Erasure Coding with Replication to Defend Against Malicious Attacks in DTN

Sameh Zakhary, Milena Radenkovic
School of Computer Science
University of Nottingham
Nottingham, NG8 1BB, UK
[szz, mvr]@cs.nott.ac.uk

Abstract—Privacy and security in delay-tolerant networks (DTNs) have been an active research topic in the recent years, especially, as people can be involved in these networks and use their mobile devices to forward each other’s messages. Such communications require forwarding algorithms that often include replication or context awareness. In this paper, we study the security impact on specific forwarding protocols in both simulated City Scenario and using real connectivity data traces. We propose a hybrid technique combining Erasure Coding and distributed replication to defend against packet dropping malicious attack. We show that replication-alone technique - that is typically expected to improve performance and robustness - is greatly affected by such simple attacks. We show that an attacker can cause up to 50% drop in the success ratio when compromising about 30% of nodes across various scenarios. We use mobile nodes with different speed, transmission range and processing capability, fixed infrastructure access points in our experiments. Results show that using Erasure Coding and message replication at intermediaries achieves up to 250% improvement in the message success ratio compared to using replication only.

Keywords: Security, Replication, Erasure Coding, Robustness, DTN

1. INTRODUCTION

Security in DTNs has been a focus of recent research due to the many challenges arising from its unique characteristics such as open nature, lack of infrastructure, lack of central management and reliance on opportunistic wireless communication between nodes. For example, the bundle protocol [1] allows a node to transform the message delivery ownership to another node (called “Custody Transfer”). Moreover, security is a critical element in the deployment of many DTNs, as it has been shown in [2] that the impact of attacks on DTN is more devastating compared to the impact of the same attack against traditional networks such as the Internet.

Many forwarding protocols for handling message delivery in DTN have been proposed [3-8]. These protocols are responsible for controlling the message forwarding to maximize the probability of successful delivery - if ever possible, for example, Direct Delivery (DD), Epidemic [8], Spray and Wait/Focus [3, 4], Encounter-Based Routing (EBR) [5], Retiring Replicants (RR) [6], PROPHET [7]. Such protocols vary widely. On one end, there are protocols which flood the network with an unlimited number of message copies, such as [8]. On the other end are protocols which utilize a limited number of copies using replication techniques such as [3, 4]. Different techniques have been proposed to control/adapt these copies to achieve distinctive objectives such as optimize resource usage [7], minimize delay [5], minimize network congestion [6]. The choice always presents an interesting trade-off between successful data delivery and required node’s resources. Our focus is on recent proposed forwarding schemas that use a controlled number of message replicas to achieve a higher success ratio than DD and at the same time avoid overloading participating nodes in the DTN network. This class of protocols is referred to as Quota-based Protocols [5].

In this paper, we present analysis and evaluation of the effectiveness of selective dropping attack in DTN where two quota-based protocols are used. This attack is characterized by malicious nodes dropping traffic sent to/from specific node, or class of nodes. This attack is called Gray Hole (GH) attack. GH is hard to detect as it cannot be easily distinguished from the nodes’ unintentional dropping behavior, which might be caused by lack of resources, messages timeout or intermittent failure. Our novel contribution is in proposing a forwarding protocol that employs both Erasure Coding (EC) and distributed replication at Intermediaries to counter such attacks. Gray Hole attack is different from selfish behavior described in [9], where nodes are not willing to forward messages for other nodes (i.e. not acting as a router), and are only interested in their own benefit through forwarding messages which they send to other nodes (selfishness is the opposite of altruistic behavior). Replication-based forwarding protocols, as a sub-class of quota-based protocols, are mostly used to achieve a higher success ratio through allowing multiple copies of each message to exist in the network. We show that these protocols cannot provide adequate protection in the presence of a simple GH attack and propose a technique to minimize the impact of such attacks.

The rest of the paper is organized as follows. Section 2 presents the related work. Section 3 presents the proposed protocol which combines EC with distributed replication to counter the attack. Section 4 discusses the evaluation methodology, attack model and details of each scenario, while section 5 presents the simulation results. Section 6 concludes the paper and looks into future work.

2. RELATED WORK

The problem of security in DTN is becoming an active research topic [10] [11] [12] [13]. Current research aims to
address the wide requirements, some of which are critical requirements to the success of the DTN deployment, such as: Privacy, Authenticity, Integrity and Data Confidentiality.

Most of the proposed DTN security solutions relay on cryptography, similar to the Public Key Infrastructure [13], where every participating node owns a private key (Priv.k) and a corresponding public key (PK) which need to be available to some or all nodes so that they can encrypt data to each other. PKI relays on Trusted Third Party (TTP), so that nodes can retrieve the PK for a destination before they can encrypt any data to that destination. Since TTP availability cannot be guaranteed in DTN [12] due to long disconnections, [14] proposed a protocol that utilizes offline TTP. While Public-Private key Encryption-based solutions provide authentication for sensitive information at different hops in order to protect the network against malicious manipulation or ejection of data; such solutions face the problems of performance, power consumption, initialization and key management in the absence of infrastructure, as well as key revocation. The “Bundle Security Protocol” in [13] provided basic data integrity and confidentiality, but no anonymity of the sender and receiver. Hence, GH attack can be easily carried.

The work by Parris et al. [15] is concerned with another security aspect of DTN namely user privacy, where they proposed a technique to hide the social contacts of nodes. Rongxing et al. [10] proposed a privacy preserving protocol for VANET, where a number of predetermined trusted Road Side Units (RSUs) acted as a middle-man to help achieve source/destination anonymity. Both [15] and [10] address different challenges in Social Opportunistic networks and VANET respectively in specific classes of DTN network and do not focus on malicious behavior aimed at disrupting communication. [16] proposed using a watchdog technique in MANET to detect and isolate misbehaving nodes using EC, where intermediaries check passing messages in order to detect such behavior and inform other nodes. However, it is assumed in [16] that delays and disconnections are minimal, most nodes have end-to-end connectivity, and nodes can overhear the wireless medium. These assumptions are not applicable to DTN. Chuah et al. [17] proposed a protocol for networks employing network coding, where the sender generated more network-coded packets when the delivery performance has been degraded due to selective dropping attacks. Their proposed technique relies on feedback sent from receivers back to senders, and evaluation completed with syntactic mobility models with no real connectivity traces.

Similar work by Choo et al. [18] attempted to quantify the robustness of DTN in the absence of authentication. The focus of their work was to analyze the effectiveness and show the level of performance degradation in DTN under targeted flooding and acknowledgement counterfeiting attacks, but they did not propose a solution. Chen et al. [19] proposed to combine EC and simple replication at source in order to achieve lower delays in homogenous and small DTN scenario; but no malicious attacks were considered, and replication was only at the source. In [9], Hui et al. focused on understanding the effect of selfishness behavior in mobile social networks. Another work by Qinghua et al. [20] proposed incorporating the intermediate node willingness (or “social selfishness”) into the forwarding protocol. This work focused on analyzing the selfish behavior from a social perspective and/or ways to cope with such behavior. The malicious (Gray/Black Holes) were not studied, where nodes are not socially driven but maliciously and actively collecting interesting messages from the network to drop and disrupt communication. Moreover, GH’s impact goes beyond their social community where it is data rather than social centric.

Other research efforts have been aimed at increasing the robustness of DTN, through proactive detection of congested areas of the network. This included controlling and diverting traffic at intermediate nodes to prevent message dropping [21], or controlling the replication rate at each intermediary [6]. In [21], Radenkovic et al. presented a number of heuristics to offload the traffic from congested parts of the network and spread it over less-congested parts of the network while considering the social network of the nodes to increase the buffer availability. While Thompson et al. [6] proposed an algorithm for intermediary nodes to decide – with a locally calculated network congestion level - on the total number of replicas to produce at each hop. Both solutions [6, 21] assume that nodes are neither malicious nor greedy, and that nodes are correctly executing the proposed protocol.

3. Proposed Erasure Coding with Replication (ECR)

A. Erasure Coding Background

EC is a subtype of the generic network coding [22], where the message is split into many smaller blocks at the source. Each of these blocks is coded and sent out independently. The destination requires only a subset of these coded blocks in order to recover the original message. To better present EC, we use the following notations. We assume $S$ is the size of the original message. The replication factor is denoted with $r$, which reflects the level of data redundancy in the generated coded blocks. The encoded block size is $b$. The number of resulted coded blocks is calculated using (1).

$$\text{EC Coded Blocks (N)} = S \cdot \frac{r}{b}$$

The generated coded blocks are all equal-sized blocks, and the destination needs to receive $\left(1+\alpha \right) \frac{S}{b}$ coded block to recover the original message. Where $\alpha$ is a constant specific to the actual coding algorithm. The exact algorithm for EC is orthogonal to our study, and we refer the reader to [22] for further information. The important point to highlight is that using EC with replication factor $r$ (e.g. $r=2$), the destination requires $1/r$ of the created coded blocks to retrieve the message (e.g. $\frac{1}{2}$ the coded blocks). We also ignore $\alpha$ for simplicity, and $b$ (coded block size) is an implementation dependent as in [23].

B. The Proposed Idea

Our basic idea is to distribute the replication of the coded blocks at the intermediate nodes from source to destination using spraying techniques in [3, 4], rather than using the standard replication factor embedded in EC ($r$) [22] which is done at the block level at the source only. We compare this to the simple replication-only technique and show that combining EC (i.e. using $r$ closer to 1) with a very limited number of replicas at the intermediaries (about 3 initial replicas) leads to
much better performance compared to replication-only techniques with up to 13 initial message replicas and no EC.

Our proposed EC protocol is similar to [23] [24], in that we employ EC as a forward error correction (FEC) technique, but we consider n-hops forwarding and un-trusted intermediaries in City Scenario. Our focus in this paper is different as we aim to understand the applicability and usability of using EC in an un-trusted real world environment where an attacker is capable of compromising highly central nodes.

We propose that source nodes use EC to create coded message similar to [23]. Instead of just replicating at the source, these encoded messages are sent through the opportunistic network and handled independently by a forwarding heuristics at each intermediate node. The separation between the EC and forwarding ensures that our proposed protocol is dynamically reacting to the possible contact opportunities and adapting to the changing DTN topology. Intermediate nodes (Relays) ensure that adequate number of each encoded message is replicated in the network asynchronously for successful delivery.

In [23], authors showed that using EC and splitting the coded blocks on a number of relays \((K)\), could lead to an almost constant delay in a given DTN network. Each of these \(K\) relays must successfully deliver the message chunk to the destination, in order for the destination to be able to reassemble the original message. We show in our evaluation that this could lead to negative impact on the delivery ratio in a hostile environment and using a fixed number of relays \((K=5)\), as it also increases the probability that some of these \(K\) chunks gets dropped by a malicious node. We show that increasing the replication rate at the intermediaries in contrast to at the source as in [19] [23] in case of existence of malicious nodes has increased the success ratio.

4. EVALUATING ERASURE CODING WITH REPLIATION IN TWO HETEROGENEOUS APPLICATION AND CONNECTIVITY SCENARIOS

We evaluate the effectiveness of our proposed protocol “Erasure Coding with Replication” (ECR) which combines EC and replication, against dropping attacks in DTNs. We implement ECR as an extension to the Opportunistic Network Environment (ONE) simulator [25]. We use “Map-Based Movement” models. These models utilize map data in order to constrain node movement to the streets, roads and tram path according to the node type (Pedestrians, Cars or Trams). Moreover, we utilize ONE’s ability to use real connectivity data traces to evaluate our proposed protocol in a real-life conference environment.

The objective of our study in this paper is to measure the effectiveness of a selective drop attack in highly realistic city area. In order to achieve a realistic experiment, a number of elements needed to be configured and used in ONE. Firstly, we use a real city map (Helsinki City Map). Since there is no data available about actual network deployment to be used with this map, we analyzed the map using the map-based mobility models to determine the best location to deploy the access points and road side units. Secondly, we use a more realistic map-based movement mode (compared to Random Way Point), where different nodes have different constraints moving around the city as it will be detailed shortly. We are aware of some new movement models that have been proposed, which provide a closer step to simulating a real-life scenario [26], but these have not been used for the evaluation purpose in this paper as real connectivity trace was used instead.

A. In the Heterogeneity of the City Center:

Our City Scenario, which we use to evaluate our protocols, composes of a set of heterogeneous mobile nodes (Pedestrians, Cars, and Trams) that are laid on top of Helsinki map. The different nodes exhibit different characteristics. People carried mobile devices have smaller buffers compared to trams carrying mobile units. Our City scenario has 80 pedestrians, 40 cars and 6 trams and 10 access points/RSUs. All mobile elements of our scenario follow a movement model that attempts to closely match real world, where cars drive on roads or highways, trams have predefined and fixed route that goes across the city, and pedestrians walk on streets.

Trams in our scenario are acting as data carriers, i.e. they are not considered as sources or destination for any traffic. We assume that in a typical city scenario, these trams would have access at the different stations to an infrastructure to offload its messages if needed. Hence, the six trams in Helsinki scenario have access to the infrastructure at different asynchronous times of their journey, and form additional access points that can bridge disconnected areas around the city.

Figure 1 Location of 5 deployment points (each location has both an access point and a road side unit) around Helsinki City Center.

We assume that Infrastructure Interfaces provide the following services to the mobile devices, 1) Infrastructure devices are fixed and positioned at highly social intersections around the city center similar to [10]. Fixed infrastructure are either access points, which provide services to mobile devices carried by pedestrian; or RSUs, which provide services for moving vehicles. Other nodes acting as data mules (such as trams) are assumed to have access to infrastructure through being equipped with high-gain antennas. 2) All messages arriving at one of the infrastructure devices are readily available to any other infrastructure devices, as in most cases these devices are connected to the Internet.

We run many experiments to determine the best location to position the infrastructure devices around the most busy locations (that are crowded) over the Helsinki map. We have defined a set of Points of Interest, Points of Interest are various locations around the map where mobile nodes visit more often – i.e. with higher probability - than other areas around the city. For example, people go to the city center more than they visit
urban areas of the city. These points of interest represent areas around the city where the people visit more often such as workplace, business centers, market places, and similar locations.

1) Application scenario:
In our City Scenario, we simulate a planned attack – such as terrorism booming - that takes place at multiple different locations around busy areas in the city center. Potentially large number of people at various locations around the city center will attempt to communicate in ad-hoc mode with surrounding neighbors to get or report updates about the events currently taking place in their surrounding area. We also assume that the ad-hoc communication is the only available communication network between nodes, as cellular network would be destroyed or inaccessible.

We assume that the attacker would aim to disrupt the communication of any information to the police about some events. The attacker will aim to drop messages destined to a group of police cars but forward correctly any other messages. This makes the attack much harder to detect with watchdog monitoring techniques (such as [16]) that might be used by other nodes. This assumption is consistent with our application scenario where police cars are subscribing to witnesses’ live report of important scene information in case of a rescue or disaster situation.

2) Attack Model:
In this section, we explain the attack details and the specific configuration parameters used in our experiments. We assume that the attacker will compromise highly central pedestrian carried mobile nodes – as shown in Figure 7 (a) -. The number of nodes compromised is 40, which represents around 30% of the number of nodes in the city-center scenario. These nodes are pedestrian nodes which the attacker will pick based on monitoring the forwarding path of some messages exchanged around the city center. It is trivial for the attacker to identify which nodes are responsible for forwarding the greatest number of messages [27]. Hence, these nodes are set as prominent targets to be compromised to cause the greatest damage.

We assume a passive GH attack, in contrast to an active attack. Passive attack means that the attacker will exert no effort to divert the traffic to go through malicious nodes. Messages traversing through a GH node are silently dropped. This dropping behavior depends on the attacker’s objective; and it can be limited to dropping messages about a specific topic, or messages sent to a certain group (e.g. police cars), or a combination of these. In addition, this malicious behavior can be made time variant, in order to make it harder for any observing neighbor to identify the attacker malicious behavior. In this paper, we do not study the active malicious behavior by the compromised nodes in which these nodes would alter the forwarding heuristic (through lying in their exchanged meta-forwarding data) in order to cause the maximum damage possible.

We assume that the attacker would aim to disrupt the communication of any information to the police about some events. The attacker will aim to drop messages destined to a group of police cars but forward correctly any other messages. This makes the attack much harder to detect with watchdog monitoring techniques (such as [16]) that might be used by other nodes. This assumption is consistent with our application scenario where police cars are subscribing to witnesses’ live report of important scene information in case of a rescue or disaster situation.

3) Application Simulation Parameters
We validate our proposed protocol using an extensive set of experiments to study the impact of GH attack on the communication in the City scenario. Each point in the success ratio and latency figures is an average of 50 experiments.

Both mobile devices - carried by either pedestrians or cars - are assumed to be equipped with Bluetooth low range networking capability (range of 10 m). Trams have both Bluetooth as well as a high-gain wireless antenna with a range of 1000m and transmission speed of 8 Mbps.

<table>
<thead>
<tr>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simulation Time</td>
<td>43200 seconds (12 hours)</td>
</tr>
<tr>
<td>Publishers Pool</td>
<td>120</td>
</tr>
<tr>
<td>Subscribers Pool</td>
<td>40</td>
</tr>
<tr>
<td>Message Size</td>
<td>1 Megabyte</td>
</tr>
<tr>
<td>Publishing Rate</td>
<td>3 message/minute</td>
</tr>
<tr>
<td>Interface Type</td>
<td>Bluetooth</td>
</tr>
<tr>
<td>Transmit Speed</td>
<td>2 Mbps</td>
</tr>
<tr>
<td>Wireless Range</td>
<td>10 m</td>
</tr>
<tr>
<td>Total Number of Nodes</td>
<td>126</td>
</tr>
<tr>
<td>Pedestrian Nodes</td>
<td>80</td>
</tr>
<tr>
<td>Car Nodes</td>
<td>40</td>
</tr>
<tr>
<td>Trams</td>
<td>6</td>
</tr>
<tr>
<td>Infrastructure Devices</td>
<td>12</td>
</tr>
<tr>
<td>Map Size (W x H)</td>
<td>4500 m x 3400 m</td>
</tr>
<tr>
<td>Movement Model (Pedestrian,</td>
<td>Shortest Path Map-Based Movement</td>
</tr>
</tbody>
</table>
For our media sharing application, we assume that the publishers are generating content with rate equal to 1 MByte messages every 20 seconds (3 messages per minute). Message Time-To-Live (TTL) is assumed infinite as it is orthogonal to evaluating the attack effectiveness, and only impact the resources required for delivery by intermediate nodes. The results shown represent 95% confidence interval of the 50 experiments per graph point. For each experiment, we have set a random initial positioning using the “Shortest Path Map-Based” movement model, and ran a reference protocol (Epidemic [8]) to obtain a theoretical maximum number of forwarded messages that can be achieved for each of these runs.

B. Real dataset Evaluation:

In this paper, we validate our protocol by using real data traces (INFOCOM-2005 [28]). As part of the Haggle project, infocom05 data trace was collected, and it consists of a 4-day long trace in a conference environment.

1) Application Scenario:

Infocom05 data is real human mobility traces gathered during INFOCOM-2005 conference. A total of 41 iMote devices were handed to volunteers. Each of these devices is capable of scanning and recording surrounding Bluetooth-equipped devices. The volunteers carried these devices for four days around the conference venue and recorded their encounter with other Bluetooth devices both internal – other volunteers carrying a similar device - or external Bluetooth device which was not part of the experiment.

The 41 internal devices is divided into three unequal groups. First group consists of a number of malicious nodes that are selected as described in the following section explaining the attack model. The second group is a set of publishers, these nodes are selected from nodes that are less central (e.g. less socially popular) compared to the malicious nodes. The third group contains all remaining devices and is treated as a pool; subscribers were picked from this pool at random. We assume that neither publishers nor subscribers are malicious. Publishers are producing one video messages of size 1 MByte every 10 seconds.

External devices were included as intermediate relays only, where they can participate in forwarding the messages between the original 41 internal devices (i.e. they are neither selected as publishers nor subscribers). This is because encounters recorded for external devices were incomplete. These devices were not part of the experiment, and hence their encounters do not provide a complete view of the mobile network in such conference environment, but they could still be useful as data mules (i.e. to carry data between internal devices).

2) Attack Model:

In our experiments, we assume that attacker will compromise high average degree centrality nodes and convert them into malicious GH nodes. We assume that the attacker will compromise 6 nodes out of 41 devices. The attacker would follow a graph analysis similar to the analysis done for the City Scenario and shown in Figure 7 to determine which nodes to compromise. The details of the analysis are omitted due to space constraint. Since this real dataset involves humans, we set the attacker objective to intercept messages destined to any of the remaining 41 internal devices that are subscribers to this application and with a specific topic of interest to the attacker (such as “Police Report”). For any other topic, the intermediate malicious nodes are executing the forwarding protocol correctly and not maliciously dropping any messages.

3) Application Simulation Parameters

TABLE II shows the details of the experiment parameters used to replay the real traces and setup of our media sharing application in ONE.

5. RESULTS

In this section, we present evaluation of our proposed protocol. We first show and discuss the results in the City-Center scenario, then we validate these results by comparing them to results using real-life connectivity traces. We show the performance in terms of: “Success Ratio” (SR) which calculates the average number of unique messages delivered successfully to its destination. “Average Delay” (AD) showing the average end to end time spent from the time the message leaves the source until it reaches its destination, and the “Network Overhead” defined as the number of message transmitted. Results are reported for three different scenarios: 1) Protocols examined under no attack scenario to show a baseline performance, 2) Original protocols performance under GH attack, 3) Our proposed protocol under the GH attack. Each protocol is evaluated once with a “wait” phase and another time with a “focus” phase as described in [3, 4].

TABLE II summarizes the Protocol Keys used in the figures. The plotted performance metrics (Success Ratio, Average Delay) are normalized as a ratio using the Epidemic protocol metric as a base line for each run independently. For example, if SnF delivered only 150 messages at a specific run, while Epidemic forwarding can deliver 300 messages, then the normalized value for SnF at this point is 150/300= 50%. At each new run, the nodes are randomly positioned at different points around the city center, and then the simulation is started.

<table>
<thead>
<tr>
<th>Key</th>
<th>Protocol</th>
</tr>
</thead>
<tbody>
<tr>
<td>SnW</td>
<td>Spray and Wait</td>
</tr>
<tr>
<td>SnF</td>
<td>Spray and Focus</td>
</tr>
<tr>
<td>SnF+GH</td>
<td>Spray and Focus under GH attack</td>
</tr>
<tr>
<td>SnW+GH</td>
<td>Spray and Wait under GH attack</td>
</tr>
</tbody>
</table>
Figure 2 shows the percentage of messages delivered in scenario without attack for increasing number of initial message replicas in City Scenario. The GH attack is for 40 pedestrian malicious nodes. Comparing the two scenarios where EC is used (ECRF+GH and ECRW+GH), we see that replication cause a small difference as a very high success ratio is achieved with only two replicas, and almost the maximum with three replicas.

Figure 2 Map-Based - Success ratio for Spray and Wait, Spray and Focus and ECR protocols in three different scenarios.

Figure 2 shows that using EC with distributed replication, ECRW+GH performance is enhanced to more than 50% compared to SnW+GH, while ECRF+GH achieves two-folds increase compared to SnW+GH to reach 350%. SnF performing worse than SnW, and this is mostly due to the overhead in exchanging contact information, that exhaust the short encounter times in our scenario and shorten the interval which could be used to exchange useful application data. This same performance can be observed when the GH attack is introduced, affecting both protocols proportionally in the same way. Moreover, the figure shows that the success ratio is increasing as we increase the initial number of message replicas. From this empirical approach, we can determine the number of replicas to achieve maximum success ratio and resources. At the same time, it ensures the lowest latency possible as shown from the delay figure (Figure 3 ) on the respective curves.

Figure 3 Average delay (latency) of the three protocols with different message replicas.

Figure 3 shows how SnF has lower latency compared to SnW, and this continues to be true even when GH attack is introduced to both protocols (SnF and SnW). Since end-to-end delay is only measured for the first successfully delivered message to the destination, all subsequent message replicas to the destination are ignored; hence, comparing the delay metric of two protocol would reflect whether a protocol could discover/utilize shorter path to the destination with no regard to how many messages (unique or replicated) was delivered.

It is clear that the latency drops as we increase the initial number of message replicas. This is expected in a quota-based protocol, as the probability of the message being successfully delivered is positively correlated with the number of replicas this message had in the network, as it provides additional probabilities for the message to be forwarded through different and independent paths in the network graph. It is important to highlight how the latency increases when we introduce EC, as can be seen in ECRW+GH and ECRF+GH having a peak close to 250%. This is due to the source having to buffer messages waiting for additional new content to encode.

Figure 4 shows ECRW+GH outperforming other protocols using the composite metric as well. As each metric (i.e. delay, delivery ratio) differs, we want to be able to compare the protocols using multiple metrics in a combined form. As in [5], we construct a composite metric, we join both the success ratio and latency by multiplying the first by the inverse of the second (so as to penalize the protocol that has a higher latency and reward protocol that has a higher success ratio). This ensures that higher success ratio and lower latency would be given higher value on the composite scale.

Figure 5 shows the network overhead as we increase the initial number of message replicas. Each point is an average of 50 experiments, and we show the ratio of the transmitted message by a specific protocol relative to the number transmitted using epidemic forwarding – as it presents a theoretical maximum for network transmission cost -. SnF and SnW score the highest, while both EC protocols (ECRF and ECRW) are the lowest. This is due to optimizing the retransmission using EC, which leads to fewer messages being sent and a lower network overhead, as the destination could reconstruct messages more efficiently using the Forward Correction Code (FEC) feature of EC. This is interesting as it reflects the level of power preservation using the EC schema as nodes require fewer re-transmissions, but at the expense of slightly additional power consumption due to CPU encoding/decoding operations.

Figure 6 shows the distribution of the success ratio against all nodes in the case of SnF and SnW, and against GH nodes’ average centrality for (SnF+GH and SnW+GH) in each of the different 50 city-center Scenarios. As shown, the attacker compromises high centrality nodes where the malicious nodes’ centralities are not above the mean (around 35) but distributed around the mean. The attacker has still managed to cause considerable damage to the success ratio of a maximum of 50% in case of SnF.

In Figure 7 (a), the x-axis represents the measurement of degree centrality of the 50 different runs. The circle diameter is proportional to the number of nodes that have the centrality plotted on the y-axis. It shows that the publishers and subscribers were picked randomly so that their average degree centralities are spread away from the degree centrality average...
(Gray circle showing “All Nodes”), while the malicious nodes were closely clustered around the mean (of about 35) demonstrated by the smaller blue circle (labeled “GH”) compared to all nodes average degree centrality.

Figure 4  Composite metric to compare the different protocols.

Figure 7 (b) shows the contact duration between any two nodes using log scale on the x-axis. Both Figure 7 (a) and (b) show that the scenarios exhibit a high probability for short encounters and low probability of long encounters. This distribution of probability is closer but not identical to the characteristics of real-life traces that have been shown to exhibit Power-Law distribution (logging human carried mobile devices in real life) [28] [29]. Figure 7 (b) shows that about 80% of the contact durations are under nine seconds; hence, the scenarios are very challenging. As we have shown with SnF success ratio results, nodes exchange forwarding meta-data that represent encounter information and therefore lose few seconds of these short encounter durations. Furthermore, scarce energy is exhausted in data transmission [30].

Figure 7  a) Different group of nodes and their associated average degree centrality, b) Average contacts vs. the contact duration.

Figure 8 shows the probability distribution of the average contact duration between nodes in our experiments. Reaching more realistic mobility models has been the aim of active research recently [31]. Inter-contact times and contact durations have been used as the main metrics for characterizing mobility in DTNs especially when node density is low. Inter-contact time is a measure of the elapsed time interval between successive contacts between any two nodes. At that time, the two nodes are not in contact with each other and hence cannot exchange data. The contact duration measures the time a contact between two mobile nodes exists, i.e. the nodes are within each other communication range. Inter-contact times show how often two nodes have the opportunity (hence called opportunistic network) to exchange messages with each other for a maximum period equal to their contact duration.

Figure 8  Contact duration cumulative probability.

Figure 9 shows our evaluation of ECR protocol using real connectivity traces (INFOCOM05 [28]). Protocol ECRW+GH has the highest success ratio, while ECRF+GH comes next due to the overhead of exchanging contact history data required for the focus phase. We show that even though the GH attack has an impact on the success ratio, increasing the initial number of replicas has no positive impact on the success ratio but only increases the overhead (as shown in Figure 10). We relate this performance to the average number of intermediate hops the message traverse to reach destination. For INFOCOM05 dataset, the average number of hops between source and destination is one, i.e. the source is following a DD protocol. Using DD means that the source directly delivers messages to the destination without intermediate nodes’ help. This explains that increasing the number of replicas does not increase SR.

Figure 9  INFOCOM05-Success ratio comparison for the proposed protocol.
6. CONCLUSION

This paper presented an impact analysis of a simple GH attack (selective dropping) and how successful the attacker was in disturbing the communication in a DTN network in a city-center scenario using heterogeneous nodes (Pedestrian, Cars, Trams). We quantified the impact using the GH attack on two replication-based protocols (Spray and Wait, Spray and Focus) where the attacker compromised 30% of the central pedestrian nodes. We showed the significant impact on the success ratio which dropped to 50% compared to the simulation with no attack.

We then proposed a protocol that combines erasure coding and distributed replication. Evaluation and testing results of the proposed protocol have been presented using an extensive set of simulation and real connectivity traces. The simulation used a realistic map-based movement model in a city center where a peer-to-peer media sharing application was running using arbitrarily selected publishers/subscribers, and using heterogeneous node types (Pedestrian, Cars, Trams) at a disaster event. We showed a performance enhancement of up to 350% in some scenarios using our proposed erasure coding with replication protocol.

We plan to investigate the impact of variables affecting erasure coding. Furthermore, we plan to test with other coding techniques such as network coding. Extending our work to cooperatively detect and isolate malicious nodes. It is also interesting to test our proposed protocol against other active attacks such as Byzantine [32] and Warm Hole attack [33].

REFERENCES

[34] Qinghua, L., Z. Sencun, and C. Guohong. Routing in Socially Selfish Delay Tolerant Networks. in INFOCOM. 2010. IEEE.