Short Message Multichannel Broadcast Encryption

José Luis Salazar, Jose Saldana, Julián Fernández-Navajas, José Ruiz-Mas, Guillermo Azuara

Departamento de Ingeniería y Comunicaciones. Universidad de Zaragoza
{jsalazar, jsaldana, navajas, jruiz, gazuara}@unizar.es

Abstract. The current use of short messages in wireless networks is highly growing. Messaging applications in mobile terminals with wireless coverage are very common in shopping, educational and transport centers, i.e. in centers of massive influx of people. This requires improving its efficiency, without losing security in such a hostile environment. In this paper, we propose an improvement in the use of the medium through a new multichannel broadcast encryption paradigm. Firstly, we rigorously demonstrate the security of our model that is characterized by two main issues: short messages and maintaining privacy in a shared frame. The improvements are obtained by reducing the transmitted overheads, saving bandwidth and airtime. To implement them, we improve the efficiency of communications, reducing the security headers to a single one, which will be shared by all receivers, while the payload is multiplexed via Chinese Remainder Theorem. In this way we reduce the packet length (less headers) and set the ratio of the encrypted text/plaintext equals to one, if we do not take into account padding and security headers. Although the model can be used by all types of networks, both wired and wireless, the improvement is more noticeable in the latter type. To make it remarkable, we quantify what this gain will consist of.

Keywords: Multichannel Broadcast Encryption, Provable Security, Channel Utilization Efficiency.

1 Introduction

The extended use of mobile services in the last decades has led to an increasing use of small packets [1], especially for certain services with tight interactivity constraints, e.g. VoIP or online games. In order to achieve this, small pieces of information (voice samples, game updates) have to be sent with a high frequency. Since each of these packets includes the headers imposed by the different protocol architecture layers, the inefficiency is stressed for these real-time services.

In this context, to find a good balance between the efficiency of communications and their security becomes a challenging issue, often influenced by the different service requirements. The problem becomes more relevant when physical constraints affect the Quality of Service (QoS), as it happens in wireless communications.

On the one hand, aA clear example of this inefficiency (useful data with respect to total transmitted bytes) can be found when sending small packets over 802.11: according to the definition of the MAC frame of the standard [2], the header plus the FCS is 40 octets long. Therefore, for a VoIP packet of 40 bytes, the efficiency is 50%. However, for big packets the problem is negligible because the payload may account for the vast majority of the frame size. In addition, the use of a shared medium requires some time for media access control mechanisms, which is another source of inefficiency. Nevertheless, there exist successful solutions aimed at addressing this problem, e.g. frame aggregation in 802.11 [3]. Other solutions have been proposed in the literature for wired scenarios [4, 5]. In addition, the use of broadcast packets is seen as a way to increase network efficiency, especially in wireless scenarios [6].

On the other hand, security has a negative influence on efficiency also: the addition of security headers can increase the overhead in a significant way. Therefore, approaches that jointly address aggregation and security become very important: if a secure header can cover a number of small packets, its overhead is amortized between them.

In the present article we propose to improve the balance of security and efficiency for fixing the problem. We jointly aggregate several small encrypted packets, and broadcast them, assuring that each of the sub-packets can only be decrypted by a unique entity: its legitimate receiver.

In any classic secure broadcast cable TV system (see Table 1, left column), a number of individualized headers u_i (i.e. the ones required for building the user private key for decryption), are usually sent together in a single multiplexed frame. In addition, a common field Hdr is required, and finally the encrypted content of each user (c_i) is appended.

However, our proposal (Table 1, right column) consists of merging all the individual headers, jointly with Hdr, to obtain a single header Hdr'. This can reduce the total amount of information to be transmitted, providing real savings in terms of bandwidth. At the same time, only the legitimate recipient *i* of each packet will be able to decrypt the information u_i, using Hdr' and their own private key.

Table 1. Comparison of encrypted information in classic models and our proposal.

Classic Model	Our proposal
(u_1, u_2, …, u_n, *Hdr*, c_1, c_2, … , c_n,)	(*Hdr'*, c_1, c_2, … , c_n,)

Previous models for broadcasting encryption have been proposed [7], but they are mainly designed for access control in encrypted file systems [8, 9]. In an abstract view, access control in an encrypted file system can be seen as a broadcast encryption problem, where the file system is the broadcast channel, and the key is broadcast (via the file header) to the subset of users that can access a concrete file. Different security models in cloud storage have been recently proposed in research literature [10, 11, 12].

Our proposal combines the underlying scheme of broadcast encryption, but using more efficiently the spectrum: we propose to aggregate the encrypted packets following

the way of the Chinese Remainder Theorem, and to implement random encryption where the random seed is shared by all users.

All in all, the specific contribution of the present paper is twofold:

- The proposal of a new security scheme able to merge a number of individual headers, resulting in a reduction of the total size of the information to be transmitted.
- To apply our proposal to wireless networking (802.11) in order to increase its efficiency by the reduction of the amount of information that has to be sent through the wireless medium. We also provide an analysis of the tradeoff that appears: security level (in terms of key size) vs efficiency (in terms of channel utilization, i.e. achieved rate divided by nominal rate).

In the next section (2) we describe the elements of a broadcast encryption system, together with the security model to be analyzed. In Section 3 we explain our proposal in two stages, for a better understanding: first, a basic description for a deterministic encryption and second, the subsequent introduction of randomness. Section 4 studies the balance of security and efficiency achieved, using IEEE 802.11. The paper ends (section 5) with the Conclusions.

2 Broadcast Encryption

2.1 Syntax

In this section we will use the model for a multi-channel broadcast encryption system proposed in [14], and will adapt it for our specific issues. Formally, such a system consists of four probabilistic algorithms:

- **Setup** (λ): Takes as an input the parameter security λ, generates the global parameters *param* of the system, and returns the encryption secret key EK.
- **Extract** (i, EK): Takes as an input the user's index i, together with the encryption key, and outputs the user's private keys (p_i, x_i).
- **Encrypt** (u_1, u_2, …, u_n, m_1, m_2, … , m_n, EK): Takes as an input the identifiers of n users (u_i), n messages (one per user, m_i) and the encryption key EK. It outputs (Hdr, ET) where Hdr is a random number for encryption/decryption, and ET is the encrypted text, computed with all plaintexts.
- **Decrypt** (Hdr, ET, i, p_i, x_i): Takes as an input the random header Hdr, the encrypted text ET, the user's index i, and their private keys p_i and x_i, and it outputs the cleartext for i, m_i.

For correctness, we require that for all $i \in \{1,\ldots, n\}$, if EK $\leftarrow$ **Setup** (λ), (p_i, x_i) $\leftarrow$ **Extract** (i, EK) and (Hdr, ET) $\leftarrow$ **Encrypt** (u_1, u_2, …, u_n, m_1, m_2, … , m_n, EK), then $m_i \leftarrow$ **Decrypt** (Hdr, ET, i, p_i, x_i).

2.2 Security Model

We define the security model of a broadcast encryption system, adapted from [14], with the following game between an *adversary* $\mathcal{A}$ and a *challenger*:

- **Setup** (λ): The *challenger* runs the setup algorithm for generating the global parameters *param* of the system, and returns the secret encryption key EK. Corruption and decryption lists Λ_C, Λ_D are set to empty lists.
- **Query phase 1:** The *adversary* $\mathcal{A}$ adaptively asks queries:
  - Corruption query for the *i*-th user: the *challenger* runs **Extract** (*i*, EK) and forwards the resulting private keys (p_i, x_i) to $\mathcal{A}$. The user u_i is appended to the corruption list Λ_C.
  - Decryption query on (Hdr, *ET*, *i*). The *challenger* answers with **Decrypt** (Hdr, *ET*, *i*, p_i, x_i). (Hdr, *ET*, *i*) is appended to the decryption list Λ_D.
  - Encryption query for the target users. The *challenger* answers with **Encrypt** (u_1, u_2, …, u_n, m_1, m_2, … , m_n, EK).
- **Challenge:** The *adversary* $\mathcal{A}$ defines an index *j*, which specifies the attacked user, u_j. The challenger runs **Encrypt** (u_1, u_2, …, u_n, m_1, m_2, … , m_n, EK) and gets (Hdr, *ET*), i.e., the encrypted component of the challenge. Next, the *challenger* picks a random $b \overset{\$}{\leftarrow} \{0,1\}$ and sets $c_b = (\text{Hdr}, \quad ET)$. Moreover, it is defined the function:

$$\boldsymbol{Encryptj}: \boldsymbol{Header} \times \mathbb{Z}_{\boldsymbol{p_j}} \rightarrow \boldsymbol{Rang}\left(\mathbf{Encrypt}_{\boldsymbol{y}}\right)$$

$$(\boldsymbol{Hdr}, \quad \boldsymbol{z_j}) \rightarrow \mathbf{Encrypt}_{\boldsymbol{y}}(\boldsymbol{u_1}, \boldsymbol{u_2}, \ldots, \boldsymbol{u_n}, \widehat{\boldsymbol{m}}_{\boldsymbol{1}}, \widehat{\boldsymbol{m}}_{\boldsymbol{2}} \ \ldots \ , \boldsymbol{z_j}, \ldots, \widehat{\boldsymbol{m}}_{\boldsymbol{n}}, \boldsymbol{EK})$$

  where $\widehat{m}_i = m_i$, used for computing c_b. (Let us remember that Encrypt: (Encrypt_x, Encrypt_y) was defined above). Then, $H \overset{\$}{\leftarrow} Header$ and $R \overset{\$}{\leftarrow} Rang(Encryptj)$ are picked randomly and $c_{1-b} = (H, \quad R)$ is set, i.e., the random component. It then outputs $(c_0, \quad c_1)$ to $\mathcal{A}$.
- **Query phase 2:** The *adversary* $\mathcal{A}$ continues to adaptively ask queries as in the first phase.
- **Guess:** The *adversary* $\mathcal{A}$ eventually outputs its guess $b' \in \{0, 1\}$ for *b*.

We say the *adversary* wins the game if $b' = b$, but only if $u_j \notin \Lambda_C$ and $(Hdr, \quad ET, \quad j) \notin \Lambda_D$. We then denote by $Succ^{ind}(\mathcal{A}) = Pr[b' = b]$ the probability that $\mathcal{A}$ wins the game, and its advantage is: $Adv^{ind}(\mathcal{A}) = 2 \cdot Succ^{ind}(\mathcal{A}) - 1 = Pr[1 \leftarrow \mathcal{A} | b = 1] - Pr[1 \leftarrow \mathcal{A} | b = 0]$.

Definition 1 (Full Security): A broadcast encryption scheme is said (t, ε, q_C, q_D, q_E) – secure if for any *t*-time algorithm $\mathcal{A}$ that makes at most q_C corruption queries, q_D decryption queries and q_E encryption queries, $Adv^{ind}(\mathcal{A}) \leq negl(\cdot)$, where $negl(\cdot)$ is a negligible function. We denote by $Adv^{ind}(t, \varepsilon, q_C, q_D, q_E)$ the advantage of the best time *t*-time *adversary*.

There are two classical restricted scenarios: a selective attacker provides the target users at the security game, and one can also restrict the *adversary* not to ask some queries.

Definition 2 (Basic Selective Security): A broadcast encryption scheme is said (t, ε, q_C) – selectively secure if it is (t, ε, q_C, 0, 0) – secure against a selective adversary. We denote by $Adv^{b\text{-}ind}(t, q_C)$ the advantage of the best time *t*-time basic selective adversary.

Definition 3 (Strong Selective Security): A broadcast encryption scheme is said (t, ε, q_C, q_D, q_E)– selectively secure if it is (t, ε, q_C, q_D, q_E) – secure against a selective adversary. We denote by $Adv^{s\text{-}ind}(t, q_C, q_D, q_E)$ the advantage of the best time t-time strong selective adversary.

3 Proposed solutions

In this section we first describe a proposal with basic security, for a better understanding of the process in general. We can realize that the security is only basic because it is a deterministic encryption. Then, a second proposal avoids that problem by using a random parameter, strengthening the security.

3.1 Basic security

In this proposal we do not use the encryption header because the "noise" introduced by Chinese Remainder Theorem parameters is enough in order to hide the clear text.

Description. Let us now formally describe our scheme for Short Message Basic Broadcast Encryption (SMBBE). We shall then prove its security.

- **Setup** (λ)**:** The algorithm takes as input the parameter security λ, it generates the global parameters *param* of the system as follows: first, the algorithm randomly picks n primes p_i. Then it sets $N = \prod_{i=1}^{n} p_i$ and $EK = (p_1, p_2, \ldots, p_n)$ where every prime of this set is the secret decryption key of each user.
- **Extract** algorithm will then send the corresponding key to each user.
- **Encrypt** ($u_1, u_2,\ldots,u_n, m_1, m_2, \ldots, m_n$, EK): Set $ET = \left(\sum_{i=1}^{n} m_i \frac{N}{p_i}\left[\left(\frac{N}{p_i}\right)^{-1} (mod\ p_i)\right]\right) (mod\ N)$. Its output is ET, the encrypted multiplexed text, computed with all plaintexts.
- **Decrypt** (ET, i, p_i): The algorithm computes $m_i = ET(mod\ p_i)$. This result is proved by the Chinese Remainder Theorem.

Security.

Theorem 1:. SMBBE is (t, ε, q_C) – selectively secure.

Proof:. We claim, without loss of generality for the proof [15], that $\Lambda_C=\{u_1, u_2,\ldots,u_n\}\backslash\{u_i, u_j\}$, where u_j is the attacked user. The set of non-corrupted nodes must contain two or more elements because if there is only one, u_j, then it is very easy to compute $p_j = \frac{N}{\prod_{k\neq j} p_k}$. Since the output of the function $Encrypt(u_1, u_2,\ldots,u_n, m_1, m_2, \ldots, m_n$, EK) is only an integer $ET \in \mathbb{Z}_N$, then we adapt the challenge, being $c_b = ET$, and $c_{1-b} = R$, where $Encrypt_y(u_1, u_2,\ldots,u_n, m_1, m_2, \ldots, m_n$, EK) = $Encrypt(m_j)$.

Since that function accomplishes all the Chinese Remainder Theorem's requirements, then *Encryptj* is an injective function and also bijective in *Rang*(*Encryptj*).

Hence, if m_j is a random variable, picked randomly in $\mathbb{Z}_{p_j}$, we can define another random variable $E: \mathbb{Z}_{p_j} \rightarrow Rang(Encryptj)$ with identical probability distribution, via *Encryptj*, i.e., $Pr[E(x) = z] = Pr[x = Encryptj^{-1}(z)]$. Now, we can say that *Encryptj* is a pseudorandom function. Thus,

$$\left| Pr\left[D\left(Encryptj(m_j)\right) = 1\right] - Pr\left[D(E(x))\right] = 1 \right| \leq negl(p_i) \quad (1)$$

where *D* is a distinguisher function of randomness and *negl* is a negligible function.

Now we assume that there exists an adversary $\mathcal{A}$ that obtains a non-negligible advantage for the security model challenge. Then,

$$|Pr[1 \leftarrow \mathcal{A}|b = 1] - Pr[1 \leftarrow \mathcal{A}|b = 0]| > negl(p_i) \quad (2)$$

But, $|Pr[1 \leftarrow \mathcal{A}|b = 1] - Pr[1 \leftarrow \mathcal{A}|b = 0]| = \left| Pr\left[D\left(Encryptj(m_j)\right) = 1\right] - Pr\left[D(E(x))\right] = 1 \right| \leq negl(p_i)$, and this is contradictory with (2). Then, the adversary $\mathcal{A}$ does not exist. #

3.2 Strong security

We can easily realize that the basic security model works with a deterministic encryption, and therefore it will not resist an attack with chosen plaintext. To overcome this problem, we can improve it with a new paradigm embedding probabilistic encryption.

Description. Let us now formally describe our construction for Short Message Strong Broadcast Encryption (SMSBE). We shall then prove its security:

- **Setup** (λ): The algorithm takes as an input the security parameter λ. The global parameters *param* of the system are generated as follows: first, the algorithm randomly picks n primes p_i (one per user), and random $x_i \in \mathbb{Z}_{p_i}^*$, such that $g.c.d.(x_i, \quad p_i - 1) = 1$. Then it sets $N = \prod_{i=1}^{n} p_i$ and EK=((p_1, x_1), (p_2, x_2), …, (p_n, x_n)) where every pair of this set is the secret decryption key of each user that will be sent them by the **Extract** algorithm.
- **Encrypt** (u_1, u_2,…,u_n, m_1, m_2, …, m_n, EK): Pick a random scalar $\text{Hdr} \overset{\$}{\leftarrow} \mathbb{Z}^*_{\min_j p_j}$, and define $\text{Hdr}_i = min\{g \geq \text{Hdr } such\ that\ g\ is\ a\ generator\ for\ \mathbb{Z}_{p_i}^*\}$ then set $ET = \left(\sum_{i=1}^{n}\left(m_i + Hdr_i^{x_i}(mod\ p_i)\right)\frac{N}{p_i}\left[\left(\frac{N}{p_i}\right)^{-1}(mod\ p_i)\right]\right)(mod\ N)$. It outputs (Hdr, *ET*).
- **Decrypt** (Hdr, *ET*, *i*, p_i, g_i): This algorithm computes $m_i = \left(ET - \text{Hdr}_i^{x_i}\right)(mod\ p_i)$. This result is proved by the Chinese Remainder Theorem.

Security. From the last description, it is easy to see that the attacks with chosen (no random) plaintext make useless the proof for the basic security. The inclusion of the random parameter Hdr in the protocol solves the problem.

Theorem 2: SMSBE is (t, ε, q_C, q_D, q_E) – selectively secure.

Proof: We claim, without loss of generality for the proof [14], that Λ_C = {u_1, u_2,…,u_n}\{u_j}, where u_j is the attacked user. Now, we only need the unique non-corrupted node to be the attacked one, because the secret key x_i is not revealed. Since the discrete logarithm assumption holds in $\mathbb{Z}_{p_i}^*$, it cannot be computed from $\mathrm{Hdr}_i^{x_i}$ $(mod\ p_i)$ by an eavesdropper.

First, we define some variables for improving the understanding of the proof. Let us name $q = \min_{j\in\langle 1,\dots,n\rangle} p_j$, and the function:

$$M: \mathbb{Z}_q^* \times \mathbb{Z}_{p_i} \to \mathbb{Z}_q^* \times \mathbb{Z}_{p_i}$$
$$(Hdr,\quad m_i) \to \left(Hdr,\quad m_i + Hdr_i^{x_i}\ mod\ p_i\right)$$

It is easy to see that M is a bijection, since $M^{-1}(h,\quad c) = \left(h,\quad c - h_i^{x_i}\ mod\ p_i\right)$, where $\mathrm{h}_i = min\{g \geq \mathrm{h}\ such\ that\ g\ is\ a\ generator\ for\ \mathbb{Z}_{p_i}^*\}$.

Hence we consider *Encryptj* to be a random variable, defined in 2.2. Now, we can define another random variable $E: \mathbb{Z}_q^* \times \mathbb{Z}_{p_i} \to Rang(Encryptj)$ with identical probability distribution, via *Encryptj*, i.e., $[E(H,\quad x) = z] = Pr[(H,\quad x) = Encryptj^{-1}(z)]$. Now, we can say that it is a pseudorandom function. Thus,

$$\left|Pr[D(Encryptj(Hdr,\quad m_i)) = 1] - Pr\left[D\left(E(H,\quad x)\right)\right] = 1\right| \leq negl(q, p_i) \qquad (3)$$

where D is a distinguisher function of randomness and *negl* is a negligible function.

Now, we assume that there exists an adversary $\mathcal{A}$ that obtains a non-negligible advantage for the security model challenge. Then,

$$|Pr[1 \leftarrow \mathcal{A}|b = 1] - Pr[1 \leftarrow \mathcal{A}|b = 0]| > negl(q, p_i) \qquad (4)$$

But, $|Pr[1 \leftarrow \mathcal{A}|b = 1] - Pr[1 \leftarrow \mathcal{A}|b = 0]| =$
$\left|Pr\left[D\left(Encrypt(u_1, u_2, \dots, u_n, m_1, m_2, \dots, m_n, \mathrm{EK})\right) = 1\right] -\right.$
$\left. Pr[D(H,\quad E(H,\quad x))] = 1\right| = \left|Pr[D(Encryptj(Hdr,\quad m_i)) = 1] -\right.$
$\left. Pr\left[D\left(E(H,\quad x)\right)\right] = 1\right| \leq negl(q, p_i)$, and this is contradictory with (4). Then, the adversary $\mathcal{A}$ does not exist. #

4 Security vs efficiency

Once the proposed solutions have been presented in the previous section, in the present one we analytically study the efficiency tradeoff in a wireless scenario: the IEEE 802.11 (WiFi) standard.

Given the needs of communication of small packets in terms of security and efficiency, the restrictions implied by the use of SMSBE mainly limit the size of the used keys. Our main objective is to find a balance between efficiency and security that are usually becoming antagonistic issues.

4.1 Bandwidth efficiency

For making a quick balance, we will assume that our approach sends multicast frames including information for *n* users. We will also assume that users' payload size is i <k bytes. In the standard method, the length of the whole encrypted text would be $n\left(48 + 16\left\lceil\frac{i}{16}\right\rceil\right)$, since each of the *n* packets is built with a 40-byte MAC header and 8-byte CCMP header. In our approach, we broadcast the same encrypted payload with $k(n+1)+40$ bytes. Rounding $16\left\lceil\frac{i}{16}\right\rceil \approx i$, then, our proposal will potentially be more efficient when $n(48+i)>k(n+1)+40$ bytes $\Rightarrow\ i > \frac{k(n+1)+40}{n} - 48\ bytes$.

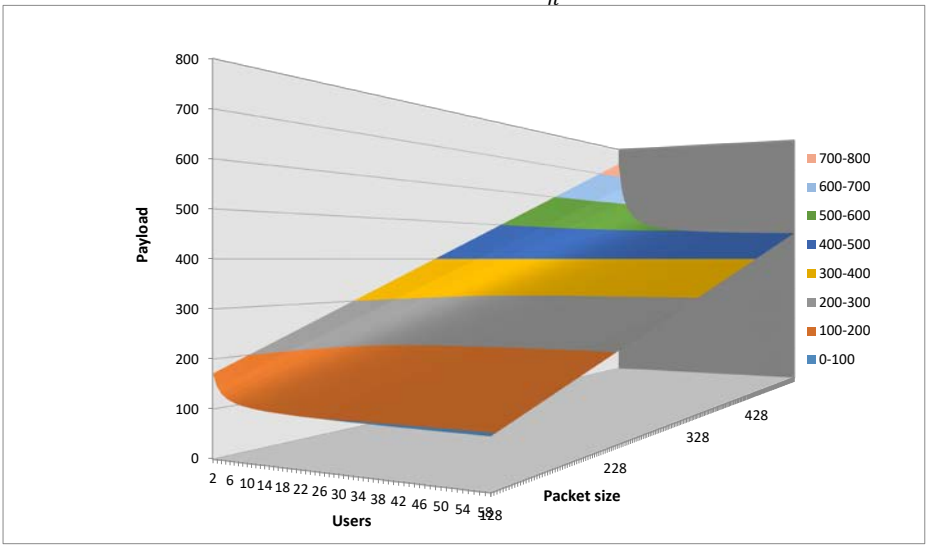

Fig. 1. Delimiter of the areas for improvement of the proposal for IEEE 802.11.

Therefore, our proposal is more efficient for 802.11, provided that user's packets size has an upper bound. The savings will be optimal if the key size is only slightly higher than the packet size. Thus, in services that send packets of the same size (e.g. VoIP), an optimal key can easily be selected. In Fig. 1 we represent the delimiter of the areas for improvement of our proposal. It should be noted that efficiency is improved in the space above the curve. It can be observed that the minimum average size grows linearly with the level of security.

4.2 Channel Utilization Efficiency

When we analyze a wireless link, we must take into account not only the bandwidth (in terms of bytes sent), but it is also necessary to consider the mechanisms for medium access. If we have to wait for the medium to be free, then when we have to send a lot of packets through it, we will accumulate a lot of waiting time. Hence, sending a smaller number of frames (a single frame for a number of users instead of a number of unicast

ones) can improve the spectrum utilization. The version 802.11n of the standard included two aggregation mechanisms: A-MPDU (Aggregated Media Access Control Protocol Data Unit, that sends a number of MPDUs together, and A-MSDU (Aggregated Media Access Control Service Data Unit), that makes the same at MSDU level [3]. The use of these aggregation mechanisms for sending multicast frames has been proposed in the literature [15, 16].

In Fig. 2 we compare our proposal of using secured multicast A-MSDU, versus the use of secured A-MPDUs, in terms of efficiency in the downlink. The size of each of the aggregated packets is determined by the size of the key: e.g. 128 bytes for the 1024 bits key. Three different key sizes (1024, 2048 and 4096 bits) are used. Different numbers of UDP packets are aggregated (X axis).

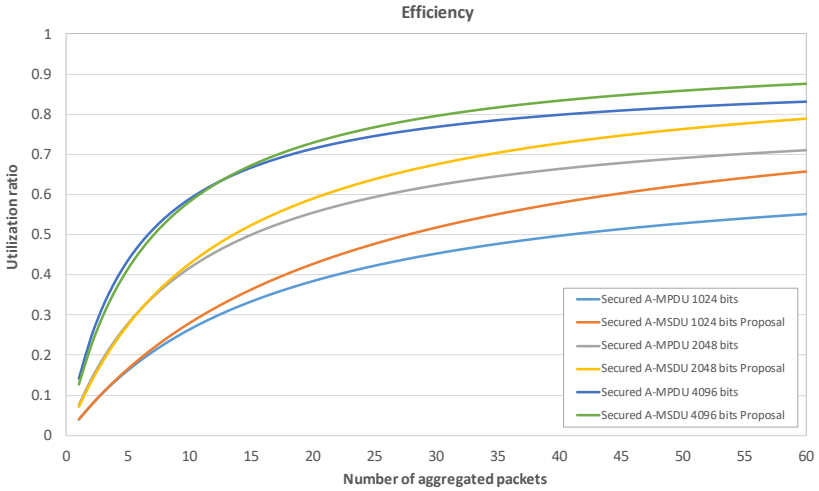

Fig. 2. Channel utilization ratio with 1024, 2048 and 4096 bit-long average packet size

We can see how the general efficiency grows when the number of aggregated packets or the bit-long average packet size is increased. It can be observed that in general, our proposal based on multicast A-MSDUs outperforms the one based on A-MPDUs. Only for the biggest key (4096 bits) it presents a lower performance if the number of users is below 13.

5 Conclusions

We have designed an efficient broadcast encryption system for traffic of small packets. We have used the Chinese Remainder Theorem to multiplex the encrypted messages and made unique the random source for encryption. The result improves the bandwidth saving and the air time connection when compared to encryption of multiple unicast packets, granted that the average packet size requirements are satisfied.

References

1. Huawei, Smartphone Solutions White Paper, Issue 2, 2012.07.17. Available at: https://www.huawei.com/mediafiles/CBG/PDF/Files/hw_193034.pdf, accessed 3 February 2020.
2. IEEE Std. 802-11 (1997). IEEE standard for wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specification, http://www.ieee802.org/11/, accessed 3 February 2020.
3. Ginzburg, B., Kesselman, A.: Performance analysis of A-MPDU and A-MSDU aggregation in IEEE 802.11n. In IEEE 2007 Sarnoff Symposium, pp. 1-5, IEEE. Princeton (USA), 2007.
4. Saldana, J., Fernández-Navajas, J., Ruiz-Mas, J. et al.: Improving Network Efficiency with Simplemux.In: IEEE CIT 2015, International Conference on Computer and Information Technology, pp. 446-453, IEEE, Liverpool (UK), (2015).
5. Saldana, J., Fernández-Navajas, J., Ruiz-Mas, J. et al.: Emerging Real-Time Services: Optimising Traffic by Smart Cooperation in the Network. IEEE Communications Magazine, (11), 127-136, (2013).
6. Coronado, E., Riggio, R., Villalón, J., et al.: Efficient real-time content distribution for multiple multicast groups in SDN-based WLANs. IEEE Transactions on Network and Service Management.15(1), 430-43 (2017)
7. Boneh,D., Gentry, C., Waters, B.: Collusion resistant broadcast encryption with short ciphertexts and private keys. In V. Shoup, editor, Advances in Cryptology – CRYPTO 2005, LNCS, vol. 3621, pp. 258–275, Springer, (2005).
8. Goh, E., Shacham, H., Modadugu, N., et al.: Sirius: Securing remote untrusted storage. In Proceedings of Network and Distributed System Security Symposium 2003, pp. 131–145, San Diego (USA), (2003).
9. Kallahalla, M., Riedel, E., Swaminathan, R., et al.: Plutus: Scalable secure file sharing on untrusted storage. In Proceedings of USENIX Conference on File and Storage Technologies (FAST) 2003, pp. 29-42, USENIX, San Francisco (USA), (2003).
10. Yan, Z., Li, X.Y., Wang, M.J., Vasilakos, A.: Flexible data access control based on trust and reputation in cloud computing. IEEE Transactions on Cloud Computing, 5(3), 485-498, (2017).
11. Brandenburger, M., Cachin, C., Knežević, N.: Don't trust the cloud verify: Integrity and consistency for cloud object stores. ACM Transactions on Privacy and Security (TOPS), 20(3), Article no. 8, (2017).
12. Zheng, W., Li, F., Popa, et al.: MiniCrypt: Reconciling Encryption and Compression for Big Data Stores. In Proceedings of the Twelfth European Conference on Computer Systems (EuroSys '17), pp. 191-204, ACM Press, New York (USA), (2017).
13. Phan, D.H., Pointcheval, D., Trinh, V.C.: Multi-channel broadcast encryption. In K. Chen, Q. Xie, W. Qiu, N. Li, W. G. Tzeng, (eds), ASIACCS 13, pp. 277–286. ACM Press, Hangzhou (China), (2013).
14. Baudron, O., Pointcheval, D., Stern., J.: Extended notions of security for multicast public key cryptosystems. In E. Welzl, U. Montanari, J.D.P. Rolim, (eds.) ICALP2000. LNCS, vol. 1853, pp. 499-511,Springer, Heidelberg (2000)
15. Park, Y.D., Jeon, S., Kim, K., et al.: Ramcast: Reliable and adaptive multicast over IEEE 802.11n wlans. IEEE Communications Letters 20(7), 1441-1444, (2016)
16. Park, Y.D., Jeon, S., Jeong, J.P., et al.: FlexVi: PHY Aided Flexible Multicast for Video Streaming over IEEE 802.11 WLANs. IEEE Transactions on Mobile Computing. (2019). In press.