*

Reputation system is based on the beta probability density function so it can be used to represent probability distributions of binary events so it provides a mathematical base for combining feedback and for expressing reputation ratings and thats why so it can only handle the ratings positive, negative and neutral. The posteriori probability estimates of binary events can be : The beta distribution f(ρ|α, β) by gamma functioncan be expressed using the gamma function as:

$$f\ \rho\ \alpha, \beta = (\tau(\alpha+\beta))/(\tau(\alpha)\tau(\beta))\ \ \rho\text{^}(\alpha-1)\ \ [\![(1-\rho)]\!]\ \text{^}((\beta-1))$$

$$\text{Where}0\leq,\rho\leq1,\ \alpha>0,\ \beta>0.$$

With the restriction that the probability variable ξ≠0 if α<1 and ξ≠1 if β<1.The probability expectation value of the beta distribution is given by E (ξ) =α/ (α+β). 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 is lead to the definition of the reputation function which is subjective that if agent gives feedback just 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.

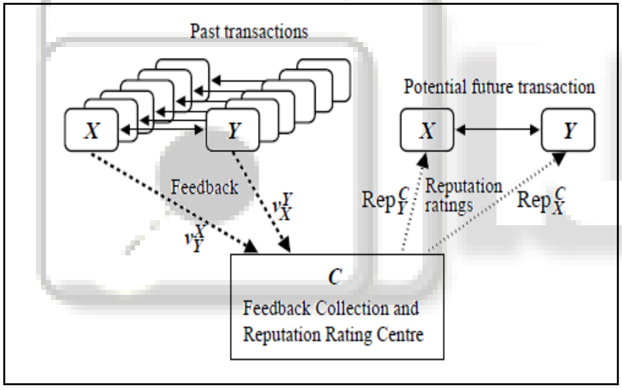

Fig. 3: Framework for collecting feedback and providing reputation ratings[1]

An engine calculate reputation score by the different feedback providers Propagation which lets the agent to get reputation values when required. There are two approaches for user reputation propagation. In the centralized approach reputation values are stored in a central server, and when there is a require, users forward their query to the central server for the reputation value.

Next Is The Algorithm Of The Beta Reputation System:

Step:1 The Reputation Function:

When dealing with the binary values the possible outcomes are {x,x }. starting step taking the integer value of past observations of x and x for estimate the possibility of x,to predict the expected relative frequency with what will happen in the future in simple words for prediction.

Step:2 The Reputation Rating

This step is more supreme for mathematical handling, and less supreme for reputation computation rating to human beings simple representation is required the notion of a probability value is opted E (ξ) reputation rating in the range [0,1] where 0.5 would be neutral rating.

Step:3 Combining Feedback

By gathering all the received parameters from the feedback provider the score is calculated.. Guess that 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:

1. $r_t^{x,y}\ = r_t^x + r_t^y$

2. $s_t^{x,y}\ \ = s_t^x\ + s_t^y$

T's combined reputation function by X and Y is,

$$\varphi(\rho, r_t^{x,y}\ \ s_t^{x,y}\ ) = T\varphi\ (\rho, r_t^x\ s_t^x\ ) \oplus T\varphi\ (\rho, r_t^y\ \ s_t^y\ )$$

Step:4 Discounting

Feedback value from high reputed agents conveys more weight compared to feedback from agents with lower reputation rating. So discount the feedback is function of the agent providing the feedback. a metric called view to describe about the truth of statements.

Step:5 Forgetting

The past feedback value is not for all time be applicable for the actual reputation rating, as the agent may modify it over time. The past feedback is give less weight than more current feedback. A forgetting factor which can be adjusted according to the expected rapidity of change in the observed entity.

B. *Lik Mui, Mojdeh Mohtashemi Et Al. A Computational Model Of Trust And Reputation [5]:*

The model which is given here is inspired by Ostrom's 1998 Presidential Speech to the American Political Society, which proposed a qualitative behavioral model for collective action.

The model description, agents and their environment are to be defined. Consider that agent a_j is evaluating a_i's reputation for being supportive. The all agents that a_j ask for this estimate can be considered to be a unique society of N agents A

Agents: A = {a1, a2, … aN}

The reputation of an agent a_j is relative to the particular fixed social network in which a_j is being evaluated.

It should be clear that reciprocity, trust and reputation are highly related concepts. The following relationships are expected:

- If a_i's reputation is increase in A so trust is also increase from the other agent.
- now, if a_j's trust is increase for for a_i so that type of probability is increase that a_j positively reciprocate to a_i's action
- Now, if a_i's reciprocity is increase to other agent in A than also increase a_i's reputation.

Decrease in any of the three variables should lead to the reverse effects.

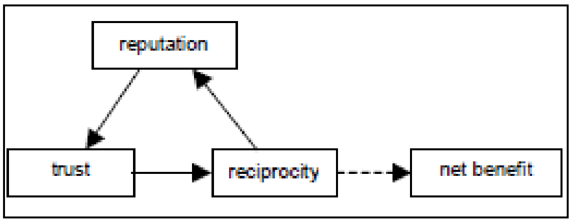

Fig. 4: Relationships between the three variables [2]

This model shows the relationships among trust, reputation and reciprocity. The direction of the arrow indicates the direction of influence among the variables. The dashed lineindicates a mechanism not discussed.

Reciprocity: mutual exchange of deeds
Reputation: perception that an agent creates through past actions about its intentions and norms
Trust: a subjective expectation an agent has about another's future behavior based on the history of their encounters.

C. Zhou et. al. The PowerTrust System Concept[6]:

The Power Trust system is motivated by the power-law which use the Bayesian method to create local trust scores where some power nodes are dynamically selected based on reputation by using a distributed ranking mechanism which is implemented by Distributed Hash Table (DHT) globally.

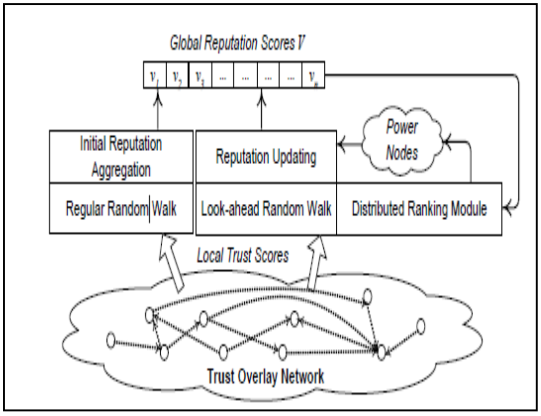

Fig. 5: Functional modules in the PowerTrust System and the control flow pattern in local trust score collection and global reputation aggregation[3]

A trust overlay network (TON) is built for all peers a P2P system. All peers evaluate each other, when a transaction takes place between a peer pair. All global scores form a stature vector, V = (v1, v2, v3, …..,vn), which is the output of the PowerTrust system. All global scores are the normalized. The regular random walk module is initial reputation aggregation. The look-ahead random walk is use to update the reputation score, periodically works with a circulated ranking module to recognize 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 decreasing list of feedback amounts.

Step 1: Selection of top-m peers (Power nodes)
global reputation stored among score managers which are input for every node i score manager j calculate, hash reputation value H(vi) using locality preserving hash and insert the (vi, i, j) to the node of H(vi) stored in the rising order of their reputation values in the DHT hash space due to the property of LPH. initialize node x = successor node of the maximum hash value

Step 2: Global Stature Aggregation
Local trust scores stored in the nodes which 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 short-term variable pre=0 is initialized; the error threshold ∈ 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 reputation V_jfrom the score manger of node j V_k= V_k+ V_j r_jk Compute δ = | V_k–pre| until δ<εoutput is Global stature for each node.

Step 3: Global Stature updating procedure:
The score managers collaborate with each other to locate the power nodes by step 1. If node x stores the triplet (i,vi, j) and find 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 every i & all node j collective local trust scores from node j send the score message (r_ij, i) to the score manager of node j for short-term variable pre=0; error threshold global reputation V_k of node k Initialize pre= V_k; V_k =0 For all accepted score pair (r_ik, j), where j is an in-degree neighbor of node k do receive node j global reputation 〖 V〗 _k from score manager of node j For node k be a power node, V_k=(1-α)Σ (vj×rjk) +α/m else V_k=(1-α)Σ 〖(V〗 _j×r_jk) δ = |V_k– pre| , until δ<ε Global reputation score for every nodes for use by score managers collaboratively to find the m most reputable nodes using is the output here.

D. Zhou Et Al Gossiptrust for Fast Reputation Aggregation [7]:

Gossiptrust deals with the speedy aggregation of global reputation scores. It deals with two steps the first one is local score aggregation and second one is global score distribution which are Performed. mathematically, for reputation calculation we require to calculate the weighted sum of all local scores s_ij score given by I for node j for every peer j= 1, 2, …,n , where the values of the feedback score normalized global scores and weights are applied

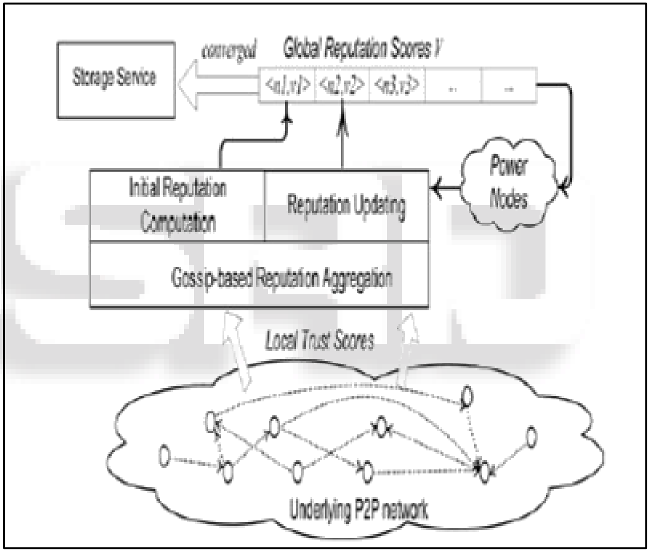
Storage Service
converged
Global Reputation Scores V
<n1,v1> <n2,v2> <n3,v3> .. ..
Initial Reputation Computation
Reputation Updating
Power Nodes
Gossip-based Reputation Aggregation
Local Trust Scores
Underlying P2P network

Fig. 6: Working of gossip group protocol[4]

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

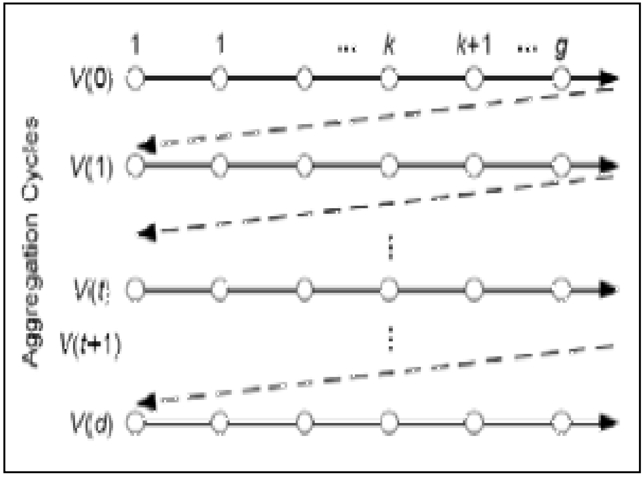

Fig. 7: Working of Gossip trust reputation aggregation cycle[4]

Step 1: Vector initialization

Initially the global reputation vector is V(0) .

Step 2: Recursive matrix vector calculation

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

Step 3: Exchange of global reputation

Vectors are exchanged from all node to other, which are joint with existing reputation vector, and the updated score is sent to a random node in the network

Step 4: Gossip aggregation of reputation

local score sij, global score vi(t-1) for i = 1,2,…,n and gossip threshold ε

X_i←S_ij×V_i (t-1) weighted score X_i is initialized

if (i == j), set wi ← 1, else wi = 0 consensus factor wi

k ← 0 k is gossip step

u ← xi/wi is previous score {(xr, wr)} is gossip pair sent to i in previous step

X_i ←Σ rX_r, W_i ←Σ rX_rUpdate the score and weight updated score is sent to a random node in the network (1/2 X_i, 1/2 W_i) to node it and itself k ← k+1 Next gossip step until |xi/wi – u| ≤ ε vj(t) ← xi/wi

Step 5: Storage of global reputation

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

E. Gupta Et Al. Debitcredit Reputation Computation[8]:

This reputation system is for p2p network to dependably calculate reputation score as a base for an motivation system and it suits 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 M Bytes data depending on growing the reputation score. Bandwidth factor b, b ∈ real, identify nodes for bandwidth Time factor in hours t, t ∈ integer.Period for the peer cooperation by sharing and staying online is satisfied The reputation is computed by the agent called reputation computation agent to sometimes update to the feedback providing agent's reputation, and to ensure that feedback value provide by them is kept locally so that it can be retrieve quickly. Reputation calculation agent does not play any role while finding and modifying so that it does become restricted access for the normal operation of the P2P system:

Step:1 Query-Response Credit (QRC)

Agents originally require to register then they get credit for providing their feedback to the system processing the query-response messages. key pair i.e. public and private key are generate on The registration: The agent choose to send these proof of m process to the RCA(Reputation computing agent) for receiving the credits.

Step:2 Upload Credit (UC)

Every 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, }. When the file download For downloading {requester identity, file_name, file size, time stamp} and encrypt it with its pivate 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.

Step:3 Download Debit (DD)

When downloading a file an agent requires to debit for downloading the file. For negative reputation score, the RCA retains the negative scores in the form of debit state with itself until those peers send some credits for processing.

Step:4 Sharing Credit (SC)

Registered agents are gets credit which will to be shared for staying online, based on the number of files they are sharing it can be achieved in two forms. First that it is deal 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 is that 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.

Step:5 Expiration and Consolidation of Reputation Scores

The time stamp is more 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 obtain one encrypted score back.

F. Kerschbaumet Al The Coercion-Free Stature System [9]:

A reputation system provides absolute privacy of the ratings, like, neither the ratee nor the reputation system will learn the value of the rating. Here We take both cryptographic as well as a non-cryptographic approaches.

Privacy of ratings may promote bad mouthing attacks in which an attacker leaves with intent bad feedback. We limit the possibility for this attack by giving a token system like, one can only leave feedback after a transaction, and provide a cryptographic proof of the privacy of our system.

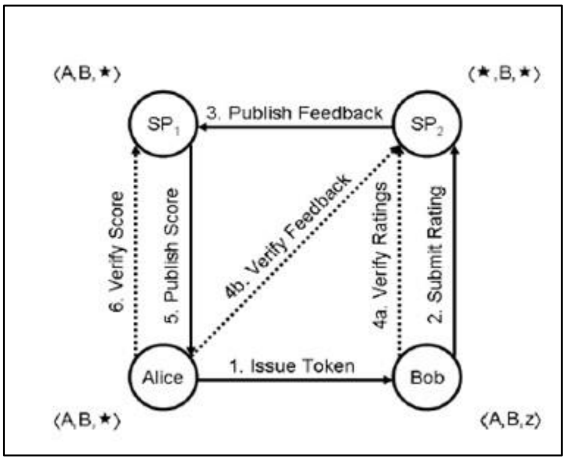

Fig. 8: Working of Centralized, Coercion-Free reputation System architecture[6]

An overview of this reputation system is depict in Figure above and the steps of this system is proceed as follows.

1) Alice (A) and Bob (B) two entities connect in a transaction. Alice issues Bob a token, that to provide feedback. Token should be issued earlier than the result of the transaction is known. else it should be refuse, if the result was negative and stop Bob from leaving negative

feedback. No transaction should be connect without having token for feedback first.

2) Bob leaves his feedback rating with SP2.
3) SP2 collects feedback from various raters and publish all feedback on a public bulletin board.
4) All the feedback providers validate the published feedback that no feedback for them is present for which they did not issue a token. All raters confirm in the published feedback that every rating is as they left it.
5) SP1 compute the aggregate reputation score for all ratee and publish it in the same bulletin board.
6) All the feedback providers validate that SP1 has compute their score and according to the its left feedback.

Assume that a single identity, e.g. throughout a public key infrastructure Denot by SX() a signature using the private key of party X. Parties can be rate as well as be rated. binary ratings $z \in \{0, 1\}$ in this section where 1 denote as a positive rating and 0 denote as a negative rating. There are two service providers SP1 which is first service provider for reputation and SP2 is the second reputation service provider and X is the set of all ratees and raters.

Step 1 Registration

An entity at first wantss to register arbitrarily choose the two secret keys $s \in Zp$ and $t \in Zp$. Then it send a public key gs to SP1 and gt to SP2. SP1 publish a record with every public keys gsX and their identities X or otherwise issues a certificate in the same manner SP2 does for gt

Step 2: Token problem

For a transaction between Alice and Bob Alice issue a token. Alice choose a arbitrary number $r \in Zp$ sends to Bob $\alpha = gr$, $\beta = grs$, $\gamma = grt$, Bob validate it $e(\alpha,gs) = e(\beta, g)$ and that $e(\alpha, gt) = e(\gamma, g)$. Alice stays a copy of r & list of the transaction. re-randomization is done to make token not nice for SP2 and based on Alice identifying any feedback forged by Bob.

Step 3: Feedback compliance

Bob provides his feedback z and encrypts by homomorphic encryption ESP1(z). chooses two random numbers l and m $\in Zp$. sends $\delta = gr,gl,ESP1(grl),grsAl$, $\rho = gm$, $\delta = grm$, $\varepsilon = grtm$, $\theta = ESP1 (z)$, SB(gr, gl,ESP1 (grl), grsl, gm, grm, grtm,ESP1(z)) to the second service provider SP2 and then SP2 validates with token generated.

Step 4: Feedback Publication

SP2 publish all the collected feedback values and publish $\eta = gr$, $\theta = gl$, $\theta = ESP1 (grl)$, $\kappa = grsl$, $\lambda = gm$, $\mu = grtm$, $\nu = ESP1 (z)$ Alice scans and checks whether $e(\eta, \theta)s = e(\kappa, g)$ true, Alice will conclude and it will be used for reputation calculation list r matching $gr = \eta$ and validates that $e(\lambda, gt)r = e(\mu, g)$ if it is fails, she claims that the feedback is forged by anyone and initiate with an investigation Bob could likewise scan every feedback and verify whether $e(\eta, v)t = e(\mu, g)$, but he performs an opposite verify by compare $\delta = \eta$ that his rating is unaltered $o = \theta = ESP1(z)$. If there is any verification is fails then he similarly claims a forged feedback.

Step 5: Reputation Score Calculation

SP1 decrypts $DSP1 (\iota) = \pi = grl$ and verifies $e(\eta, \theta)s = e(\pi, g)$ For each gsX verifies whether $e(\pi, gsX) = e(\kappa, g)$ If it is true then SP1 it should use this feedback & decrypt it $z = DSP1 (o)$, compute a reputation score then publishe that score along with Alice's identity. SP1 claims a forged feedback, if there is any related gsX and cannot use it in any score calculation. SP1 must create and publish a zero knowledge proof (ZKP) for the accurate calculation of the score from the ciphertexts o.

Step 6: Argument decision

If there is any party claims that any feedback has been forged, a trusted third party D is called upon. All party presents as proof the published feedback and the monitor D decides which party is at fault. If a party can show its purity the next party will be accuse. SP2's evidence is the signature SB(gr, gl,ESP1(grl), grsl, gm, grm, grtm,ESP1(z)) submitted by Bob. D validate the sameness of every entry in the signature with the published feedback and if each verification is succeed it and then accepts the proof. Bob's evidence is the signature SA(gr, grs, grt) received with the token. D validate that $gr = \eta$, $e(\theta, grs) = e(\kappa, g)$ and that $e(\lambda, grt) = e(\mu, g)$. If every verification succeed then it accept the evidence false claims of a forged feedback in will be erased.

Step 7:Send-off Self Feedback

This is a case for forged positive feedback, No one party alone can make a decision that Bob has left feedback for himself. By rising the service provider SP2's vision to contain the ratee, sendgrl in place of its ciphertext ESP1(grl), but there exist a more privacy-preserving solution. Bob publish gst and it can be validate by checking $e(gs, gt) = e(gst, g)$. Bob submits another value gr2lm with his feedback. SP2 then verifies it if e(grls, grmt) and e(gr2lm, gst) be different. without SP1 being capable to tie the feedback to gt either, which revealing grm would do. SP2 does so by selecting a arbitary number $n \in Zp$ and publishing grmn and gr2lmn next to with the feedback.

G. Hasan, Et Al Decentralized Privacy Preserving Reputation Protocol [10]:

A privacy preserving reputation protocol which protect the users by beating their individual feedback and revealing only the reputation score. All the source agents are based on at most k agents to preserve its privacy. On its own information of their trustworthiness in the context of preserving privacy and sends all of them an preservative shares of his private feedback value.

Step 1: Opening & choose Trustworthy Agents

This is done by querying an agent for computation of the reputation of a target agent. The source agent obtain the feedback providers in a context. All the agent can chooses up to k other agents with the possibilities that the selected agents will crack agent's privacy .

Step 2: Prepare Shares

At a time the source agent build the k other feedback providing agents the number one decides is stated as K. Agent prepares k + 1 share for covert feedback and the k shares are arbitrary numbers equally distributed over a large interval. But the last k+1 share (Fat-Σ individual feedback) mod M.M is publically identified overweight be feedback of a source agent a about a target agent t.

Step 3: Encrypt Shares

The record of all shares which are implemented by agents own public key therefore that only agent can open it and also each k th share which are encrypted by public key of the feedback agent so that only one can have access to its own share by any once private key

Step 4: Generate Zero-Knowledge Proofs

Agents computes for an agent the zp(zero knowledge proof) zp=(E(1) x…xE(k+1)) mod n2 public rsa modulus. The result 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 contact is not wanted and prove to a that the ciphertext has an encrypted value that stretch outs in that is the ciphertext hold feedback value within range. Non-interactive plaintext sameness zero-knowledge proofs. here the two ciphertexts, encrypted with the public key of feedback provider and other encrypted with the public key of complete record, enclose the same plaintext. Assure that agent a has structured the shares such that they insert up to a truthful feedback value and are trustworthy agents match to those correct shares.

Step 5: Send Encrypted Shares and Proofs

Every encrypted shares & zero-knowledge proofs which are sent simply for feedback providing relay on trusted agents

Step 6: Verify the Proofs

All agent calculate zp and validates the proofs which are received from an agent that shares are prepared correctly.

Step 7: Relay the Encrypted Shares

An agent basis to every agent a, the encrypted shares wich are received for it from trustworthy agents. Where, all encrypted share is which are joint togather, any agent who drop a message would be sense without learning any of the shares.

Step 8: Calculate Sum of the Shares

All agents receive the encrypted shares of trustworthy feedback providers. An agent calculate as the product of those encrypted shares with the ciphertext of its own k + 1th share by additive homomorphic property. An agent decrypts to get the plaintext sum and by adding the ka + 1'th share provides security

Step 9: Encrypt the Sum

An agent then encrypts the sum with k+1 from earlier step the sum of the shares suitably And calculate Reputation

Step 10: Generate Zero-Knowledge Proof

An agent create a non interactive plaintext sameness zero-knowledge proof, assures that the proof has the accurate sum of the shares.

Step 11: Send Encrypted Sum and Proof

An agent sends the encrypted sum and the zero-knowledge proof to query agent

Step 12: Verify the Proof

Query agent calculates a and checks the zero-knowledge proof received from all agent. which assure agent has computed

H. Androulaki Et Al. A Reputation System For Anonymous Networks[11]:

In this reputation system a peer agent which is represented by a pseudonym and cooperate with each other by removal pseudonyms like, their identity is not open to each other. These pseudonyms are not nice the individual and the peers they share the same reputation score. The values of the reputation to each peer sum up to make that peer's reputation value which are publically made available. anonymous credential systems, e-cash, and blind signatures. Reputation is switch in the form of e-coins called repcoins. The higher the amount of repcoins which is received from other users, the higher is the reputation of the user. A centralized entity bank, keep the three databases in which the first one is the repcoin quota database which gives repcoin one peer can provide to another the reputation database: amount of repcoin earned by other peers and the the old database to avoid for single time operation of the points

Step 1: Pseudonyms creation

All peer which create pseudonyms without registering with Bank. It just provide the random series for show 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 sequence. Digital signature which is used for sig and the pseudonym is for validation.

Step 2: RepCoin Withdrawal

Let B be the Bank. The U is peer and EC is the e cash. First message is from the user to bank, then bank checks and then respond to the user in accordance to strength. A wallet W of n repcoins has been withdrawn. Repcoins which are used to offer anonymity and single spending of the coins.

Step 3: Reputation Award

It Can be just stated reputation providing as two pseudonyms are there in this step, it is does not add actual identities rather than two pseudonyms are added as no direct contact but the pseudonym which are used so no knowledge of identities are revealed.

Step 4: Reputation Update

Take place when a peer needs to raise reputation having the repcoins received presenting itself to Bank And other peers as a pseudonym. But this cannot be easy as peer U wishes to deposit a received repcoin as pseudonym everyone is unaware except U the owner of PU. So the other peer could try to deposit the repcoin by to Bank as U. if peer's uniqueness kwon then anonymity is not preserved. So peer links Bank obtain blind permission been deposited, then deposits that blind permission.

Step 5: Reputation Demonstration

For demonstrating ones reputation to other both peers cooperate by using pseudonyms. For group G which is relay on positive reputation levels which are managed by Bank. For a peer to demonstrate reputation to peer validater V, the bank clutchs the group and registers in the group G. Peer links a Group and registers to the group by providing master public key the public key of group and a zero knowledge proof of knowledge that the master secret key which belong to it has been formed correctly and he is the owner. Group verifies that peer's reputation actually belong to that group or higher, and then access Grant for credential. Peer cooperate with the validater P under his pseudonym PU show by carry out and then Validate Credit having credential from group G. Specifically, PU shows that its owner has registered under a group of membership

I. Eleni Koutrouli Et Al. Reputation-Based Trust Systems For P2P Applications: Design Issues[12]:

In Peer-to-Peer (P2P) work out area trust issues have growing focus as a outcome of the decentralized nature of P2P systems.

- Policy-based trust systems, in which peers use credential validation to allow access control to limited resources.
- Reputation-based trust systems, in which use information allowing for earlier connections with an entity to build a reputation compute that will support a trust conclusion .

Conceptual Representation of Reputation-based Trust Systems:

The fundamental parameters of a reputation-based trust model are the following:

1. Trustee:

The entity that is provides a reputation value for a service it provides, e.g. the result of a transaction, or an attribute it posseses,.

2. Trustor:

The peer that wants to estimate the trustee´s reputation in order to build a trust decision about it, such as to decide whether to perform a transaction with it.

3. Third Party or Witness:

A peer that gives a advice for the trustee that relay on its own experiences with the latter.

4. Context:

The reputation of a peer which based on the exact context in which it applies, like a specific service the trustee provides, attributes of such a service, etc.

5. Recommendation:

Refers to the feedback given by peers about another peer's trustworthiness.

6. Trustworthiness or reputation:

An indicator of the quality of the trustee's services or attributes, based on recommendations, as well as the specific context and time.

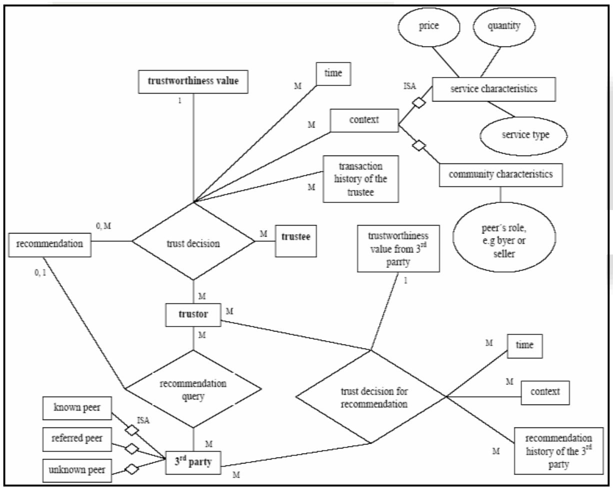

Fig. 9: Conceptual model of a reputation-based trust system[9]

1) Design Of P2P Reputation Systems:

In this system a reputation-based trust system assists peers in choosing a reliable peer to transact with.then it collects information on the transactional behavior of each peer. Transacting entities produce ratings about each other's performance, Each peer can store such information and can give it on demand or by broadcasting it in the network

Aggregates the trust information that worry the transactional behavior of the trustee and creates a trustworthiness (or reputation) value for it. Sometimes it is impossible or too costly to get ratings from all contacts with a given peer, a reputation score is based on a subset of ratings.

ranks peers according to their trustworthiness with a threshold in order to permit the trustor to prefer a peer to transact with and the system to take act against malicious peers while satisfying contributors.

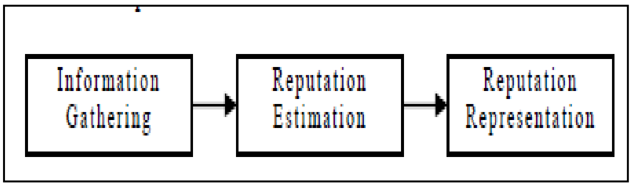

Fig. 10: Components of a P2P reputation-based trust system[9]

In information gathering it involves Trust information storage, dissemination and search mechanisms, Local control over trust information stored locally on a peer, Credibility of the recommender, Type of behavior taken into account, Context dependency.

In Reputation Estimation it involves Initialization of trust information, Scope of trust information (global vs. localized information), Trustworthiness estimation method, Transitivity extent, Recency dependency.

In Trustworthiness Representation it involves Range of trustworthiness values, Rank or threshold based, Distrust representation.

J. Ankita Thadani Et Al. Enhancing Privacy Preservation Of Stature System Through Homomorphic System[13]:

Stature can be stated as status i.e. What is one's status and by depandancy on that we do our activities for a person community or organization. There are variety of websites now a days used by us and the situation occurs when people transact with unidentified agents and catch decision for these agents for by considering the stature score. In this stature system homomorphic cryptosystem to preserve the privacy of an agent's message value and it can be used for evoting that is particularly suitable for the peer to peer network.

The one important word is homomorphic cryptosystem is used for this stature system which satisfies the homomorphic property and deals with the cipher text rather than plain text. Homomorphic cryptosystem the receiver works get the encrypted message.

1) The Classification By The Three Dimensions As Being Fundamental To Any Reputation System:

Formulation: a system may accept Positive and negative.

Calculation: mathematical calculation can be perform and it subdivides the types of communication.

Dissemination: the result of calculation it allow to system's users to obtain status.

2) Additive Homomorphic Cryptosystems:

In the homomorphism can be categorized into three categories like, additive,multiplicative and hybrid additive homomorphic cryptosystem. The additive homomorphic system is an asymmetric cryptosystem. Here we can compute E(x + y), by providing E(x), E(y), and PK.,Where x and y are the plain text Simply can be stated as E(2 + 2) = E(4).

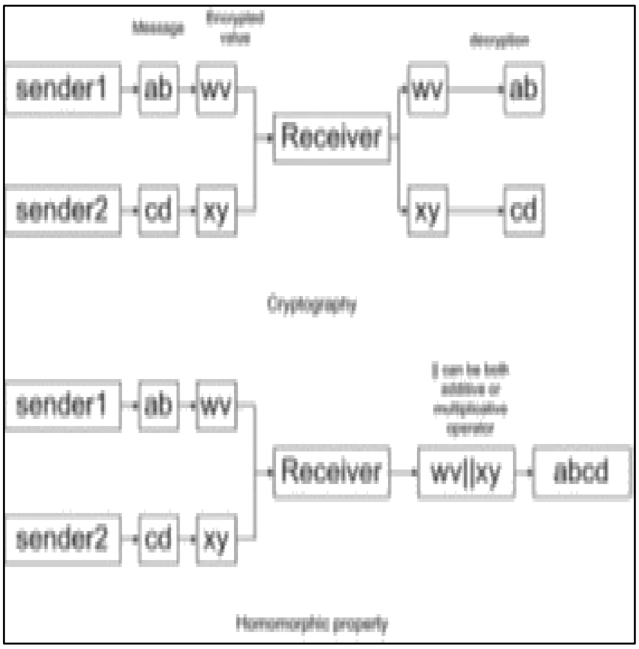

Fig. 11: Homomorphic property[13]

3) Praposed algorithm:

Let t be the target agent for the stature is going to be computed. Where in input K trusted other agents must be there selected on basis on some context ψ. the trusted agents can be {ta1, ta2,…,tak} those who gives there feedback value privately for the target agent {f1, f2…fn}and the output is the final stature score of the target agent. But considering the case where some agents can be malicious for the

Enhancing Privacy Preservation of Stature System network may deviate from the protocol. The protocol is to be decentralized and security is provided by homomorphism cryptosystem because it proves the randomized encryption. But it can only support the additive homomorphism but to increase the systems security even making the system work for fully homomorphic. Figure 11 gives the flow of the basic stature system.

Step 1: Initiation & Select Trustworthy Agents

By querying agent for computation of the reputation of a target agent. Source agent gets the feedback providers in a context .Each agent can selects up to k agents.

Step 2: Prepare Shares

At a time the source agent makes the k other feedback providing agents the number one decides is stated as K. 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-Σ individual feedback) mod M. M is publically known Fat be feedback of a source agent a about a target agent t.

Step 3: 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. Send Encrypted Shares and Proofs: All encrypted shares & zero-knowledge proofs are sent simply for feedback providing based on trusted agents.

Step 4: 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.

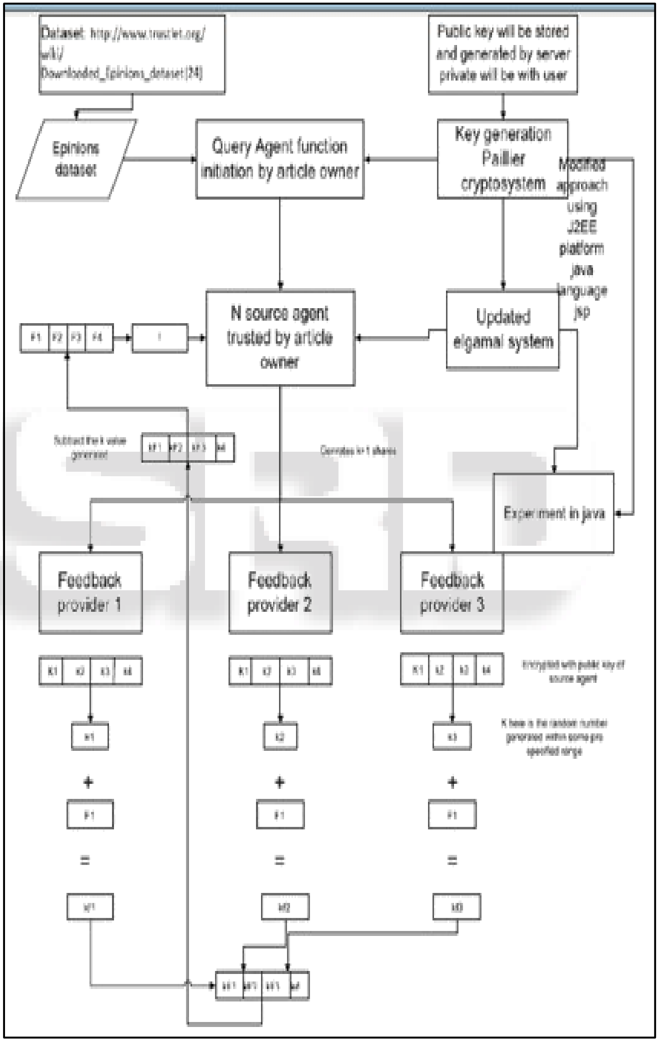
Dataset http://www.trustlet.org/wiki/Downloaded_Epinions_dataset[24]
Public key will be stored and generated by server private will be with user
Epinions dataset
Query Agent function initiation by article owner
Key generation Paillier cryptosystem
Modified approach using J2EE platform java language jsp
F1
F2
F3
F4
N source agent trusted by article owner
Updated elgamal system
Subtract the k value generated
kF1
kF2
kF3
k4
Generates k+1 shares
Experiment in java
Feedback provider 1
Feedback provider 2
Feedback provider 3
K1
k2
k3
k4
Encrypted with public key of source agent
k1
k2
k3
K here is the random number generated within some pre specified range
+
F1
=
kf1
kf2
kf3
kf1
kf2
kf3
k4

Fig. 12: Working of stature system[13]

Sr	System/	Architecture	Pros	Cons	Suitable for
No.	Protocol				
A.	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
B.	B. Lik Mui, Mojdeh Mohtashemi et al.[5] A	Decentralized	Make exlicit difference between trust and reputation	Reputation only apply for parallel network	Multi agent environment

	Computational Model of Trust and Reputation				
C.	Zhou et al [6] 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
D.	Zhou et al [7] 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
E	Gupta et al[8] 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
F	Kerschbaum et al [9]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
G	Hasan, et al [10] decentralized privacy preserving reputation protocol for the malicious adversarial	Decentralized	Zero knowledge transferred Secure ,robust	Can't prevent slandering	malicious adversarial, reputation systems
H	Androulaki et al. [11] A Reputation System for Anonymous Networks	Decentralized	represented by a pseudonym	bank, which is a centralized entity. no negative feedback	P2p malicious adversary
I	I.Eleni Koutrouli et al.[12] Reputation-Based Trust Systems for P2P Applications: Design Issues	Decentralized	Comperission & issues of the reputation based trust protocol supporting the right choices regarding these issues when designing a reputation system for a particular P2P application.	P2P applications that need to be addressed, such as handling of anonymity, supporting fault tolerance and scalability and various types of misbehavior and attacks that can affect a reputation system's reliability	P2p network
J	J.Ankita Thadani et al. [13] Enhancing Privacy Preservation of Stature System Through Homomorphic System	Decentralized	It provides more security, used to construct a threshold cryptosystem	Works only for the trusted agent	distributed network, ranking systems

Table 1: Comparision of Reputation Systems

III. Conclusion

This paper has surveyed the literatures on reputation models for verity of network. The centralized as well as decentralized different aggregation methods for diversity 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.

References

[1] Hasan, Omar, Elisa Bertino, and Lionel Brunie. "Efficient privacy preserving reputation protocols inspired by secure sum." Privacy Security and Trust (PST), 2010 Eighth Annual International Conference on. IEEE, 2010.

[2] Tajeddine, Ayman, et al. "PATROL: a comprehensive reputation-based trust model." International Journal of Internet Technology and Secured Transactions1.1-2 (2007): 108-131.

[3] 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.

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

[5] 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.

[6] 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.

[7] 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.

[8] 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.

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

[10] 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.

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

[12] Koutrouli, Eleni, and Aphrodite Tsalgatidou. "Reputation-based trust systems for P2P applications: design issues and comparison framework." Trust and privacy in digital business. Springer Berlin Heidelberg, 2006. 152-161.

[13] Thadani, Ankita, and Vinit Gupta. "Enhancing Privacy Preservation of Stature System Through Homomorphic System." Emerging Research in Computing, Information, Communication and Applications. Springer India, 2015. 439-449