SECURITY INTELLIGENCE for BROADCAST: THREAT ANALYTICS

Sumit Chakraborty
Fellow, Indian Institute of Management Calcutta, BEE (Jadavpur University), India
E-mail: surya20046@yahoo.co.in, schakraborty2010@hotmail.com; Phone: 91-9940433441

Abstract: Broadcast or multicast is one of the most fundamental concepts in data communication and distributed cryptography. A central entity wishes to broadcast a secret data stream to a dynamically changing privileged subset of recipients in such a way that non-members of the privileged class cannot learn the secret. This work presents an Adaptively Secure Broadcast Algorithm (ASBA) based on threats analytics and case based reasoning. It defines the security intelligence of an adaptively secure broadcast comprehensively with a novel concept. It recommends a set of intelligent model checking moves for the verification of security intelligence of broadcasting mechanism. The algorithm is analyzed from the perspectives of security intelligence, communication complexity, computational intelligence and efficiency of mechanism. The computational intelligence is associated with the complexity of broadcast scheduling, verification of security intelligence of broadcasting system, key management strategies and payment function computation. The cost of communication depends on number of agents and subgroups in the broadcasting group and complexity of data. The algorithm is applicable to the analysis of intelligent mechanisms in static and dynamic networks, auction or combinatorial auction for e-market, digital content distribution through computational advertising, cloud computing, radio and digital TV broadcast, SCADA and sensor networks.

Keywords: Broadcast, Algorithm, Security intelligence, Computational intelligence, Communication complexity

1. INTRODUCTION

Broadcast is one of the most fundamental concepts in distributed cryptography. A central broadcast entity wishes to broadcast a secret data stream to a dynamically changing privileged subset of the recipients in such a way that non-members of the privileged class cannot learn the secret. Here, the critical objective is to optimize the cost of communication, the computation effort involved in key construction and the number of keys associated with each recipient. A broadcasting system is vulnerable to various types of malicious attacks. An adaptively secure broadcasting system is expected to be a *resilient system*. The resiliency measures the ability to and the speed at which the system can return to normal performance level following a disruption [27]. The vulnerability of a broadcasting system to a disruptive event or threat should be viewed as a combination of likelihood of a disruption and its potential severity. It is essential to do two critical tasks: assess risks and mitigate the assessed risks [20]. To assess risks, the security intelligence of the broadcasting system should be explored: what can go wrong in a broadcasting mechanism? what is the probability of the disruption? how severe it will be? what are the consequences if the disruption occurs?

The security issues of a broadcasting system have been extensively studied in [3,4,5,6,7,8,9,10,11,19,32]. The simple solution of a broadcasting problem is to give each recipient its own key and transmit an individually encrypted message to each member of the privileged class. But, it requires a high cost of communication. Another simple solution is to give each possible subset of users with a key. In this scheme, each user has to store many keys. Ultimately, it requires a system which should be efficient in terms of cost of computation and communication. A broadcast encryption scheme allocates keys to the recipients for a subset of S of U, the center can broadcast messages to all users where all members of S have a common key. [5] introduces a parameter 'resiliency' that represents the number of users that have to collude so as to break the broadcasting security scheme. The scheme is considered broken if a recipient that does not belong to the privileged class can read the secret. A scheme is called k-resilient if it is resilient to any set of size k. A broadcast allows a sending agent to distribute secret data among a set of recipients such that each recipient gets the same value, even if the broadcaster is dishonest. Broadcast was introduced in [11]; it was shown that an adversary who can corrupt up to t players can be tolerated for perfectly secure broadcast if and only if $t < n/3$. A setup with digital signature was shown in [19]; it was stated that broadcast tolerating an arbitrary number of corrupted agents ($t < n$) is possible. The suggested protocols are polynomial in the number of players. A recent work in [18] argues that the communication model adopted by [3] is unrealistically pessimistic. The problem of adaptively secure broadcast in a synchronous model is possible for an arbitrary number of corruptions. The existing works have several gaps. The security intelligence of a broadcasting system has been defined weakly, incompletely and imprecisely. The broadcast algorithms lack intelligent model checking or system verification mechanisms based on rational threat analytics. The algorithms are not analyzed properly in terms of efficiency and complexity. The algorithms lack efficient communication protocol and strategic moves.

The contributions of the present work are as follows. This work presents an adaptively secure broadcast algorithm (ASBA) based on threats analytics and case based reasoning. It defines the security intelligence of an adaptively secure broadcast comprehensively. The algorithm explores various scenarios of agents, input, output, network topology including static and dynamic network, communication model, broadcast mechanism, security intelligence verification and revelation principle. It

recommends a set of intelligent model checking moves for the verification of security intelligence of the broadcasting mechanism. The algorithm is analyzed from the perspectives of security intelligence, communication complexity, computational intelligence and efficiency of mechanism. The algorithm is proposed based on case based reasoning, threat analytics and use of the concepts of information and network security. The logic of the algorithm is explored through case based reasoning of different critical applications such as wired and wireless communication network, internet, sensor network, mobile adhoc network, defense system, SCADA network, combinatorial auction or reverse auction in electronic market, radio, TV, digital content distribution and flocking [14,15,16,21,25,36]. The security intelligence is explored through threats analytics. The model checking algorithm assesses the risks of various malicious attacks and the relevant risk mitigation plans. The basic building blocks of the proposed algorithm are information and network security, distributed cryptography and secure multiparty computation.

The work is organized as follows. Section 1 starts with introduction, the definition of the problem of adaptively secure broadcast, research methodology and states contributions. It reviews existing literature and analyzes the gaps. Section 2 presents adaptively secure broadcast algorithm (ASBA). Section 3 analyzes the algorithm from the perspectives of security intelligence, computational and communication complexity and efficiency of mechanism. Section 4 concludes the work.

2. ADAPTIVELY SECURE BROADCAST ALGORITHM (ASBA)

Notations used in ASBA : S **-** Sending agent, R - Receiving agent, A - System administrator or regulator, C - Case, M - Move, T - Threat, V - Verification, P_d - Demand plan [d, b] where d is demand and b is the budget of a receiving agent, $D_{j;\ j=1,...,x}$ - Data stream in digital or analog signal (e.g. direction, speed, vision) to be broadcasted by the sending agent to the receiving agents, 1-n : one-to-many communication, m-n:many-to-many communication, p - combinatorial factors, P_b - Broadcast plan, p_f - payment function, t_r - maximum response time, n - number of requests meeting the deadline, T - sum of response time, r - revenue; t_d : time deadline, m - profit margin of S, FIFO - First-In-First-Out, LIFO - Last-In-First-Out, SI - security intelligence of the broadcasting system, QoS - Quality of Service, k_e - encryption key, k_d - decryption key.

****

Agents: C1. S, $R_{i;i\ =1,...,n}$, A; C2. S, $R_{i;i\ =1,...,n}$;

Network Topology: C1. Dynamic network; C2. Static network;

Communication model: C1.1-n; C2. m-n; C3. 1-n-p; C4. m-n-p;

Input: $D_{j;\ j=1,...,x}$;

Broadcast Mechanism:

1. $R \rightarrow S : P_d\ [d, b]$;
2. $R \leftrightarrow S : [P_b, p_f]$;
   objectives : [minimize t_r, minimize n, minimize T, maximize r] s.t. *constraints* : [t_d, b, m];
   moves : M2.1 FIFO, M2.2 LIFO, M2.3 priority queue, M2.4 load consolidation, M2.5 data filtering M2.6 unidirectional communication, M2.7 bidirectional communication, M2.8 synchronous communication, M2.9 asynchronous communication, M2.10 single round communication, M2.11 multiple rounds communication;
   payment function (p_f): C1. auction or combinatorial auction; C2: discriminatory price negotiation protocol; C3: swing option; C4: negotiate payment terms: C4.1 prepaid, C4.2 postpaid, C4.3 free service, C4.4 incentive;
3. S generates, refreshes adaptively and distributes keys to R for private broadcast : C3.1 encryption and decryption, C3.2 digital signature, C3.3 signcryption and unsigncryption, C3.4 randomization, C3.5: summarization / aggregation, C3.6 generalization, C3.7 suppression, C3.8 de-identification, C3.9 k-anonymity, C3.10 biometric or credential based access control;
4. $S \rightarrow R_{i;i\ =1,...,n}$: C3.1 broadcasts encrypted data $D' = \{ D_{;j=1,..k}\}_{ke}$; C3.2 broadcasts non-encrypted data D; C3.3 perception of signal by R from S without using any channel;
5. $R_{i;i\ =1,...,n}$: decrypts or unsigncrypts data. C4.1 $\{D'\}_{kd}$, C4.2 receives D; verifies security intelligence of broadcast.

Security Intelligence Verification:

1. Call threat analytics, assess risks of single or multiple attacks on broadcasting system : T1 false data injection attack, T2 sybil, T.3 node replication, T4 wormhole, T5 blackhole, T6 jellyfish, T7 rushing, T8 neighbor, T9 coremelt, T10 node deletion, T11 flaws in broadcast schedule, T12 poor QoS, T13 malicious business intelligence, T14 corruption in secret sharing, T15 information leakage through weak security algorithms;
2. do model checking of security intelligence : {authentication, authorization, correct identification, privacy: group, forward and backward, audit, fairness, correctness, transparency, trust and accountability};

V1. Audit if S is compromised: V1.1 R alerts A. V1.2 A alerts R. V1.3 S alerts R if A is compromised. Verify correct identity and authenticity of S.

V2. Audit if R is compromised in secret sharing: V2.1 R_i alerts S and A. V2.2 S alerts A.

V3. Audit if the communication channel is compromised: V3.1 S alerts R, A. V3.2 R alerts S, A. V3.2 A alerts S, R. V3.4 verify web service oriented computing schema.

V4. Audit false data injection attack. R checks fairness, correctness, trust and quality of broadcast.

V5. Do traffic congestion analysis: V5.1 R complains to S, A. V5.2 S alerts A, R. V5.3 A alerts S and R.

V6. R checks flaws in broadcast schedule and alerts S and A. V6.1 delay in schedule, V6.2 error in scheduling logic, V6.3 exception handling error;

V7. Audit QoS : R alerts S and A. V7.1 Denial of Service (DoS), V7.2 non-repudiation, V7.3 network connectivity and internet speed, V7.4 data corruption, V7.5 noisy signal, V7.6 data loss, V7.7 data integrity;

V8. Audit malicious business intelligence by verifying transparency and accountability in payment mechanism. V8.1 violations in contract between S and R, V8.2. error in payment function computation, V8.3. error in channel and package configuration, V8.4. flaws in pricing algorithm;

V9. Verify access control policy during join, leave, merge, split and subgroup change to ensure group, forward and backward privacy;

V10. Verify e-passport or trusted explicit and implicit certification of nodes and do resource testing. Call challenge response protocol

for sensor node attestation verification; check whether a sensor node is tampered by an adversary, check the configuration and correct setting of each sensor node, detect whether malicious software is loaded into sensor nodes, verify the integrity of the code, perform secure code updates and ensure untampered execution of code.

3. The honest agents compute penalty function and charge the corrupted agents; mitigate risk of threats on the broadcasting system by applying regulatory compliance.

Output: P_b, Security intelligence.

****

3. ALGORITHMIC COMPLEXITY

This section analyzes ASBA algorithm from different perspectives. First, it explains the algorithm in terms of agents, inputs, output, network topology, communication model, broadcast mechanisms and security intelligence verification mechanism. Secondly, it analyzes the communication complexity, security intelligence, computational intelligence and efficiency of broadcast mechanism. The algorithm has two critical components: broadcast mechanism i.e. multicast communication protocol and security intelligence verification mechanism or model checking algorithm.

3.1 Communication Complexity

Theorem 1: The cost of communication is O(n) where n is number of agents involved in the broadcast. It also depends on strategic moves of broadcast communication.

A broadcasting system may adopt different types of communication models such as one-to-many or single sender multiple receivers (SSMR), many-to-one or multiple senders single receivers (MSSR) and many-to-many or multiple senders multiple receivers (MSMR) communication models. In a three party model a sending agent, multiple receiving agents and a system administrator are associated with the broadcasting system. In a bi-party model a sending agent and multiple receiving agents operate without the support of any administrator. The topology of the broadcast communication network may be static or dynamic. In a static network, the number of agents is constant and the topology is also fixed. In a dynamic network, the number of agents change with time internally through change of subgroups within a group or merge or split operations or externally through join and leave operations [15,32]. The topology is not fixed with time. The sending agent generally sends a data stream or a set of data to the receiving agents through a secure communication channel. Alternatively, the broadcast may not be a private communication. The communication signal may be digital or analog. The cost of communication is o(n) where n is the number of agents associated with the broadcasting system. It also depends on the intelligence of broadcast plan, number of communication rounds of a broadcast session, complexity of data stream and network congestion.

The next critical issue is broadcast mechanism or multicast communication protocol. The receiving agents exchange their demand plans to the sending agent. The agents jointly settle broadcast plan (P_b) and payment function (p_f) through collaborative planning, forecasting, negotiation and exception handling. The sending agent (S) selects a set of strategic moves for intelligent communication. S consolidates the communication load requested by the receiving agents. S selects an efficient scheduling logic for adaptively secure broadcast: FIFO, LIFO, priority queue and data filtering. The data stream is filtered and multicasted to different sub-groups within a broadcasting group. S may send data in a single round or multiple rounds in case of multi-party negotiation. The sending agent communicates with the receiving agents through unidirectional or bidirectional or synchronous or asynchronous mode. S tries to explore an intelligent broadcast plan by solving a single or multi-objective optimization problem minimizing maximum response time, number of requests meeting the deadline, the sum of response time and optimizing revenue subject to various constraints like time deadline and budget of the receiving agents and target profit margin of the broadcasting agent. In case of private broadcast, S encrypts or signcrypts or signs the broadcasted data with digital signature and sends the private data through a secure communication channel. S may also adopt privacy preserving data mining algorithms through privacy randomization, summarization or aggregation, generalization, suppression, de-identification or k-anonymity. The receiving agents decrypt or unsigncrypt the received data and verifies security intelligence of the broadcasting mechanism.

3.2 Computational Intelligence

Theorem 2 : The cost of computation is a function of the complexity and efficiency of security algorithms, key management strategies, broadcast scheduling algorithm, model checking algorithms, payment and penalty computation.

It is a combinatorial issue for ASBA. In a specific case, computational theory of perception is applicable for broadcast in flocking. The cost of broadcast scheduling algorithm depends on the complexity of optimization problem: single objective or multiple objectives function, number of constraints and scheduling logic [33,34]. The cost of payment function depends on the complexity of discriminatory pricing algorithm, package configuration and incentives. The cost of model checking algorithm is a function of the complexity of threat analytics, risk assessment and mitigation plans.

Security algorithm: The cost of security algorithms depends on the choice of one or more functions such as encryption, digital signature, signcryption, randomization, summarization or aggregation, generalization, suppression, de-identification and k-anonymity [12,22]. The efficiency of the proposed broadcast key management is evaluated in terms of key storage, encryption, decryption and communication overhead. The basic objective of adaptive key construction is to improve the efficiency of broadcast by reducing the cost of different overheads. Key storage overhead denotes how many keys should be stored by each recipient. Decryption overhead denotes the computation time required by a recipient to perform the recovery of the plaintext. Encryption overhead denotes the computation time the sender is supposed to invest in order to parse the given revocation instruction and sample the ciphertext that disables all users that are meant to be excluded from the transmission and produce the cipher text. Communication overhead refers to the actual length of

the ciphertexts. A broadcast encryption scheme BE is a set of algorithms - KeyGen, Signcrypt, Unsigncrypt and Keyupdate. The parameter of the scheme is n, the number of recipients and is associated with three sets K, M, C corresponding to the sets of keys, plaintexts and cipher texts respectively.

Key Gen : It is a probabilistic algorithm that on input 1^n, it produces (sk, uk_1, . . . , uk_n). The decryption key uk_i is assigned to the i^{th} recipient. It is a symmetric encryption scheme where sk is the signcryption key. The algorithm also produces a membership test for a language L that encodes all possible revocation instructions for the signcryption function.

Signcrypt : It is a probabilistic algorithm that on input $m \in M$, a string $\lambda \in L$ and sk, it outputs a ciphertext $c \in C$. $c \in$ Signcrypt(sk, m, λ). It indicates that c is derived according to the distribution of the encryptions of the plaintext m based on the revocation instruction λ.

Unsigncrypt : It is a deterministic algorithm that on input c derived from Signcrypt(sk, m, λ) and a user-key $uk_i \in K$ where (sk, uk_1, . . . , uk_n) $\leftarrow$ Key Gen(1^n), it either outputs m or fails.

Key Update : It is a set of protocols that update the signcryption and unsigncryption keys to preserve group, forward and backward privacy and key independence [10,32]. Group key privacy guarantees that it is computationally infeasible for a passive adversary to discover any group key. Forward privacy guarantees that a passive adversary who knows a contiguous subset of old keys cannot discover subsequent new keys. Backward privacy ensures that a passive adversary who knows a contiguous subset of group keys cannot discover preceding group keys. Key independence guarantees that a passive adversary who knows any proper subset of group keys cannot discover any other group key not included in the subset.

Secure communication is a critical issue of broadcasting system. The basic objective is to provide confidentiality, data integrity, authentication and non-repudiation in the communication of sensitive data. *Signcryption* can ensure efficient secure broadcast communication. In case of secure communication, cryptography ensures privacy and secrecy of sensitive data through encryption method. The sender (S) encrypts a message (m) with encryption key and sends the cipher text (c) to the receiver (R). R transforms c into m by decryption using secret decryption key. An adversary may get c but cannot derive any information. R should be able to check whether m is modified during transmission. R should be able to verify the origin of m. S should not be able to deny the communication of m. There are two types of key based algorithms: symmetric and public key. Symmetric key encryption scheme provides secure communication for a pair of communication partners; the sender and the receiver agree on a key k which should be kept secret. In most cases, the encryption and decryption keys are same. In case of asymmetric or public-key algorithms, the key used for encryption (public key) is different from the key used for decryption (private key). The decryption key cannot be calculated from the encryption key at least in any reasonable amount of time.

A ***digital signature*** is a cryptographic primitive by which a sender (S) can electronically sign a message and the receiver (R) can verify the signature electronically. S informs his public key to R and owns a private key. S signs a message with his private key. R uses the public key of S to prove that the message is signed by S. The digital signature can verify the authenticity of S as the sender of the message. A digital signature needs a public key system. A cryptosystem uses the private and public key of R. But, a digital signature uses the private and public key of S. A digital signature scheme consists of various attributes such as a plaintext message space, a signature space, a signing key space, an efficient key generation algorithm, an efficient signing algorithm and an efficient verification algorithm.

Traditional signature-then-encryption is a two step approach. At the sending end, the sender signs the message using a digital signature and then encrypts the message. The receiver decrypts the cipher text and verifies the signature. The cost for delivering a message is the sum of the cost of digital signature and the cost of encryption. Signcryption is a public key primitive that fulfills the functions of digital signature and public key encryption in a logically single step and the cost of delivering a signcrypted message is significantly less than the cost of signature-then-encryption approach [12,13]. A broadcasting system is vulnerable to insecure communication. The basic objective is that the system properly signcrypts all sensitive data. A pair of polynomial time algorithms (S,U) are involved in signcryption scheme where S is called signcryption algorithm and U is unsigncryption algorithm. The algorithm S signcrypts a message m and outputs a signcrypted text c. The algorithm U unsigncrypts c and recovers the message unambiguously. (S,U) fulfill simultaneously the properties of a secure encryption scheme and a digital signature scheme in terms of confidentiality, unforgeability and nonrepudiation.

Theorem 3: A broadcasting system adopts adaptive key refreshment protocols to preserve group, forward and backward privacy for join, leave, subgroup change, merge and split.

Adaptive key refreshment management is associated with various types of events of a broadcasting system such as join, leave, split, merge and change of subgroup of the recipients [15]. When a recipient wants to join the broadcasting group, the group controller authenticates the new member by distributing a group key, a subgroup key and an individual key. Leave protocol is called when a recipient wants to leave permanently from the group. A recipient may change its subgroup and join a new subgroup leaving from the old subgroup. Merger protocol is called when several recipients merge together to form a new sub-group. Split protocol is called when several recipients want to break a merger and split.

Secure group communication requires efficient key management protocols [9,10,32]. To prevent the recipients who have already left from accessing future communications of a group, all keys along the path from the leaving point to the root node of the key tree are to be changed. In case of a change of subgroup within a group, only old subgroup key is replaced with a new subgroup key. It ensures forward privacy. To prevent a new recipient from accessing past communications, all keys along the path from the joining point to the root node of the key tree are changed. In case of a change of subgroup within a group, only old subgroup key is replaced with a new subgroup key. It ensures backward privacy.

There are three different approaches of key management: centralized, decentralized and distributed [10]. In case of centralized approach, a single entity acts as a group controller. But, the central controller is a single point of failure; the entire

group will be affected if there is a problem with the controller. In the decentralized approach, a set of subgroup controllers are used to manage change of membership of each subgroup locally. In case of distributed key management approach, there is no group controller. The group key can be either generated in a contributory way or generated by a member. All the members may participate in access control and generation of group key. The cost of computation and communication is a function of group size, number of subgroups, number of tiers in the key tree and number of keys to be stored by each recipient. The algorithm can adopt centralized key management approach for adaptively secure broadcast where the broadcasting agent is assigned the role of key management and refreshment dynamically.

3.3 Efficiency of Mechanism

Theorem 4: **The business intelligence of the broadcasting mechanism depends on the design of payment function.**
The payment function should be designed innovatively, fairly and rationally in terms of intelligent contract, pricing strategy, payment terms, incentives and penalty function. The payment function is negotiated through various ways such as auction, combinatorial auction, discriminatory price ladder, swing option, choice of payment terms and mode, price change and price protection strategies. Generally, the broadcasting entity and the recipients are supposed to act cooperatively. The broadcaster communicates the secret data to the recipients who decrypt the encrypted data, validate it and pay to the broadcaster. This is a fair and rational business scenario. But in case of malicious attack, one or more players may be corrupted and act non-cooperatively. They disclose the secret data or the decryption keys to the adversary. The corrupted agents may be the sender or recipients. In case of corruption, the corrupted agents receive the payment from the adversary. Alternatively, the broadcaster computes payment function dishonestly through flawed package configuration. It is essential to audit malicious business intelligence by verifying transparency and accountability of the payment mechanism from the perspectives of violation in contractual clauses among the agents, flaws in payment function computation or pricing algorithm, channel and package configuration. It is interesting to test the security intelligence of a broadcasting system through a fair and stable game between the adversary and the challenger.

3.4 Security Intelligence

Model checking is an automated technique for verifying a finite state concurrent system. Model checking has three steps: represent a system by automata, represent the property of a system by logic and design model checking algorithm. Model checking verifies specific set of properties of a system such as reachability, safety, liveness, deadlock freeness and fairness [28]. The basic objective of verification or model checking algorithm of ASBA is to ensure secure group communication of a broadcasting system. It provides one or more security services by detecting, preventing or recovering from one or more threats.

Theorem 5: The security intelligence of ASBA is explored through threat analytics.

ASBA defines security intelligence in terms of authentication, authorization, correct identification, privacy: group, forward and backward, audit, fairness, correctness, transparency, trust and accountability of the broadcasting system. The algorithm calls threat analytics: assesses risks of single or multiple threats on the broadcasting system such as false data injection attack, sybil, node replication, wormhole, blackhole, jellyfish, rushing, neighbor, coremelt, node deletion, flaws in broadcast schedule, poor QoS, malicious business intelligence, corruption in secret sharing and information leakage through weak security algorithms [23,24,29,30,31].
A malicious agent can exploit the configuration of a broadcasting system to launch false data injection attack against state estimation and introduce arbitrary errors into certain state variables. It is very common in today's broadcast from digital media (e.g. news, budget, voting results, got up game etc.). In an open environment, sensor nodes operate without any supervision; a malicious attacker can capture a node for reconfiguration or extract the private data stored in the node through cryptanalysis. An attacker may be able to deploy multiple physical nodes with same identity through cloning or node replication attack. An adversary may be able to deploy multiple identities of a node to affect the trust and reputation of a broadcasting system through Sybil attack. The attacker may be able to build an additional communication channel to capture private communication in sensor network through wormhole attack.
A key can be compromised either by physical extraction from a captured node or by breach in security protocol. The denial of service attack renders a node by overloading it with unnecessary operations and communication and may be able to make the whole distributed computing system inoperable. Coremelt attacks can target communication links blocking the exchange of useful information and results traffic congestion in broadcast network. Replay attack allows an attacker to record messages at one instance and replay it later at different locations. There are other possibilities of different types of attacks on multicast such as blackhole, jellyfish, neighbor and rushing attack. There are risks of snooping, phishing, cross site scripting, distributed denial of service, unauthenticated request forgery, authenticated request forgery, intranet request forgery and exploitation of distribution on web enabled broadcasting system such as digital TV [35]. The basic objective of the threat analytics is to assess risks of different types of malicious attacks and explore risk mitigation plans accordingly.

Model checking and system verification: The basic objective of digital defense is to verify the security intelligence of a broadcasting system and to protect the system from a set of threats and malicious attacks. An efficient algorithm is evaluated in terms of privacy, correctness, independence of inputs, guaranteed output delivery and fairness [17]. It ensures correctness if each party receives correct output. Corrupted parties select their inputs independently of the inputs of honest parties and honest parties must receive their output. Corrupted parties should not be able to prevent honest parties from receiving their output. Corrupted parties should receive their outputs if and only if the honest parties receive their outputs and this ensures fairness of the protocol. An algorithm preserves privacy if no agent learns anything more than its output; the only information that should be

disclosed about other agent's inputs is what can be derived from the output itself. Two models are commonly assumed: semi-honest model and malicious model. A semi-honest party follows the protocol properly with correct input. But after the execution of the protocol, it is free to use all its intermediate computations to compromise privacy. A malicious party does not need to follow the protocol properly with correct input; it can enter the protocol with an incorrect input. A third party may exist in a protocol. A trusted third party is given all data; it performs the computation and delivers the result. In ASBA, the agents are assumed to be semi-honest.

Theorem 6: **ASBA explores different scenarios of corruption in terms of broadcasting agent, system administrator, recipients, communication channel and broadcast data. It verifies security intelligence through efficient tracing and revocation mechanism.**

In the proposed algorithm, corruption may occur in various ways. The first scenario is related to corrupted sender and honest recipients; the sending agent is compromised by an adversary and broadcasts false data to the recipients; the corrupted sender gets payment from the adversary. The second scenario is associated with honest sender and corrupted recipients; the sending agent is an honest, rational and fair player and broadcasts correct message. But, several recipients are compromised by the adversary. It can be direct or indirect attack. In case of direct attack, the malicious agent gets the decryption keys from the corrupted recipients and intercept the secret message directly. In case of indirect attack, several corrupted recipients receive the secret message and disclose the same to the adversary. The third scenario is related to corrupted sender and corrupted recipients where both the sender and some recipients are compromised. The fourth scenario is associated with unsecured communication channel; the malicious adversary can capture the secret data directly from the communication channel though the sender and the recipients are not corrupted. The security issues of web enabled applications are discussed in details in [16,31]. Alternatively, the adversary may delay the flow of data by creating congestion in the communication network. In worst case, both the sender and the recipients are corrupted and the channel is unsecured.

Adversarial model **:** The adversary (A) is capable of corrupting a set of recipients so that A can access to the keys of the corrupted players. For a given plaintext-ciphertext pair (c,m), the adversary tries to distinguish if the pair is an actual plaintext ciphertext pair where m is sampled uniformly at random. It tests whether c is an encryption of m. If the adversary has no way of distinguishing a valid encryption key pair from an invalid one, then the encryption mechanism will be sufficiently strong to be used for efficient broadcast. The adversary tries to disable the revocation capability of the sender. It tries to know how an uncorrupted player responds to a decryption request.

Correctness **:** A broadcast encryption scheme is correct if for any $\lambda \in L$ that encodes a subset $R \subseteq [n]$ and for all $M = < m_1, \ldots, m_s > \in M^s$ and for any $u \in [n] \setminus R$, it holds that Probability [Unsigncrypt (Signcrypt (sk, M, λ), uk_u) $\in \{m_1, \ldots, m_s\}$] = 1 where (sk, $uk_1, \ldots, uk_n$) is distributed according to KeyGen(1^n).

Corruption [3,18]: A broadcast protocol allows a sender to distribute a secret through a point-to-point network to a set of recipients such that (i) all recipients get the same data even if the sender is corrupted and (ii) it is the sender's data if it is honest. Broadcast protocols satisfying these properties are known to exist if and only if $t < n/3$, where n denotes the total number of parties, and t denotes the maximal number of corruptions. When a setup allowing signatures is available to the parties, then such protocols exist even for $t < n$. For $t < n/3$ (i.e., when less than a third of the parties can be corrupted), fairness and guaranteed output delivery can be achieved for any function in a point-to-point network and without any setup assumptions. For $t < n/2$ (i.e., in the case of a guaranteed honest majority), fairness and guaranteed output delivery can be achieved for any function assuming that the parties have access to a broadcast channel. For $t \geq n/2$ (i.e., when the number of corrupted parties is not limited), security intelligence without fairness or guaranteed output delivery can be achieved assuming that the parties have access to a broadcast channel. ASBA results correct and fair output if all the agents, communication channel and data are free of corruption.

The corruption strategy indicates when and how parties are corrupted. In case of static corruption model, the adversary is given a fixed set of parties whom it controls. Honest parties remain honest throughout and corrupted parties remain corrupted. In case of adaptive corruption model, adaptive adversaries are given the capability of corrupting parties during the computation. The choice of who to corrupt, and when, can be arbitrarily decided by the adversary and may depend on its view of the execution. In the semi-honest adversarial model, corrupted parties follow security protocol correctly. However, the adversary obtains the internal state of all the corrupted parties and attempts to use this to learn information that should remain private. In case of malicious adversaries, the corrupted parties can arbitrarily deviate from the protocol specification, according to the adversary's instructions. The adversary is allowed to run in (probabilistic) polynomial-time. Alternatively, the algorithm is computationally unbounded i.e. the adversary has no computational limits whatsoever. ASBA is based on the assumption of adaptive corruption and malicious adversary model with polynomial-time algorithm.

Revocation mechanism [1,2,6,7,9]: The revocation scheme enables to revoke upto t players in the worst case where $t < n$. The revocation scheme is characterized by traitor tracing and self-enforcement property. The self enforcement property is obtained by giving each player a decryption key which contains its private sensitive data. So, a player is reluctant to disclose its key to the adversary. The key revocation scheme is a secret sharing scheme where the sender or broadcasting entity prepares a key in advance to be used after the revocation. In initialization phase, each recipient receives a share of this key. In the revocation phase, the sender broadcasts the share of the revoked agents. Each other recipient can combine this information with its own share and obtain the new key while even a coalition of all the revoked recipients does not have enough shares to compute any information about the new key. The central broadcasting entity plays the role of group controller and updates suitable keys (group/ subgroup / individual) to preserve group, forward, backward privacy and key independence.

A broadcasting mechanism is characterized by two important concepts: secret sharing and secure multi-party computation. In the classical problem of m-out-of-n sharing, a sender wishes to broadcast a secret to a group of n players such that any subset of

m or more players can reconstruct the secret, but less than m players cannot learn anything of the secret. In a secret sharing scheme, the players run a secure multiparty computation protocol on their shares to reconstruct the secret. The basic objective is to design a fair and stable protocol for adaptively secure broadcast. The protocol should enable distribution of secret data in a secured way. It restricts the players from disclosing their keys to the adversary. It traces the identity of the corrupted players whose keys were used to construct unauthorized decryption keys. It revokes the decryption keys of the corrupted agents privately and adaptively updates the relevant keys to preserve group, forward and backward privacy of the broadcasting system.

Theorem 7: A corrupted communication channel is a real threat to a web enabled broadcasting system.

The model checking algorithms must verify a set of critical parameters such as the risk of snooping and phishing, validation of SoC schema in terms of logic, main flow, sub flows and exception flows of the application, cross site scripting, injection flaws, malicious file injection by testing application programming interfaces and code, insecure direct object reference, cross site request forgery, information leakage and improper error handling, broken authentication and session management, insecure cryptographic storage and failure to restrict URL access [16,37].

Theorem 8: The recipients must verify the fairness, trust and correctness of broadcast data to detect false data injection attack.

False data injection attack broadcasts incomplete, corrupted, noisy, got-up and incorrect data through intrusion of malicious agents or corrupted sending agent. The receiving agents and the system administrator must verify the fairness, trust and correctness of broadcasted data in time. The risk mitigation plan requires true and honest feedback and complaints from the receiving agents about the data corruption in time. The system administrator must take strict actions against broadcast of false data through Right to Information (RTI) act. In a sensor network, the measurements are transmitted from the sensor nodes to the broadcasting system. A malicious agent can compromise the sensor nodes to inject errors. It is hard to detect bad measurements in real-time. The attacker may try to inject arbitrary errors in certain state variables. Real-time system monitoring is essential to detect false data injection attack and intrusion of malicious agents in the broadcasting system. Auditing is primarily required to validate the security policies and to review the observed behaviors of broadcasting applications, users and database. User profiling monitors and analyzes the activities of the users such as the sending and receiving agents. Data profiling analyzes the broadcasted data. In case of anomaly detection, the data of repetitive and usual behavior of the users is collected and suitably represented as normal profiles. The system administrator compares the profile and the activities of the current user with the normal profile. If there is a significant mismatch, it indicates an intrusion in the broadcasting system. It is useful for unknown attack. Misuse detection is useful for known attack.

Theorem 9: The broadcasting system should monitor traffic congestion in real time to avoid coremelt, blackhole, jellyfish, neighbor and rushing attack.

The malicious attackers send traffic between each other and not towards a victim host in coremelt attack. It is a powerful attack since there are $O(n^2)$ connections among n attackers which can cause significant congestion in core network. Broadcast networks often use web service to enable coordination among physical systems. The malicious attackers are able to flood the end hosts with unwanted traffic to interrupt the normal communication. This is a specific type of Denial-of-Service (DoS) attack where the network link to system server is congested with illegitimate traffic such that legitimate traffic experiences high loss and poor communication performance. Such a poor connectivity can damage critical infrastructure with cascading effect. There are three steps to launch a coremelt attack [30]. First, the attackers select a link in the communication network as the target link. Then, they identify what pairs of nodes can generate traffic that traverses the target link. Finally, they send traffic between the identified pairs to overload the target link. Thus, the attacker uses a collection of nodes sending data to each other to flood and disable a network link. To address such attacks, it is important to identify the source of excessive traffic and prioritize legitimate traffic. An efficient system should allow end hosts to identify long-running legitimate traffic. During heavy load, the router forward packets with proper priority and capabilities while dropping packets without capabilities.

A blackhole attacking agent tries to intercept data packets of the multicast session and then drops some or all data packets it receives instead of forwarding the same to the next node of the routing path and results very low packet delivery ratio. A jellyfish attacker intrudes into the multicast forwarding group and delays data packets unnecessarily and results high end-to-end delay and degrades the performance of real-time application. A neighborhood attacking agent forwards a packet without recording its ID in the packet resulting a disrupted route where two nodes believe that they are neighbors though actually they are not. Rushing attack exploits duplicate suppression mechanisms by forwarding route discovery packets very fast.

The broadcasting system requires an efficient network traffic monitoring system to avoid these attacks. A broadcaster seeks to minimize own delay of data communication and the malicious agents seek to maximize the average delay experienced by the rational players. Congestion is a critical issue in both wired and wireless communication channel. The broadcaster should monitor the congestion in communication channel in real time so that all the recipients receive the data stream in time without any loss of data or delay.

Theorem 10: Efficient tracing mechanisms are essential to detect sybil, node replication, node deletion and wormhole attack.

A broadcasting communication network is defined by a set of entities, a broadcast communication cloud and a set of pipes connecting the entities to the communication cloud. The entities can be partitioned into two subsets: correct and faulty. Each correct entity presents one legitimate identity to other entities of the distributed system. Each faulty entity presents one legitimate identity and one or more counterfeit identities to the other entities. Each identity is an informational abstract representation of an entity that persists across multiple communication events. The entities communicate with each other through messages. A

malicious agent may control multiple pseudonymous identities and can manipulate, disrupt or corrupt a distributed computing application that relies on redundancy. This is known as sybil attack [23]. Sybil attacks may affect fair resource allocation, routing mechanisms, voting, aggregation and storage of distributed data by injecting false data or suppressing critical data. A large-scale distributed system is highly vulnerable to sybil attack; it includes sensor and mobile ad hoc networks, p2p applications and multicast network.

It is really complex to trace the corrupted players in the broadcast. There are various types of tracing mechanisms against sybil attack: trusted explicit and implicit certification, robust authentication, resource testing and incentive based game [24]. In case of trusted certification, a centralized authority assigns a unique identity to each entity. The centralized authority verifies computing, storage and bandwidth capability of the entities associated with the broadcasting system on periodic basis. The recipients validate the received data from the sender and checks logically whether there is any inconsistency or chance of injection of false data in the decrypted message. Another approach of tracing is to adopt incentive based game wherein the objective of the detective is to compute the optimum possible reward that reveals the identity of maximum number of corrupted agents [24]. A local identity (l) accepts the identity (i) of an entity (e) if e presents i successfully to l. An entity may validate the identity of another identity through a trusted agency or other entities or by itself directly. In the absence of a trusted authority, an entity may directly validate the identities of other entities or it may accept identities vouched by other accepted entities. The system must ensure that distinct identities refer to distinct entities. An entity can validate the identity of other entities directly through the verification of communication, storage and computation capabilities. In case of indirect identity validation, an entity may validate a set of identities which have been verified by a sufficient count of other identities that it has already accepted.

A *wormhole* attacker records packets at one point in adhoc wireless communication network, tunnels the packets possibly selectively to another point and retransmits them there into the network. The attacker may not compromise any hosts and even if all communication protocols provide authenticity and confidentiality correctly. Packet leashes may be used for detecting and defending against wormhole attacks [16]. A leash is any information that is attached with a packet to restrict its maximum allowed transmission distance. A geographical leash ensures that the recipient of the packet is within a certain distance from the sending agent. A temporal leash ensures that the packet has an upper bound on its lifetime which restricts the maximum travel distance.

Sensor node attestation verification is a critical requirement of a smart broadcasting system [26]. It securely ensures whether a sensor node associated with the broadcasting network is tempered by a malicious attack. Each node should be attested with a valid digital test certificate. The verification algorithm must verify the identity and tampering status of each node. The basic objective of device attestation is that a malicious agent should not be able to configure or change correct setting of each node. A challenge response protocol is employed between a trusted external verifier and a sensor node. The external verifier sends a random challenge to the sensor node. A self checking verification function on sensor node computes a checksum over its own instructions and returns the result to the external verifier. If an adversary tampers with the verification function, either the computed checksum will be incorrect or there will be significant increase in computation time. If the external verifier receives the correct checksum within the expected time, it is concluded that the verification function code on the sensor node is unaltered. The verification function includes a cryptographic hashing function.

4. CONCLUSION

The basic objective of the proposed Adaptively Secure Broadcast Algorithm (ASBA) is to verify the security intelligence of a broadcasting system. This study can be extended in various ways. An important research agenda is to improve the resiliency of a broadcasting system against various types of malicious attacks using analytics. A broadcasting system is expected to be a resilient system. A vulnerability map can be modeled through a set of expected risk metrics, probability of disruptive event and the magnitude of consequences. It is also important to design a broadcast performance scorecard (BPS) based on a set of performance metrics and rating scale (1-5; 1-very dissatisfied, 2 - dissatisfied, 3- neither satisfied nor dissatisfied or neutral, 4 - satisfied, 5 - very satisfied). The performance metrics may be based on different criteria of security intelligence, QoS, quality of broadcasted data, broadcast schedule and payment function.

It is interesting to explore the application of the algorithm for the design of intelligent broadcast mechanisms in wired and wireless communication network, web enabled e-mail services, sensor network, mobile adhoc network, radio, digital TV, SCADA network, combinatorial auction or reverse auction in electronic market, digital content distribution (e.g. software, e-music, e-film), defense and flocking. Adaptively secure broadcast is important for positive impact of digital advertising. Different types of experiments may be designed for different types of malicious attacks on a broadcasting system through simulation games. Innovative broadcasting systems should be designed based on smart service oriented computing, networking, data, application and security schema. It is an interesting research agenda to explore intelligent strategic moves for model checking and broadcast communication protocol of the proposed algorithm. The existing list may not be an exhaustive one. The knowledge should be extracted by interviewing network security experts and broadcast system administrators. Another critical agenda is to improve the cost of computation and communication in private broadcast. The business intelligence of the broadcasting mechanism may be explored through innovative payment function, penalty function and pricing algorithms based on algorithmic game theory and secure multi-party computation.

REFERENCES

1. A. Shamir. How to share a secret. Communications of the ACM, volume 22, 612 - 613,1979.
2. G.Kol and M.Naor. Cryptography and game theory: Designing protocols for exchanging Information. Proceedings from 5th Theory of Cryptography Conference (TCC), 2008.
3. M.Hirt and V.Zikas. Adaptively secure Broadcast, Eurocrypt'2010, LNCS 6110, 466 - 485, 2010.

4. L.M. Batten and X. Yi. Efficient broadcast key distribution with dynamic revocation. Security and Communication Networks, 1(4):351-362, 2008.
5. A. Fiat and M. Naor. Broadcast encryption. In Douglas R. Stinson, editor, CRYPTO, LNCS 773, 480 - 491, Springer, 1993.
6. M. Naor and B. Pinkas. Efficient trace and revoke schemes. In Y. Frankel, editor, Financial Cryptography, LNCS 1962, 1-20. Springer, 2000.
7. S. Panjwani. Tackling adaptive corruptions in multicast encryption protocols. In S. P. Vadhan, editor, TCC, LNCS 4392, 21- 40. Springer, 2007.
8. S. Pehlivanoglu. Encryption mechanisms for digital content distribution. Ph.D. thesis, University of Connecticut, 2009.
9. R. Canetti, J. A. Garay, G. Itkis, D. Micciancio, M. Naor and B. Pinkas. Multicast security: taxonomy and some efficient constructions. In INFOCOM, 708 -716, 1999.
10. S. Rafaeli and D. Hutchison, A survey of key management for secure group communication, ACM Computing Surveys, 35(3), 309 - 329, 2003.
11. L. Lamport, R. Shostak, and M. Pease. The Byzantine generals problem. ACM Transactions on Programming Languages and Systems, 4(3):382 - 401, 1982.
12. W.Mao, Modern Cryptography Theory & Practice, Pearson Education,2007.
13. Y.Zheng. Digital signcryption or how to achieve cost (signature & encryption) << cost (signature) + cost (encryption). LNCS 1318, Springer-Verlag.
14. S.Chakraborty, S.K.Sharma and A.K. Pal. Privacy-preserving 1-n-p negotiation protocol, Hawaii International Conference on System Sciences (HICSS-41), Hawaii, USA, 2008.
15. S.Chakraborty. A study of several privacy-preserving multi-party negotiation problems with applications to supply chain management. Doctoral dissertation (unpublished), Indian Institute of Management Calcutta, 2007.
16. S. Chakraborty, Digital defense : Verification of security intelligence. Technical report. 2012.
17. Y. Lindell: Secure Multi-party Protocols, LNCS 2815, 1-20, Springer-Verlag Berlin Heidelberg, 2003.
18. J.A. Garay, J.Katz, R.Kumarasen and H.Zhou. Adaptively secure broadcast revisited.
19. A.Perrig, R. Canetti, D. Song and J.D. Tygar, Efficient and secure source authentication for multicast.
20. T.J.Parenty. 2003. Digital defense what you should know about protecting your company's assets. Harvard Business School Press.
21. W.R.Dunn. 2003. Designing safety critical computer systems. IEEE Computer, 36(11), 40-46.
22. M.Gertz and S.Jajodia. 2008. Handbook of database security applications and trends.
23. J.Douceur. 2002. The sybil attack. Proceedings of Workshop on P2P systems (IPTPS).
24. Pal,A.K., Nath,D. and Chakraborty,S. 2010. A Discriminatory Rewarding Mechanism for Sybil Detection with Applications to Tor, WASET.
25. D.Choi, S.Lee, D.Won and S.Kim.2010. Efficient secure group communication for SCADA. IEEE Transactions on power delivery, volume 25, no. 2.
26. A.Seshadri, A.Perrig, L.van Doorn and P.Khosla.2004. SWATT: Software based attestation for embedded devices. Proceedings of IEEE Symposium on Security and Privacy, Oakland, California.
27. www.theresiliententerprise.com accessed on 15.11.2012
28. Berard, B., Bidoit,M., Finkel,A., Laroussinite, F., Petit, A., Petrucci, L., Schnoebelen, Ph. and Mckenzie,P. 2001. Systems and software verification. Springer.
29. Y.Liu, P.Ning and M.K.Reiter. 2009. False data injection attacks against state estimation in electric power grid. CCS'09, Chicago, Illinois, USA.
30. A.Studer and A.Perrig.2008. The Coremelt attack.
31. M.Shema. edited by A.Ely. 2010. Seven deadliest web application attacks. Elsevier.
32. C.K.Wong, M.Gouda and S.S.Lam. 2000. Secure group communications using key graph, IEEE/ACM Transactions on Networking, 18(1).
33. B.Kalyansundaram, K.Pruhs and M. Velauthapillai. 2000. Scheduling broadcasts in wireless networks. In European Symposium of Algorithms, LNCS1879, Springer Verlag, 290-301.
34. J.Kim and K.Chahwa. 2004. Scheduling broadcasts with deadlines. Theoretical Computer Science, volume 325(3): 479-488.
35. Y. Oren and A.D. Keromytis. 2014. From the Aether to the Ethernet - Attacking the Internet using Broadcast Digital Television. 23rd USENIX Security Symposium. August 20-22, USA.
36. A. Clemanti, A. Monti, F.Pasquale and R. Silvestri. 2009. Broadcasting in dynamic radio networks. Journal of Computer and System Sciences, 75(4), 213-230.
37. www.owasp.org accessed on 15.8.2012