An efficient weak secrecy scheme for network coding data dissemination in VANET

Invited Paper

Mario Gerla†, Roberto G. Cascella*, Zhen Cao†, Bruno Crispo* and Roberto Battiti*

†Computer Science Department, University of California Los Angeles, 3732F Boelter Hall, CA 90095 Los Angeles
*Dipartimento di Ingegneria e Scienza dell'Informazione, Università degli Studi di Trento, Via Sommarive 14, I-38100 Trento
Email: {gerla@, caozhen@nrl.}cs.ucla.edu, {cascella, crispo, battiti}@disi.unitn.it

***Abstract*—Vehicular networks create a new communication paradigm that enables to exploit the movement of cars to disseminate content. If network coding is used, vehicles have much more flexibility in content sharing and the system stability and scalability are promoted also in presence of mobility. Along this line, we propose an efficient mechanism to provide secrecy of the information. Traditional approaches based on encryption decrease the cooperation willingness of intermediate nodes, which have no expectation of recovering the file. Our scheme is based on obfuscation by processing and polluting the original file so that only the intended recipients, informed of corrupted blocks, can recover the information timely.**

We present several alternatives to efficiently provide weak secrecy and to foster cooperation. We simulate the file distribution in a vehicular network and show that the proposed scheme enhances content distribution in term of downloading speed and it is much more efficient than the ones that use encryption.

I. Introduction

The deployment of vehicular communication networks offers new opportunities for content and information sharing among cars. Vehicles can record and exchange information relevant to safety applications, e.g., notification of accidents or the presence of an immediate danger on the road, or exchange entertainment content, e.g., video sharing. Recent work in mobile peer-to-peer networks focuses on the possibility to exploit locality and the available bandwidth between vehicles without using a centralized server, or the infrastructure.

Within this peer-to-peer framework, a new opportunistic communication paradigm is derived by defining applications and network protocols in such a way that the peers joining the system can obtain data which is locally available. Nodes participate actively to the data dissemination process by relaying packets and generating new information so that the download time is reduced. The main idea is to apply network coding as transmission technique to increase the performance of the system and to reduce the coordination of the vehicles participating in this ad-hoc formed communication network [8].

Network coding is a technique to achieve the broadcast capacity [2] and its application to content distribution networks is proposed in [5]. In network coding a file F is divided in n blocks F_i of size l bits stored at the source. Each block consists of $m = \lceil l/q \rceil$ symbols defined as a vector over a finite field F_{2^q}, i.e., $F_i = [f_{i,1}, \ldots, f_{i,m}]$. Before each transmission, an intermediate node generates a new packet, which is the result of the linear combination of the blocks locally available. This node randomly draws out of the finite field F_{2^q} n coefficients which form the coefficient vector $C_i = [c_1, \ldots, c_n]$, with $c_i \in F_{2^q}$. Thus, a linear combination of blocks is $y = \sum_{i=1}^{n} c_i F_i$, which is the new encoded data transmitted by the intermediate node along with the coefficient vector. The potential number of the unique blocks that can be generated is only limited by the size of the finite field from which linear coefficients are picked, but larger is the field slower will be the encoding and decoding operations. When the destination node receives n linearly independent encoded blocks, the original file can be reconstructed.

In a system, that uses network coding to encode packets at the intermediate nodes, encryption at the source is the traditional solution to protect data from unwanted disclosure of sensitive information. Traditional approaches range from the adoption of public key cryptography or, more efficiently, symmetric key algorithms to encrypt the data, with the latter requiring the establishment of a shared secret between the source and the destination.

However, the benefits of using network coding to ensure timely dissemination of information, which is a strict requirement of real-time data, could be undermined if the relaying nodes do not store or process packets locally before the actual forwarding operations. If we can assume that the intermediate nodes are altruistic, then, the protocol functions properly. But this is not the case in many peer-to-peer applications, such as file sharing, as nodes participate actively only if they can benefit or they are interested in the shared content. This results in a paradox as the source wants to hide the information to unauthorized users but, at the same time, should give them incentives to cooperate.

In this paper we present a weak data secrecy scheme designed to work in a mobile environment, such as vehicular communication networks. We define weak secrecy as the capability of the source to secure the information for a bounded time as the content can then be revealed to other nodes after

Work partially supported by the project DAMASCO funded by the Italian Ministry of Research.

it is not considered sensitive anymore.

We target real-time data since the utility of this type of information is conditional to the transmission delay. Hence, in our approach we do not make extensive use of encryption because it increases the processing time at both the source and the destination, as well as, the complexity of the content dissemination process. Moreover several keys or group keys should be established to protect data addressed to different group of users. The approach we present in this paper is novel and has two features: 1) provides weak data secrecy and 2) limits selfish behaviour by fostering cooperation since different content is disseminated to different groups of nodes at once.

The rest of the paper is organized as follows. Section II discusses related works. Section III presents the system objective and Section IV details our approach to provide weak secrecy in a vehicular network that uses network coding as technique for content distribution. Section V evaluates the approach and Section VI concludes the paper.

II. Related works

Traditional approaches that provide data confidentiality consist of creating an end-to-end secure channel between the source and the destination which is used to forward information. The advantage of this solution is that it is independent from the technique for encoding and transmitting data.

A mechanism that does not require encryption is the *chaffing and winnowing* technique proposed by Rivest [12]. The idea consists of processing the information at the source to compute message authentication codes (MACs), which will be used by the destination to extract data, and of distributing more information than required in cleartext. The destination computes the MACs on the received data to verify which packets must be discarded. The security of the scheme is based on the difficulty of the attacker to identify the informative packets.

Further extensions to intentional data pollution consist of inserting a pre-processing step that implements an *all-or-nothing* encryption, which requires the decryption of the entire ciphertext before getting any information on a message block [11]. Jakobsson et al. define a new protocol, based on an all-or-nothing property, which scrambles the original message in such a way that it is sufficient for the source to encrypt only a small part to guarantee confidentiality to the whole data [7].

Stinson formalizes the concept of an *all-or-nothing* transformation and proves that this transformation is unconditionally secure if the output (all blocks except one) of this pre-processing stage do not give any information to reconstruct part of the input unless all blocks are used: the entropy of one block of the input does not change if more output blocks are available, while the conditional entropy of the input given the output is 0. In addition, Stinson proves that an invertible matrix defined in the finite field F_{2^q} is sufficient to implement a linear all-or-nothing transformation [13].

These properties have been exploited to define security mechanisms for content distribution networks which use forward correcting codes to ensure content reliability. In this case the pre-processing step and the generation of new encoded data are tight together so that only a small fraction of the generated codewords need to be encrypted to provide confidentiality [3].

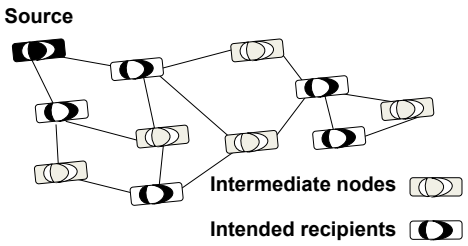

Fig. 1. Content distribution network in VANET

In the case of network coding, confidentiality without encryption can be achieved if multiple disjoint paths, at least two, are available between source and destination. The source disseminates partial information on each path and the intermediate nodes must have access to all these flows to recover the original file [9].

In vehicular applications, multiple disjoint paths are not always available as the links between vehicles are unstable and the paths can change dynamically. Moreover, full knowledge of the available links cannot be assumed as each vehicle has a limited view of the network learnt by its neighbours.

In this paper we define a new approach to guarantee weak secrecy based on an all-or-nothing transformation to pre-process the original content and we define a new incentive scheme based on the capability of intermediate nodes to contribute to the system objective if they are interested in the content itself.

III. Content distribution scenario and system objective

Our approach aims at securing for a bounded time the content distributed to a large set of vehicles authorized to access the data, as shown in Fig 1. The primary goal is to avoid other cars to reconstruct parts of the original file from few blocks while encouraging them to participate actively in the system objective so that the real-time content can be timely disseminated. Network coding is used to encode data at the source and at intermediate nodes, as it proves to be an effective technique to deploy vehicular applications also when cars move fast and the transmission channel is noisy [10].

Fig. 1 shows a source that splits the file in blocks and distributes linear combinations of the blocks to the neighbouring vehicles. The gray cars represent the intermediate nodes that have the role to encode the received packets before forwarding them to their respective neighbours, represented by the white cars in Fig. 1. Intermediate vehicles are also important to maintain the topology connected and to spread the information between vehicles that are not in the range of communication. The communication is limited to a single hop and neighbours learn of new available information by checking the encoding vector before actually downloading the block.

The complexity of our approach consists of ensuring that these intermediate nodes do not access the original file. In our

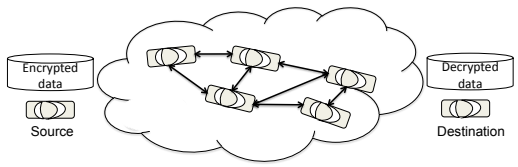

Fig. 2. The source transmits encrypted content that is decrypted once the destination recovers all the blocks.

setting, the source uses a secret group key to encrypt data so that multiple destinations, that have the same privileges, can decrypt the content. This shared key can be distributed to the members by using an out-of-band channel or at the registration phase if nodes subscribe to a service.

In traditional approaches, based on encryption to secure data, the source secures the file and then uses random network coding to distribute the encrypted blocks to the nodes, as shown in Fig. 2. Finally, the destination can decrypt the information when it successfully receives n linear independent combination of blocks, where n is the number of blocks.

In this setting, encryption preserves data secrecy but intermediate nodes will not like to collaborate as they have no control and access to the transmitted information. Thus, the benefit of network coding is reduced as vehicles have a smaller probability to receive useful information. To this extent we apply the "winnowing and chaffing" approach, defined in [12], to reduce the amount of information encrypted and to encourage groups of nodes to contribute resources. The incentive consists of their expectation to recover the data.

The basic idea is to pollute the content intentionally in such a way that intermediate nodes cannot recover timely the original file, which contains corrupted symbols. These symbols can be identified and discarded by authorized recipients as the source uses the secure channel to notify which portions of the file have been added to hide the original information.

Herein, we stress that this approach is valid to protect data that are considered sensitive for a limited amount of time as the intermediate nodes can operate on the whole file to guess those parts that have been corrupted. In fact, they can try to discard blocks and check whether the recovered information is meaningful. The time is a function of the size of the file and the number of polluted blocks, which is unknown to intermediate nodes. Starting from this assumption, we define different approaches to efficiently provide weak data secrecy in vehicular networks.

IV. Weak data secrecy

The techniques that will be used as building blocks in our weak data secrecy scheme are network coding to encode the information and Secure Random Checksums (SRCs) to identify polluted blocks. In this section we present first the role of SRCs and then our approach in detail.

A. Background: Secure Random Checksum

Gkantsidis et al. [6] define the Secure Random Checksum (SRC) as a lightweight alternative to prevent pollution attacks in content distribution systems implementing network coding. The construction of the SRC works as follows, the source generates m elements, as many as the symbols in a block, by using a secure pseudo random generator. These elements are defined in the same finite field F_{2^q} of the symbols of network coding. The SRC is computed as the sum of the result of the pairwise multiplication of the vector of random elements, i.e., $[t_1, \ldots, t_m]$, with the one of the block elements, i.e., $[f_{i,1}, \ldots, f_{i,m}]$. Then, the SRC of block F_i is $SRC_i = \sum_{j=1}^{m} t_j f_{i,j}$. The same process is repeated for all the blocks $F_1, F_2, \ldots, F_n$. The mask based checksum for the original file F is $(SRC_1, \ldots, SRC_n)$. This vector $\vec{SRC}$ is distributed with $\vec{t}$ over a secure channel to a destination node which can check if a block has been modified. Please note that different random elements t_i must be generated for every node to ensure that a malicious node could not generate a corrupted block that passes the checksum verification.

The verification of SRCs can be performed each time a node receives an encoded block $y = \sum_{i=1}^{n} c_i F_i$ by computing the $SRC(y) = \sum_{k=1}^{m} t_k y_k = \sum_{i=1}^{n} c_i SRC_i$. The SRCs are linear operations and can be computed very efficiently. In our implementation on a Intel Pentium 1.73GHz machine, the SRC generation rate is as fast as 96MB/s.

B. Weak secrecy scheme

The weak secrecy scheme uses SRCs to identify both the blocks corrupted by malicious nodes and the ones intentionally polluted by the source, as defined in [6]. The source distributes to authorized destinations valid SRCs for those blocks that contain the original content and modified values of the SRCs to identify the polluted content. When a block is received, a node computes the SRC and in case the verification fails, it uses a homomorphic hash function [6] to distinguish between encoded blocks polluted by the source or by intermediate nodes. In the second case, also the homomorphic hash verification fails and the block is discarded.

Intermediate nodes can still recover those blocks that have not been polluted. To strengthen the mechanisms by maintaining the same amount of overhead, the source pre-processes the data before obfuscation by applying an "all-or-nothing" transformation $\phi(x) = xM^{-1}$, where x is the data and M is an invertible matrix with non 0 entries in the finite field F_{2^q}, as proposed by Stinson in [13] and discussed in Section II. An important property of applying an all-or-nothing transformation consists of increasing the complexity for an intermediate node to recover the original file since all the corrupted blocks must be correctly identified before computing the inverse of the transformation.

In our scheme, the result of the transformation is distributed in cleartext using a network coding scheme, whereas the SRCs are encrypted to avoid that unauthorized users learn which blocks are corrupted. After the legitimate nodes receive enough encoded packets to recover the whole file, they can verify the SRCs and drop those blocks that do not match with their SRC values, as shown in Fig. 3. Then, the nodes invert the all-or-nothing transformation.

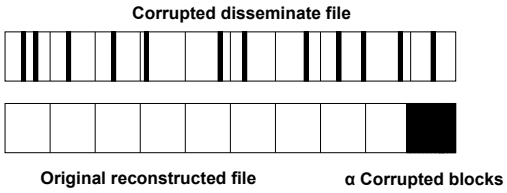

Fig. 3. File reconstruction: a legitimate destination discards the α corrupted blocks for the reconstruction of the file.

We define two approaches to efficiently provide weak secrecy by limiting the amount of corrupted data.

1) The same group as destination: In this first case, we assume that the source transmits multiple files to the same group of authorized nodes. Instead of obfuscating each file as defined above, the source computes the *bitwise exclusive or* of the files and pollutes the result. Thus, the source does not have to pollute each single file to guarantee weak secrecy but one file is sufficient to achieve the goal. Let's assume that the source has three files $F1$, $F2$ and $F3$ of the same length (padding is added otherwise) and the result of the operation is $C = F1 \oplus F2 \oplus F3$. Then, the source pollutes C to obtain C^I and transmits this file along with $C^I \oplus F2 \oplus F3$ and $C^I \oplus F1 \oplus F3$. The destinations can recover the three files by using SRCs computed on C^I to reconstruct the file C.

In its general form, we can compute the efficiency of this scheme by considering that each file is divided into n blocks. The source actually should transmit $(1+\alpha)n$ blocks to provide weak privacy via obfuscation, where $\alpha \in [0, 1]$ indicates the fraction of additional blocks required to pollute the original file. If we suppose that the source has m files to transmit, then the average number of polluted blocks for each file is $\frac{1+\alpha}{m}n$. This solution has one drawback as the source must have all the files available before starting the transmission.

2) Different groups as destination: This approach consists of a source and multiple destinations that belong to different groups. The source has m files, one for each group, that are "mixed" together, as shown in Fig. 4. This file is processed and then network coding is used to disseminate the same content to all groups. Nodes of each group winnow out useless blocks by using the SRCs, which are different for every single file, as the "garbage" part for one group is meaningful for another.

This approach further promotes the cooperation among different groups of nodes because they are not able to get their own content as efficiently as expected otherwise. In fact, if the source delivers the files for each group one by one, the intermediate nodes, which are not the intended recipients, may not cooperate. In this scheme cooperation among different group of nodes and the weak secrecy property are guaranteed if at least three disjoint sets of nodes form the groups. In fact nodes can filter out useless parts by the virtue of SRCs without being able to recover the content of any other group.

V. Evaluation

In this section we evaluate the efficiency of weak secrecy in a vehicular communication network through simulations

Fig. 4. Three different files are "mixed" to generate the data that the source transmits.

using NS-2. We have implemented an application layer agent to simulate the content distribution scenario described in Section III and we use the VanetMobiSim simulator [1], [4] to generate a realistic trace of mobile nodes (vehicles). We place 60 vehicles in an area of $2,400m \times 2,400m$ by using the Manhattan Grid consisting of 60 road segments of $300m$.

We simulate nodes coming from different directions that travel toward the center of the grid where they retrieve the content, and, then, move in other directions. The movement of vehicles is simulated to follow the *Intelligent Driver Model* [14] with the Intersection Management model working as follows: vehicles slow down and stop at each intersection.

In our simulations, we consider 3 distinct groups of mobile nodes, each of size of 20 vehicles, and a server that stores 3 different files, one per group. Each file is divided into 100 blocks of 1 KB and we implement network coding, defined in a F_{2^8} field, to distribute the file. Finally, we assume that cooperative nodes contribute a packet every 0.1 second.

A. Comparison of weak secrecy schemes

We evaluate the second approach described in Section IV and analyze the benefit of network coding in VANETs.

1) Files distributed separately.: The source disseminates the three files to each group, one after another, without using network coding. We assume that the vehicles of the groups, that have no interest in the file currently shared, do not participate actively. In our simulation, we let those uncooperative nodes contribute a packet every p $(p > 0.1)$ second, where 0.1 second is set to be the cooperative packet rate. In this approach, we do not use our weak secrecy scheme and confidentiality can be guaranteed only by encryption.

2) Weak secrecy to each file.: The three files are still distributed separately to each group but in this case we use network coding, to disseminate the data, and the weak secrecy scheme, to protect the information and to foster cooperation. We set the packet rate of the vehicles, that are not the intended recipients, to a value smaller than the cooperative rate. We let those uncooperative nodes contribute a packet every q $(p > q > 0.1)$ second, which is simulated to be grater than p as the weak privacy scheme retains nodes' cooperation.

3) Weak secrecy via mixing: The third approach consists of "mixing" the three files together before transmission. As discussed in Section IV, the polluted content for one group is meaningful for the others, so all the nodes will cooperate to distribute the files. Thus, we let all nodes in the simulation remain at full cooperation level, i.e., a packet is transmitted every 0.1 seconds.

In Fig. 5 we plot the three cases to show the efficiency of the weak secrecy scheme via mixing compared to the basic

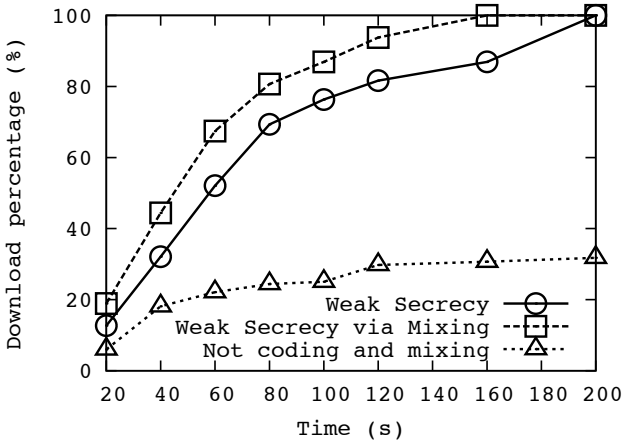

Fig. 5. File download rate versus time elapsed

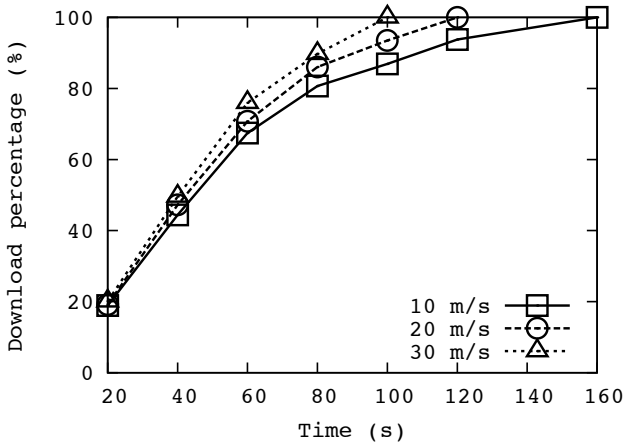

Fig. 6. File download rate versus simulation time with different node speed

implementation without mixing and without network coding. In our simulation, the node average mobility is set to $10m/s$, and we set $p = 0.5$ and $q = 0.3$ without loss of generality.

Fig. 5 clearly shows that our scheme efficiently increases the average download rate of vehicles, and in the meantime it guarantees weak secrecy protection of the data addressed to different groups. Moreover, network coding reduces drastically the download time in vehicular networks.

B. The impact of mobility

Next, we investigate the impact of mobility on the download rate by simulating the third scenario presented above. We increase the node mobility from $10m/s$ to $20m/s$ and $30m/s$ and we plot the results in Fig. 6. Fig. 6 shows that mobility helps cars to find more informative packets in the system, thus, the downloading rate is higher. In fact, the dissemination is performed only to single hop neighboring cars, thus, if greater the node speed is, then more vehicles are contacted on average.

Compared to a static scenario, mobility gives a higher probability to overhear from different cars new information, which might be more valuable. This is because the data are more likely to be encoded blocks that are independent of the current codewords stored on the node.

VI. Conclusion

This paper presents a novel and efficient weak secrecy scheme for network coding. It exploits network coding properties to disseminate content in vehicular communication networks and an all-or-nothing implementation of the "winnow and chaffing" approach to filter out corrupted blocks. We discuss two possible approaches to improve the efficiency of the scheme and show that even if intermediate nodes have the necessary information to compute the whole file, they are not able to immediately reconstruct the original information. Indeed, an all-or-nothing transformation is used to increase the complexity for intermediate nodes to recover the file.

We evaluate our scheme in a vehicular communication network. Simulation results show that our scheme based on mixing and obfuscation can help distribute the file more efficiently than using encryption. The scheme provides data secrecy and retains the cooperation level of intermediate nodes even if they are not the intended content consumers. This demonstrates that our scheme is a suitable solution for network coding based content distribution in vehicular networks.

References

[1] Vanetmobisim project homepage. http://vanet.eurecom.fr/.

[2] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung. "Network Information Flow". *IEEE Transactions on Information Theory*, 46(4):1204–1216, July 2000.

[3] J. Byers, J. Considine, G. Itkis, M. C. Cheng, and A. Yeung. Securing bulk content almost for free. *Computer Communications*, 29:280–290, 2006.

[4] M. Fiore, J. Harri, F. Filali, and C. Bonnet. Vehicular mobility simulation for vanets. In *Proc. of the 40th Annual Simulation Symposium (ANSS '07)*, pages 301–309, Norfolk, VA, USA, March 26-28 2007.

[5] C. Gkantsidis and P. Rodriguez. "Network Coding for Large Scale Content Distribution". In *IEEE INFOCOM*, Miami, FL, USA, March 2005.

[6] C. Gkantsidis and P. Rodriguez. Cooperative security for network coding file distribution. In *IEEE INFOCOM*, Barcelona, Spain, April 2006.

[7] M. Jakobsson, J. P. Stern, and M. Yung. Scramble all, encrypt small. In *Proc. of the 6th International Workshop on Fast Software Encryption (FSE '99:)*, volume 1636 of *LNCS*, pages 95–111, Rome, Italy, March 24-26 1999.

[8] U. Lee, J.-S. Park, J. Yeh, G. Pau, and M. Gerla. Code torrent: content distribution using network coding in vanet. In *Proc. of the 1st international workshop on Decentralized resource sharing in mobile computing and networking (MobiShare '06)*, pages 1–5, Los Angeles, CA, USA, 2006.

[9] L. Lima, M. Médard, and J. Barros. Random linear network coding: A free cipher? *CoRR*, abs/0705.1789, May 2007.

[10] J.-S. Park, U. Lee, S. Y. Oh, M. Gerla, and D. Lun. Emergency related video streaming in vanets using network coding. Technical report TR-070016, UCLA Computer Science Department, 2006.

[11] R. L. Rivest. All-or-nothing encryption and the package transform. In *4th International Workshop on Fast Software Encryption (FSE)*, volume 1267 of *LNCS*, pages 210–218, Haifa, Israel, January 20-22 1997.

[12] R. L. Rivest. Chaffing and winnowing: Confidentiality without encryption. *CryptoBytes (RSA Laboratories)*, 4(1):12–17, 1998.

[13] D. R. Stinson. Something about all or nothing (transforms). *Design, Codes and Cryptography*, 22(2):133–138, 2001.

[14] M. Treiber and D. Helbing. Explanation of observed features of self-organization in traffic flow. *arXiv*, Pre-print cond-mat/9901239, January 1999.