Broadcast Encryption with Free-Riders and Unconditional Security

Andre Adelsbach and Ulrich Greveler

Horst Görtz Institute for IT Security
Ruhr University Bochum
Germany
(andre.adelsbach,ulrich.greveler)@nds.rub.de

Abstract. We propose two schemes for efficient broadcast key establishment that enables a sender to communicate to any subset of the user-base by allowing a small ratio of *free-riders*. The schemes do not require stateful receivers and one scheme is unconditionally secure. The free-riders are unable to learn from the past whether they might become free-riders for a certain transmission again.
We present a new trade-off facet for broadcast encryption, namely the number (or ratio) of free-riders vs. the number of messages to be sent or the number of keys stored by each user. Finally, our schemes follow an approach, related to the gradual exchange of a secret, which has been previously mainly considered in the context of fair exchange. As such, it offers a new approach to the problem of broadcast encryption, which may result in further advanced protocols.

1 Introduction

Solutions to the problem of transmitting data to a group of receivers in a way that only the correct subset of all possible receivers can decrypt the data is needed by a number of applications: pay-TV, audio streaming, real-time business data, multicast communication are current examples. The subset of receivers can change for every transmission (e.g. pay-per-view) so an efficient scheme for a quick establishment of a secure channel to the new subset is desirable.

In the literature one can find very efficient revocation schemes which are suitable for a small set $\mathcal{R}$ with $|\mathcal{R}| \ll |\mathcal{N}|$ of revoked receivers (e.g. pirate receivers or traitors) compared to a huge number of total users $\mathcal{N}$ so that the broadcast communication can only be decrypted by the users $\mathcal{N}-\mathcal{R}$. The most efficient known schemes [1–3] require a message header of lenght $O(|\mathcal{R}|)$ and user's individual private key size of $O(\log(|\mathcal{N}|))$. However, these revocation schemes are not intended nor suitable for a general subset case, e.g., cases where $|R| \approx \frac{1}{2}|\mathcal{N}|$.

The trivial scheme to address exactly all the users in the target set $\mathcal{N}-\mathcal{R}$ is to send the message encrypted individually for each user yielding a total number of $|\mathcal{N}-\mathcal{R}|$ messages to be sent via broadcast and only $O(1)$ keys to be stored by a user. This straightforward scheme is still the best known approach for the case where $|\mathcal{N}-\mathcal{R}| \ll |\mathcal{N}|$.

In this paper we will consider the case where *arbitrary* subsets are addressed by the sender. Assuming that any subset of $\mathcal{N}$ is chosen with equal probability for a transmission an average number of $\frac{1}{2}|\mathcal{N}|$ messages need to be sent via the broadcast channel for every transmission if the trivial scheme is used. In order to reduce this number it is possible to assign keys to certain or all subsets of $\mathcal{N}$ and make these keys known only to the members of the subset. But even in the best (and not realistic) case where each user is provided with a key for all $2^{|\mathcal{N}|-1}$ subsets it belongs to, the numbers of bits needed to encode the subset key identifier is approximately $|\mathcal{N}|$ so any scheme which addresses the exact subsets would need to send $O(|\mathcal{N}|)$ message bits. Apart from

that lower bound, a trade-off between the number of key stored by each user and the number of messages to be sent to establish a transmission session key needs to be considered as well as the number of colluders (users outside the target group co-operating to break the scheme) the system can tolerate is a major parameter. Finally, we are interested in the level of security (existence of one-way functions, number-theoretic or information-theoretic security) we can establish.

1.1 Relaxed Requirements, new Trade-Offs

In order to set up schemes that are more efficient than sending $|\mathcal{N}| - |\mathcal{R}|$ messages we are relaxing the requirement that only the users in the target group $\mathcal{T} := \mathcal{N} - \mathcal{R}$ can decrypt the message by allowing a certain (small) number of users in $\mathcal{R}$ to also being able to decrypt the transmission as long as every user in $\mathcal{T}$ can receive the transmission. In this case new requirements on a relaxed scheme are to be considered: The number of users who can receive a transmission they have not subscribed to, i.e. the number of *free-riders*, shall be minimized and — following economic, game-theoretic requirements (see e.g. [4]) — a user shall not gain any information whether she might be a free-rider for a future transmission by examining the past transmissions.

For example in a pay-TV scenario, we want to avoid a situation where two users $u_1, u_2 \in \mathcal{N}$ are put in one subscription set so that each time user u_1 subscribes to a transmission the user u_2 becomes a free-rider. The user u_2 might learn that he often becomes a free-rider for a certain kind of transmissions preferred by u_1 (e.g., *Tarantino*-movies) and will stop subscribing for these transmissions to avoid *unnecessary* payment.

The main area of trade-off parameters to be considered in this relaxed notion of broadcast encryption is number (or ratio) of free-riders versus message header lenght versus user key size. Other major requirements on a scheme are collusion resiliency (how many non-subscribers may collude without being able to access the secured transmission?) and underlying security assumptions (e.g., unconditional security versus computational security).

1.2 Related Work

The notion of broadcast encryption was introduced by Fiat and Naor in [5]. Their work described several methods making it possible to remove users from the target group by setting the requirement that only t users may collude where $t \leq k$ (*k-resiliency*). One method unconditionally achieves a message header of size $O(k^3 \log |\mathcal{N}|)$ and a user key storage of $O(k|\mathcal{N}| \log |\mathcal{N}|)$. The method is improved by assuming the existence of one-way-functions to user key storage of $O(k \log^2 |\mathcal{N}|)$ and by assuminung hardness of root extraction modulo a composite to user key storage $O(k \log |\mathcal{N}|)$.

Naor, Naor and Lotspiech [1] presented their *complete subtree* method that is secure under any number of colluders ($|\mathcal{R}|$-resiliency) and requires a header length of $|\mathcal{R}| \log(|\mathcal{N}|/|\mathcal{R}|)$ and $O(\log |\mathcal{N}|)$ keys per user. An improved version, the *subtree difference* method, requires header length $2|\mathcal{R}| - 1$ and $O(\log^2 |\mathcal{N}|)$ keys per user. Both methods are very efficient in the $|\mathcal{R}| \ll |\mathcal{N}|$ case and use PRNGs to assign keys in a Steiner tree structure.

Halevy and Shamir [2] presented a modified subset difference method with $O(\log^{1+\epsilon}(|\mathcal{N}|)$ key storage and $O(|\mathcal{R}|/\epsilon)$ message header size where ϵ can be chosen ($\epsilon = 2$ is a natural choice).

Dodis and Fazio [6] derived from the subtree difference scheme a public-key broadcast encryption scheme with a short public key.

Boneh and Silverberg [7] showed that by using n-linear maps a collusion secure scheme with a fixed size public key and message header lenght can be achieved; Boneh and Waters [8] improved this

by limiting a modified scheme to bilinear maps. Both schemes do not provide information-theoretic security.

Luby and Staddon [9] considered the information theoretic case and give general lower bounds for revocation schemes. Applying these bounds to the general case (i. e., not assuming $|\mathcal{R}| \ll |\mathcal{N}|$) shows that broadcast schemes with unconditional security are never efficient in the sense that either the message header lenght is large or the user key size is large.

1.3 Summary of results and Outline

In the following section we will propose two new broadcast encryption schemes operating in a pseudo-probabilistic way. Both schemes realise their effiency by accepting a adjustable ratio of free-riders. The first scheme is unconditionally secure but puts certain undesirable constraints on the abilities of attackers, the improved scheme is information-theoretic secure and lacks these constraints. We will give a calculation of the paramter trade-offs of our schemes and discuss the collusion resiliency.

2 The Biased Sub-Set Scheme

2.1 Notations, Definitions and Basic Idea

Let $\mathcal{N}$ be the set of all users of a broadcast scheme and $\mathcal{T}$ be the set of users which shall receive a certain transmission[1]. Each user $u \in \mathcal{N}$ is provided with a fixed set of secret keys K_u which are assigned to him before receiving any transmission. Each user owns at least one individual key $k_u^{indv} \in K_u$ only known to him and the sender; the others keys might also be shared between several users, which is not known to the users sharing a key. During a transmission any user might receive further one-time usage keys (session keys, key encryption keys, ...) which are not re-used and do not need to be stored after the transmission (thus we have a *stateless receiver*).

The basic idea of our scheme is to transmit the session key for a certain transmission bit-wise in a probabilistic way to all users in $\mathcal{N}$ so that the users in $\mathcal{T}$ receive on the average more key bits than the users in $\mathcal{N} - \mathcal{T}$, thus only a small fraction of the users in $\mathcal{N} - \mathcal{T}$ is able to decrypt the transmission. Most users in $\mathcal{T}$ are provided with enough bits of the session key to derive the full key after exhaustive search. For the great majority of the users in $\mathcal{N} - \mathcal{T}$ it is infeasible to derive the session key in due time (e.g. before the transmission starts or before the transmitted data becomes outdated).

We choose a security parameter s and the generated session key k_S consists of $|k_S| = s$ bits. For the users in $\mathcal{T}$ a minimum of $d < s$ bits is needed to derive k_S (d is chosen according to the computation power of the users). We assume potential attackers could be more powerful than the ordinary users, so they only need $d' \leq d < s$ bits to derive k_S in due time. The goal of the scheme is then that at a protocol step, the great majority of users in $\mathcal{T}$ has received more than d bits when at the same time only a small minority of users in $\mathcal{N} - \mathcal{T}$ has received d' or more bits (see Figure 1).

Our scheme works in two phases: first a number of messages each carrying one key bit of k_S is broadcasted (each message can only be decrypted by a different subset of $\mathcal{N}$ provided with the

[1] We will not use the notion of a set $\mathcal{R}$ of revoked users in this paper further as we address the problem of broadcasting to arbitrary subsets, so the set $\mathcal{N} - T$ does not refer to a small set of revoked users but to a set of users that have not subscribed to a certain transmission but might subscribe again to future transmissions

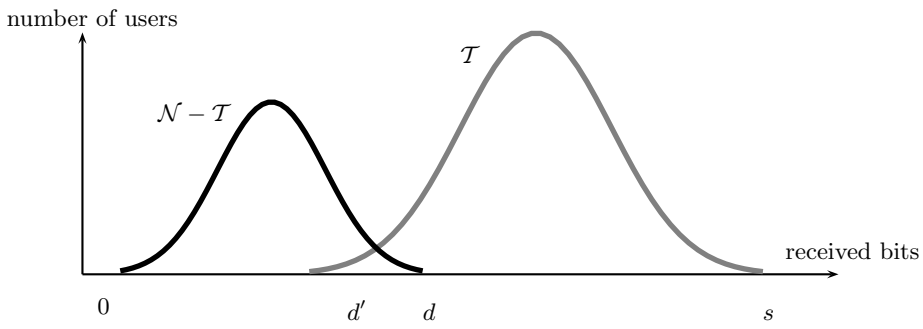

Fig. 1. Distribution of received key bits

right subset key) so that a certain number of the users $\mathcal{T}' \subset \mathcal{T}$ has received at least d bits (e.g. targeting $\frac{|T'|}{|T|} > 0.95$). In the second phase each user in $\mathcal{T} - \mathcal{T}'$ is provided individually with the full session key using the keys k_u^{indv} for all $u \in \mathcal{T} - \mathcal{T}'$.

Remark 1. Our approach broadcasts a secret by gradually broadcasting parts (bits or shares) of the overall secret to certain subsets so that any party having enough bits (or shares) can compute or recover the complete secret, e.g., the session key of a pay-TV broadcast transmission. The gradual transmission of secrets has been previously applied in the context of fair exchange [10]. In this context there is an additional "verifiability" requirement, as released parts of the secret have to be verifiable, such that a cheating party cannot gain valid parts of honest party's secret, while sending random bits to this party. In the broadcast encryption setting the verifiability requirement can be neglected as the sender is trusted in this classical model and it is only a one-way release of secrets. This gradual probabilistic broadcast of secrets represents, to the best of our knowledge, a new probabilistic approach to broadcast encryption, which may foster further advance in broadcast encryption.

2.2 Setting up the scheme

The sender selects NK subsets $N'_1 \dots N'_{NK} \subset \mathcal{N}$, which are chosen uniformly from the set of all subsets of $\mathcal{N}$ with $\frac{1}{2}|\mathcal{N}|$ elements so $|N'_i| = \frac{1}{2}|\mathcal{N}|$ for all $i = 1 \dots NK$. For each subset N'_i a *subset key* $k_{N'_i}$ is generated and let $k_{N'_i} \in K_u \Leftrightarrow u \in N'_i$, so each user knows the key assigned to each subset she belongs to, but she does not know any other subset keys, so she stores $\approx \frac{1}{2}NK$ subset keys in total (note that key storage could be reduced heavily if a PRNG based algorithm is used to generate keys before usage but then unconditional security is not achievable).

We simplify the notion here as for implementation we do intend to split the users $\mathcal{N}$ in equal sized batches of users and choose the subsets and the keys for each batch individually so we can parameterize the batch size, reduce the number of necessary subsets and are able to add new users to the broadcast scheme batch-wise after the scheme is in broadcasting operation. As the scheme is then set up and run for each batch individually we can still assume $\mathcal{N}$ to be the set of users, although $|\mathcal{N}|$ might be a rather small fixed-length number (e.g. ten thousand) compared to the possible millions of users of the full broadcasting group.

2.3 Broadcasting

For a transmission a set $\mathcal{T} \subset \mathcal{N}$ of valid subscribers is given and a session key k_S is generated for this transmission.

In order to broadcast k_S to the set $\mathcal{T} \subset \mathcal{N}$ of users the sender sorts the subsets N'_i so that we can assume for the sorted subsets $N_i := N'_{p(i)}$ that

$$N_i \geq_{\mathcal{T}} N_{i+1} \qquad \forall i : 1 \leq i < NK \tag{1}$$

where for arbitrary subsets N_a, N_b we define

$$N_a \geq_{\mathcal{T}} N_b :\Leftrightarrow |N_a \cap \mathcal{T}| \geq |N_b \cap \mathcal{T}| \tag{2}$$

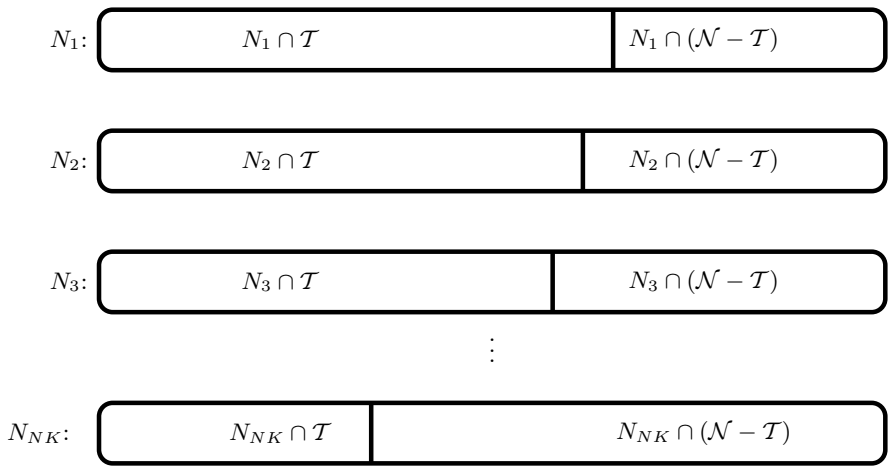

Fig. 2. Biased subsets

using a suitable permutation p for sorting the subsets. Loosely speaking, we let N_1 be the subset containing the highest number of subscribed users, N_2 is next with the second greatest bias towards the number of subscribers, so for small indices i, the bias of the subsets N_i towards $\mathcal{T}$ is high, see Figure 2 for illustration.

The scheme consists of two phases.

Phase 1 The session key k_S will be transmitted bit-wise: Let $k_{S,1}, k_{S,2}, \ldots, k_{S,s}$ denote the session key bits. First bit $k_{S,1}$ is sent to subset N_1 using subset key k_{N_1}, then $k_{S,2}$ is sent to N_2 etc. until all bits are sent. Let $\mathcal{T}'_j \subset \mathcal{T}$ be the set of users in $\mathcal{T}$ which have received at least d session key bits after $k_{S,j}$ is broadcasted.

Phase 2 For each user in $u \in \mathcal{T} - \mathcal{T}'_s$ we provide the full session key by using her unique secret key k_u^{indv} to encrypt an individual message for her and send it via the broadcast channel.

We are now interested in the number of messages sent in the two phases and the number of free-riders who can decrypt the session key although being a user in $\mathcal{N}-\mathcal{T}$ because they received d' key bits or more.

Theorem 1. *The number of free-riders given as a ratio of all users in $\mathcal{N}-\mathcal{T}$ can be approximated by $FR_{rat} = \Phi_{\mathcal{N}-\mathcal{T}}(d')$ where $\Phi_{\mathcal{N}-\mathcal{T}}$ is the distribution function of the the normal distribution $N((1-\overline{t_s})s, \sqrt{s\overline{t_s}(1-\overline{t_s})^2})$ and where $\overline{t_s}$ is the average key-bit information received by the subscribed users per key-bit transmission: $\overline{t_s}$ can be approximated by $\overline{t_s} = \Phi(1-\frac{s}{NK})$ where Φ is the quantile function of the Gauss distribution $N(\frac{|\mathcal{T}|}{2}, \sqrt{\frac{|\mathcal{T}|}{2}\frac{|\mathcal{T}|}{|\mathcal{N}|}(1-\frac{|\mathcal{T}|}{|\mathcal{N}|})})$.*

The ratio of users receiving at least d bits in phase 1 is $SUC_{rat} = \Phi_{\mathcal{T}}(d)$ where $\Phi_{\mathcal{T}}$ is the distribution function of the the normal distribution $N(t_s s, \sqrt{s\overline{t_s}^2(1-\overline{t_s})})$.

The number of messages to be sent in phase 2, which is the the number of users not having received at least d bits in phase 1, can be given by $(1-SUC_{rat})|T| = (1-\Phi_{\mathcal{T}}(d))|T|$.

Proof. We first calculate the bias of the sub-sets $N_1 \dots N_j$. As the subsets were chosen uniformly we can approximate the binomial distribution of the values $t'_i := |N'_i \cap \mathcal{T}|$ for unsorted subsets N'_i to be Gaussian i.e. $\forall i = 1 \dots NK : t'_i \sim N(\frac{|\mathcal{T}|}{2}, \sqrt{\frac{|\mathcal{T}|}{2}\frac{|\mathcal{T}|}{|\mathcal{N}|}(1-\frac{|\mathcal{T}|}{|\mathcal{N}|})})$. After sorting the NK subsets we have the most biased s values $t_i := |N_i \cap \mathcal{T}| > \frac{|\mathcal{T}|}{2}$ for $i = 1 \dots s$ with average values

$$\overline{t_i} = \Phi(1-\frac{i}{NK}) \tag{3}$$

where Φ be the quantile function of the Gaussian distribution $N(\frac{|\mathcal{T}|}{2}, \sqrt{\frac{|\mathcal{T}|}{2}\frac{|\mathcal{T}|}{|\mathcal{N}|}(1-\frac{|\mathcal{T}|}{|\mathcal{N}|})})$. As we assume the number $s << NK$ we approximate the value $t_i \approx \overline{t_s}$ for all $i < s$ so the bias' of all the subsets used for transmitting the key-bits are estimated to be equal to the bias of the last used sub-set in step s (note, the scheme is more efficient than approximated here as the other bias' are higher).

Using these approximations we can now calculate that each user in $\mathcal{T}$ has received every key-bit with probability $\overline{t_s} > \frac{1}{2}$, thus he has received $\overline{t_s}s$ key-bits on average and the number of key-bits received by each user is (by approximation) Gauss-distributed with parameters $N(\overline{t_s}s, \sqrt{s\overline{t_s}^2(1-\overline{t_s})})$. For the users is $\mathcal{N}-\mathcal{T}$ we have the probability $1-\overline{t_s} < \frac{1}{2}$, thus the distribution $N((1-\overline{t_s})s, \sqrt{s\overline{t_s}(1-\overline{t_s})^2})$.

Note, that on average a subscribed user has received $2s(\overline{t_s}-\frac{1}{2})$ more key-bits than non-subscribed users. □

Trade-Off corollary result of the theorem: The scheme can be parametrized with the values s, d. d', NK (and with $|\mathcal{N}|$ and $|\mathcal{T}|$). From these values we can calculate (approximate) free-rider ratio FR_{rat} and success-ratio SUC_{rat}, hence the number of messages: $(1-SUC_{rat})|T| + s$.

2.4 Batches of users

As mentioned before we intend to divide the user set into batches of a certain size which we still denote $|\mathcal{N}|$ to avoid unnecessary notations and run the scheme for each batch serially; the number

of batches that make up the real user-base be denoted by m so our total number of users is $|\mathcal{N}|m$. We face now the problem to select the parameters during set-up phase: batch size $|\mathcal{N}|$, number of subsets NK — and for each transmission the parameters s and d, d'. There is obviously a tradeoff: for a smaller batch size, we have better biases and need less sub-sets (and less keys to be stored by the users), but we need to run the whole scheme more often and increase the transmission lenght.

Finally we have to identify the user's and attacker's computation power in order to select for a key size s the partial-key size values d and d'. In the next section we will improve our scheme so that this is not necessary anymore.

2.5 Improvements based on Secret Sharing

The scheme introduced in the previous section uses a bit-wise distribution of the session key k_S. The scheme's security stems from the statistical certainty that unauthorised users receive on the average fewer bits (d' bits) of the session key than authorised users (d bits).

This basic scheme has some shortcomings, which are summarised below:

- The non-authorised users who are not free-riders do receive partial information as they receive a certain amount of key-bits. Thus, information-theoretic security is not achieved at all.
- Authorised users are only guaranteed to receive d bits of the session key. They have to perform an exhaustive search for the missing $s - d$ bits of the session key. This could be costly.
- Each bit that an unauthorised user does not receive (can not decrypt) doubles his computational expense required for computing the full session key k_S. However, this still requires a rather large spread $d - k_S^A$ between the number of bits received by authorised users in $\mathcal{T}$ and those received by unauthorised users in $\mathcal{N} \backslash \mathcal{T}$[2]. Furthermore, estimating the computational power of adversaries is difficult, since exhaustive key-search can be easily parallelised and media content is sufficiently popular to attract many users in participating in a parallelised search for session keys. Therefore, for being secure against powerful (distributed) attacks and offline attackers, which record the broadcasted content and decrypt it afterwards, the spread should be at least 30-40 bits.
- Attackers receive a large number of key bits. This may enable special attacks against the underlying encryption scheme, which can be significantly more efficient than naive exhaustive key search.[3]

2.6 Improved Biased Subset Scheme

Distributing the secret session key in a bit-wise manner can be seen as a naive way of sharing the secret and distributing its shares to certain sub-sets, which cover the set of authorised users. We will see in the following, that we can overcome these disadvantages by applying a cryptographic secret sharing scheme [11]:

Definition 1 (Secret Sharing Scheme). *A (k, n) secret sharing scheme consists of two algorithms:* `Share` *and* `Reconstruct`*. Given a secret s the sharing algorithm* `Share`(s) *outputs n shares*

[2] Assume that an honest authorised user, having received d bits of the session key, can compute the full session key in less than 1 minute by means of an exhaustive search for the $s - d$ missing key bits. Then a spread of 10, 15 or 20 bits takes a brute force attacker with comparable computational power 17 hours, 23 days or 728 days respectively.

[3] A straightforward countermeasure would be to apply a cryptographic hash-function on the key and use the hash value as a session key.

$s_1, \ldots, s_n$. *Given shares* $s_{i_1}, \ldots, s_{i_k}$, *the reconstruction algorithm* $\texttt{Reconstruct}(s_{i_1}, \ldots, s_{i_k})$ *outputs the original shared secret* s.
The basic security requirements of a (k, n) *secret sharing scheme are:*

- ***Availability****: Given any* k *of the* n *shares the original secret* s *can be reconstructed.*
- ***Unconditional Secrecy****: Knowledge of less than* k *shares does not reveal any information about the original secret* s.[4]

One of the first secret sharing schemes was proposed by Shamir in 1979 [11]. This scheme shares a secret $s \in F$ (e.g., $F = Z_p$ with $p > n$) by choosing a random polynomial *pol* of degree $k-1$ and with constant term s (i.e., $pol(0) = s$) over F. The shares are defined as $s_i := (i, s(i))$, $i = 1, \ldots, n$, i.e., each share is a point of the polynomial. Given k different shares, the polynomial *pol* (and consequently the secret $s = pol(0)$) can be efficiently and uniquely reconstructed by performing a Lagrange interpolation. The following theorem summarises the properties of Shamir's scheme:

Theorem 2 (Shamir's Secret Sharing Scheme [11]). *Shamir's secret sharing scheme guarantees availability and provides* unconditional secrecy*, i.e., any* $k-1$ *shares do not reveal any information about the original secret* s *(even if attackers are computationally unrestricted). Furthermore, the scheme is* computational efficient *and* optimal in the size of the shares.

The idea of applying secret sharing to overcome the limitations of the basic scheme is quite simple: The improvement is to replace the *bit-wise* broadcasting by broadcasting shares of the secret session key to the subsets instead.

In **Phase 1** of the improved scheme the sender applies a (d, s) secret sharing scheme to the session key k_S, which results in the shares $s_1, \ldots, s_s$

Instead of encrypting and broadcasting encrypted bits of the session key, the sender encrypts the share s_1 with sub-set key $k_{N'_1}$ and broadcasts the encrypted share (which can only be decrypted by members of N'_1), then s_2 is sent to N'_2, etc.

Given at least d shares a receiver can apply `Reconstruct` to efficiently reconstruct the secret k_S. Therefore, instead of performing an exhaustive search for the missing key bits a receiver only performs a Lagrange interpolation to compute the complete session key. Moreover any unauthorised user in $N-T$ is unable to gain any information about the session key as long as he receives less than d shares. This also holds information theoretically, i.e., independently of the attacker's computational power. This is a significant improvement over the basic scheme, where an attacker could use extra time or extra computation power to derive further key-bits: the threshold value d in the improved scheme is a hard threshold and provides unconditional security.

2.7 Resiliency of the scheme

The proposed schemes are highly vulnerable to colluders being able to combine their respective set of sub-set keys as these users would receive more key-bits or shares than any other user. In the case of two users sharing their sub-set key pool they would increase their portion of known sub-set keys from 0.5 to 0.75 each which is higher than a reasonable bias being achieved by sorted subsets thus the two users would become free-riders for all transmissions. Hence, the scheme does not offer any resiliency for colluders being a member of the same user batch. However, users from

[4] A computational definition of secrecy exists as well, but is omitted due to lack of space.

different batches can not gain anything from collusion as the scheme is run serially for each batch and different key encryption keys would be used so the partial key information can not be combined at all.

The resiliency is therefore 1 from a worst-case point of view or dependant on the number of batches from an average-case point of view. It can easily be seen that the birthday paradox could be applied here if every user was assigned to a certain user batch uniformly chosen. So the resiliency of our proposed schemes can be approximated by the square root of the number of batches.

Fiat and Naor [5] describe a general applicable method to convert a scheme with low resiliency (1-resilient) to one with high resiliency by randomly grouping users in small *random* sets (batches) and applying 1-resilient broadcast encryption in parallel to broadcast shares of the broadcasted secret. This construction could also be used to achieve higher resiliency for our scheme.

3 Further Improvements in Practice

3.1 Getting rid of free-riders

It is possible to avoid the existence the free-riders if the biased-subset scheme is connected with a revocation scheme (e.g., with Naor et al.'s subset-difference scheme [1]); the free-riders could be identified before the protocol starts and be removed from the set of all users by first establishing a broadcast key with revocation of the free-riders so that the biased-subset protocol only communicates to the target users and those users who will not become free-riders anyway. Note, that this extension does not support information-theoretic security anymore, but it could still be useful for implementations, especially when the sender want to avoid free-riders only for few transmissions.

3.2 Re-Using Establishment Keys for stateful receivers

If the receivers are not stateless the agreed key for a certain transmission can be learnt and re-used as a subset key for a future protocol run. In practice it is likely that the target set of one transmission is very *similar* to the target set of a related transmission, so if the key is used as a future subset key the bias towards the target set will often be much higher than that of a normal sorted subset. Hence, the scheme will become more efficient for future runs when the receivers have stored their transmission session keys for each time the receiver was in the priviliged set of users.

4 Conclusion

We proposed schemes for efficient broadcast key establishment that offer a tradeoff between the ratio of free-riders and other parameters (overall key size or message header size). The schemes do not require stateful receivers and one is unconditionally secure (the bit-wise case). Free-riders are unable to learn from the past whether they might become free-riders again, their existence can also be prevented if revocation schemes are used together with our proposed schemes.

5 Appendix: Sample Data

Fig. 3 shows several sample data values for the proposed share-wise key distribution scheme. In all cases half of the user base is in the priviliged set and the other half is not (worst-case) but different

user batch sizes, number of sub-sets, free-rider ratios are chosen so that different number of shares and total number of messages are calculated from that. The average resiliency is only dependant on the number of batches.

number of users	batch size	sub-set keys	shares	free-rider ratio	messages both phases	resiliency
1,000,000	1,000	10,000,000	64	0.1	114,000	31
1,000,000	1,000	10,000,000	100	0.05	89,000	31
10,000,000	1,000	1,000,000	125	0.05	1,500,000	100
10,000,000	10,000	1,000,000	1300	0.05	1,550,000	31

Fig. 3. Sample data for different paramters

References

1. Naor, D., Naor, M., Lotspiech, J.B.: Revocation and tracing schemes for stateless receivers. In: CRYPTO '01: Proceedings of the 21st Annual International Cryptology Conference on Advances in Cryptology, Springer-Verlag (2001) 41–62
2. Halevy, D., Shamir, A.: The LSD broadcast encryption scheme. In Yung, M., ed.: CRYPTO. Volume 2442 of Lecture Notes in Computer Science., Springer (2002) 47–60
3. Goodrich, M.T., Sun, J.Z., Tamassia, R.: Efficient tree-based revocation in groups of low-state devices. In Franklin, M.K., ed.: CRYPTO. Volume 3152 of Lecture Notes in Computer Science., Springer (2004) 511–527
4. Shavitt, Y., Winkler, P., Wool, A.: On the economics of multicasting. Netnomics **6** (2004) 1–20
5. Fiat, A., Naor, M.: Broadcast encryption. In: CRYPTO. (1993) 480–491
6. Dodis, Y., Fazio, N.: Public-key broadcast encryption for stateless receivers. In Feigenbaum, J., ed.: ACM Workshop in Digital Rights Management—DRM 2002. Volume 2696 of Lecture Notes in Computer Science., Springer-Verlag (2003) 61–80
7. Boneh, D., Silverberg, A.: Applications of multilinear forms to cryptography. Contemporary Mathematics **324** (2003) 71–90
8. Boneh, D., Waters, B.: Collusion resistant broadcast encryption with short ciphertexts and private keys. Cryptology ePrint Archive, Report 2005/018 (2005) http://eprint.iacr.org/.
9. Luby, M., Staddon, J.: Combinatorial bounds for broadcast encryption. In: EUROCRYPT. (1998) 512–526
10. Damgård, I.B.: Practical and provably secure release of a secret and exchange of signatures. Journal of Cryptology **8** (1995) 201–222 Appeared earlier in , pages 200–217.
11. Shamir, A.: How to share a secret. Commun. ACM **22** (1979) 612–613