ORIGINAL RESEARCH PAPER

How not to secure wireless sensor networks: a plethora of insecure polynomial-based key pre-distribution schemes

Chris J Mitchell

Abstract

Three closely related polynomial-based group key pre-distribution schemes have recently been proposed, aimed specifically at wireless sensor networks. The schemes enable any subset of a predefined set of sensor nodes to establish a shared secret key without any communications overhead. It is claimed that these schemes are both secure and lightweight, that is, making them particularly appropriate for network scenarios where nodes have limited computational and storage capabilities. Further studies have built on these schemes, for example, to propose secure routing protocols for wireless sensor networks. Unfortunately, as shown by the author, all three schemes are completely insecure; whilst the details of their operation vary, they share common weaknesses. In two cases, we show that an attacker equipped with the information built into just one sensor node can compute all possible group keys, including those for which the attacked node is not a member; this breaks a fundamental design objective. In the other case, an attacker equipped with the information built into at most two sensor nodes can compute all possible group keys. In the latter case, the attack can also be achieved by an attacker armed with the information from a single node together with a single group key to which this sensor node is not entitled. Repairing the schemes appears difficult, if not impossible. The existence of major flaws is not surprising given the complete absence of any rigorous proofs of security for the proposed schemes. A further recent work proposes a group membership authentication and key establishment scheme based on one of the three key pre-distribution schemes analysed here; as the author demonstrates, this scheme is also insecure, as the attack we describe on the corresponding pre-distribution scheme enables the authentication process to be compromised.

1 | INTRODUCTION

The authors are concerned with the problem of enabling groups of nodes (such as exist in a wireless sensor network (WSN)) to establish shared secret keys for use by any subset of the nodes, using only information distributed to the nodes in advance by a trusted Key Generation Centre (KGC). In particular, we examine three separate (albeit closely related) protocols described as being designed for WSNs, although they have no particular features restricting their operation to this use case. Indeed, they would work just as effectively for any case where a number of hardware tokens, such as RFID tags or smart cards, are to be distributed by a single trusted party. The schemes we consider are designed to ensure that nodes which are part of the system, but not part of a selected group, cannot access the key for that group, that is, nodes can only access keys for groups of which they are a part.

All three of the schemes we consider are polynomial based. They appear to be inspired by the work of Blundo et al. [1]; however, unlike this prior art, the schemes all have major flaws. Unfortunately, many other polynomial-based group key distribution schemes have been shown to be flawed [2–6], meaning that the approach needs to be used with care. In any event, there are many existing schemes which achieve the same objectives in an efficient way and which have rigorous proofs of security—see, e.g. Boyd et al. [7], and, of course, the previous work of Blundo et al. [1].
We also briefly examine a group membership and key establishment scheme recently proposed by Cheng et al. [8]. This scheme builds directly on the first of the schemes we cryptanalyse here, and we show how the cryptanalysis extends to this case.

The remainder of this study is structured as follows. Some preliminary remarks are made in Section 2. The 2015 Harn–Hsu protocol is described in Section 3, together with an attack against this scheme; the fact that this attack also works on the derived 2020 Cheng–Hsu–Xia–Harn scheme is also briefly discussed. The very closely related 2015 Harn–Gong scheme is briefly considered in Section 4. A more recent scheme, the 2019 Albakri–Harn protocol, is described and analysed in Section 5. Other issues are discussed in Section 6, and brief conclusions are provided in Section 7.

2 | A PRELIMINARY OBSERVATION

All the three schemes we describe involve arithmetic computed in the ring of integers \( \mathbb{Z}_N \), where \( N \) is an ‘Rivest-Shamir-Adleman (RSA) modulus’ [9], i.e. \( N = pq \) for two large prime numbers \( p \) and \( q \). This means that addition and multiplication are computed modulo \( N \). For today’s RSA applications, \( p \) and \( q \) are typically chosen to be of the order of \( 2^{1024} \), and we make this assumption throughout. This ensures that with today’s techniques and computing resources, factoring \( N \) is infeasible (and this is necessary to ensure that the RSA cryptosystem is secure).

If \( p \) and \( q \) are of this size, then the probability that a random integer \( s \) (where \( 1 \leq s < n \)) is coprime to \( N = pq \) is very close to 1. To see this, observe that \( \phi(n) \), the Euler phi function that gives the number of positive integers less than \( n \) that are coprime to \( n \) (Menezes et al. [10], Definition 2.100), satisfies \( \phi(pq) = (p-1)(q-1) \) for any primes \( p, q \) ([10], Fact 2.101). Thus, the probability that a random positive integer less than \( N = pq \) (where \( p \) and \( q \) are of the order of \( 2^{1024} \)) is coprime to \( N \) is

\[
\frac{\phi(N)}{N} = \frac{(p-1)(q-1)/pq}{2^{2048} - 2^{1023}} = 1 - 2^{-1023}.
\]

Hence, we assume throughout that all randomly chosen integers are coprime to \( N \) and so have multiplicative inverses modulo \( N \); moreover, such inverses can readily be computed ([10], Algorithm 2.142). That is, we can readily compute divisions modulo \( N \), a fact that underlies all the attacks we describe.

3 | THE HARN–HSU SCHEME

In 2015, Harn and Hsu [11] proposed a group key pre-distribution protocol for WSNs. Unfortunately, as we describe in detail below, the Harn–Hsu scheme is insecure. In particular, anyone with the shares belonging to one node can compute the keys for all possible groups, regardless of membership. The nature of the flaws giving rise to the attack is rather fundamental, meaning that it is difficult to imagine how the scheme could be rescued.

3.1 | The protocol

We next describe the Harn–Hsu protocol [11]. The protocol involves a set of \( \ell \) end nodes \( S = \{S_1, S_2, \ldots, S_\ell_\} \) and a KGC.

Note that, in this description and in the description of related protocols (in Sections 4 and 5), the terminology and notation have been changed slightly from those used in the source papers to ensure consistency. As with all such protocols, the Harn–Hsu protocol has two phases, as follows:

- **Share generation**: a pre-distribution phase in which the KGC generates a set of \( \ell - 1 \) shares \( \{s_{1,j}(x), s_{2,j}(x), \ldots, s_{\ell,j-1}(x)\} \) for each node \( S_j \in S \), \( 1 \leq j \leq \ell \), where each share \( s_{i,j}(x) \) (where \( 1 \leq j \leq \ell - 1 \)) is a polynomial of degree \( k \) over \( \mathbb{Z}_N \) (for an ‘RSA modulus’ \( N \)). It is assumed that the KGC distributes the shares to the nodes using a secure channel, for example, at the time of manufacture/personalisation, and that each node stores its shares securely.

- **Group key establishment**: where all members of a subset of nodes independently generate a group key for use by nodes in that subset. Note that, if repeated with the same subset of nodes, the key establishment process will generate the same key, but a different key is generated for each node subset.

We next describe the operation of these two phases in greater detail.

3.1.1 | Share generation

The KGC first chooses an ‘RSA modulus’ \( N \), that is, \( N = pq \) where \( p \) and \( q \) are two primes chosen to be sufficiently large to make factoring \( N \) infeasible. It is also assumed that each node \( S_j \) has a unique identifier \( ID_j \), \( j \leq \ell \), where \( 1 \leq ID_j < N \). It is (presumably) the case when the KGC keeps secret, although, as we see below, this does not appear to make a significant difference to the security of the scheme.

The KGC next chooses a polynomial \( f \) over \( \mathbb{Z}_N \) of degree \( k \) for some pre-chosen \( k \). We suppose that the polynomial coefficients are chosen uniformly at random from \( \mathbb{Z}_N \). No explicit guidance is given on the choice of \( k \), but later it is claimed that the scheme is secure if up to \( k \) nodes are captured and their secrets are revealed. We assume throughout that \( k \geq 2 \).

For every \( S_j \) (where \( 1 \leq j \leq \ell \)), the KGC calculates a set of \( \ell - 1 \) shares in the following way:

- The KGC first computes \( f(ID_j) \).
- The KGC then chooses at random \( \ell - 2 \) values \( u_{i,1}, u_{i,2}, \ldots, u_{i,\ell-2} \), where \( 0 < u_{i,j} < N \) for every \( j \) (where \( 1 \leq j \leq \ell - 2 \)).
- The KGC then computes \( u_{i,\ell-1} \) as:
\[ u_{i, \ell - 1} = \frac{f(ID_i)}{\prod_{j=1}^{\ell} u_{i,j}} \mod N \]

that is

\[ \prod_{j=1}^{\ell - 1} u_{i,j} \equiv f(ID_i) \mod N. \]

• Finally, the shares for \( S_i \) are computed as \( s_i(x) = u_i f(x) \mod N, 1 \leq j \leq \ell - 1 \), where each share is a polynomial over \( \mathbb{Z}_N \) of degree \( k \).

Finally, the KGC equips each node \( S_i \in S \) with the following:

• the shares \( \{s_i(x), s_{i,1}(x), \ldots, s_{i,\ell-1}(x)\} \);
• the values of \( N \) and the node's own identifier \( ID_i \) and
• the identifiers \( ID_j \) for all of the other nodes.

### 3.1.2 Group key establishment

Suppose some subset \( S' \subseteq S \) of sensors (|\( S' \)| ≥ 2) wish to share a group key, where |\( S' \)| = |s| (2 ≤ b ≤ \( \ell \)). The group key \( K_{S'} \in \mathbb{Z}_N \) for \( S' \) is defined to be:

\[ K_{S'} = f(0)^{\ell-b} \prod_{j \in S'} f(ID_j) \mod N. \]

Any node \( S_i \in S' \) can compute \( K_{S'} \) using the shares \( \{s_i(x), s_{i,1}(x), \ldots, s_{i,\ell-1}(x)\} \) in the following way:

• Suppose that

\[ S' - \{S_i\} = \{S_{r_1}, S_{r_2}, \ldots, S_{r_{\ell-1}}\} \]

and that

\[ S - S' - \{S_i\} = \{S_{b_1}, S_{b_2}, \ldots, S_{b_{\ell-1}}\} \]

where \( \{r_1, r_2, \ldots, r_{\ell-1}\} = \{1, 2, \ldots, \ell - 1\} \).

• Then set

\[ K_{S'} = \prod_{j=1}^{b-1} s_{i,j}(ID_j) \prod_{j=b}^{\ell-1} s_{i,j}(0). \]

However, a node \( S_k \in S' \) cannot use its shares to compute \( K_{S'} \), at least not in the way described above.

Finally, note that it should be clear that the group key \( K_S \) for the set of all sensors is simply

\[ K_S = \prod_{j=1}^{\ell} f(ID_j) \mod N. \]

### 3.2 Critical vulnerabilities

We describe below a simple insider attack which enables the discovery of all the group keys. This can be achieved using the shares held by any one of the nodes. The attack works regardless of the choice of the polynomial degree \( k \). We also show that, even without access to a set of shares, knowledge of some keys enables others to be deduced.

#### 3.2.1 Some simple deductions

We start with a simple but key observation here, and in the remainder of Section 3, where we suppose that the values \( z_1, z_2, \ldots, z_{\ell} \) satisfy

\[ f(ID_r) \equiv z_r f(0) \mod N \]

and \( 0 \leq z_r < N \) for every \( r \) (1 ≤ \( r \) ≤ \( \ell \)).

**Lemma 1** If \( s_i(x) \) is a share for user \( S_i \) (1 ≤ \( i \) ≤ \( \ell \)), then, for any \( r \) (1 ≤ \( r \) ≤ \( \ell \)):

\[ z_r = s_{i,r} f(ID_r) / s_{i,r} f(0) \mod N. \]

**Proof** 1 By definition, we immediately have:

\[ s_{i,j}(ID_r) / s_{i,j}(0) \equiv u_{i,j} f(ID_r) / u_{i,j} f(0) \mod N \]

\[ = f(ID_r) / f(0). \]

The result follows immediately from the definition of \( z_r \). □

We next have a related result.

**Lemma 2** If \( S' \subseteq S \) is some non-empty subset of the nodes, then the key \( K_{S'} \) for the group \( S' \) satisfies

\[ K_{S'} = f(0)^{\ell-b} \prod_{j \in S'} z_j \mod N. \]

**Proof** 2 By definition (and using the notation established in Section 3.1.2), we have

\[ K_{S'} = f(0)^{\ell-b} \prod_{j \in S'} f(ID_j) \mod N \]

where \( b = |S'| \). But we assumed that \( f(ID_r) \equiv z_r f(0) \mod N \), and hence

\[ K_{S'} = f(0)^{\ell-b} \prod_{j \in S'} z_j f(0) \mod N \]

and the result follows by re-arranging the products. □
This immediately gives the following.

**Corollary 1** The group key $K_S$ shared by all nodes satisfies

$$K_S = f(0)^r \prod_{j=1}^\ell z_j \mod N.$$ 

We also have the following.

**Corollary 2** Suppose $S_1 = S_2 \cup \{S_i\}$ for some $S_i (1 \leq i \leq \ell)$. Then

$$K_{S_1} / K_{S_2} \equiv z_i \mod N.$$ 

### 3.2.2 Completing the insider attack

First observe that, using Lemma 1, anyone possessing the shares belonging to a single sensor node can learn the values of $z_r$, $1 \leq r \leq \ell$, for every $r$.

Next observe that, since a single set of shares enables recovery of the group key $K_S$ shared by all nodes, the attacker can obtain $f(0)^r \mod N$ since it follows from Corollary 1 that:

$$f(0)^r = \frac{K_S}{\prod_{j=1}^\ell z_j} \mod N.$$ 

Given knowledge of $f(0)^r \mod N$ together with the complete set of values $z_1, z_2, ..., z_\ell$, Lemma 2 enables the computation of any group key $K_{S_i}$. This completes the attack.

### 3.2.3 An outsider attack

Even if an attacker does not possess any of the shares, then attacks are still possible if the attacker has access to some of the keys $K_{S_i}$ generated using the system. We give a simple example of an attack, but many other variants are possible.

Suppose an attacker has access to three group keys: $K_{S_1}, K_{S_2}$ and $K_{S_3}$, for groups $S_1$, $S_2$ and $S_3$, where the attacker also knows the membership of these groups. Suppose (to make the discussion simpler) that $S_1 = S_2 \cup \{S_y\}$ for some node $S_y$, then, by Corollary 2, the attacker can immediately compute $z_r$.

If $S_y \not\in S_3$, then the attacker can now compute $K_{S_3} = K_{S_1} / z_r \mod N$, where $S_4 = S_3 - \{S_y\}$. That is, the attacker can compute another valid group key.

This attack works because there are simple algebraic relationships between the keys.

### 3.3 Breaking the Cheng–Hsu–Xia–Harn scheme

As noted in the introduction, Cheng et al. [8] recently proposed a group membership authentication and key establishment scheme; as noted by the authors, the ‘proposed protocol is built upon a recent work by Harn and Hsu’, that is precisely the study we have analysed immediately above.

The Cheng–Hsu–Xia–Harn scheme involves a trusted KGC generating and distributing shares to all members of a group, exactly as in the Harn–Hsu scheme. When a subset of the group wishes to conduct group authentication, each subset member generates a shared key using precisely the method described in the Harn–Hsu scheme—they then use this key to authenticate to the members of the subset. However, as we have shown, every ‘subset key’ is actually computable by every shareholder, regardless of whether they are in the authorised subset or not; this means that the authentication scheme can be trivially broken.

### 4 THE HARN–GONG SCHEME

In 2015, the same year in which the Harn–Hsu scheme appeared, Harn and Gong [12] presented another group key pre-distribution scheme. It would appear that this work was actually submitted a few months earlier than the Harn–Hsu scheme. However, we consider it after Harn–Hsu as the scheme it describes is essentially just a special case of the Harn–Hsu scheme.

Similar to the Harn–Hsu scheme, a KGC distributes shares to each of a set of $\ell$ nodes. The KGC first chooses an RSA modulus $N$ and a polynomial $f(x)$ over $Z_N$ of degree $k$ for some pre-chosen $k$. The single share sent to each user $S_i$ is computed as $s_i(x) = u_i f(x) \mod N$, where $u_i$ satisfies

$$(u_i)^{\ell-1} \equiv f(1\mathbf{D}_j) \mod N.$$ 

Using the notation of Section 3, it should be clear that this is simply a special case of the Harn–Hsu scheme where the set of $\ell - 1$ shares for a user are all chosen to be equal, i.e. where $u_{i,1} = u_{i,2} = ... = u_{i,\ell-1}$ and hence $s_{i,1}(x) = s_{i,2}(x) = ... = s_{i,\ell-1}(x)$. All other aspects of the scheme are identical.

Since it is just a special case of the Harn–Hsu scheme, precisely the same attacks work and, hence, we do not consider the scheme further here.

### 5 THE ALBAKRI–HARN SCHEME

Four years after the Harn–Hsu scheme appeared, in 2019 Albakri and Harn [13] proposed yet another group key pre-distribution protocol for WSNs. Three variants of the protocol are described, a basic scheme, which has a heavy storage overhead, and two derived schemes which use the same underlying idea but reduce the storage requirement for individual
nodes. Given its fundamental role, we focus here on the basic scheme.

At first sight, the scheme is somewhat different from the two schemes we have examined so far. However, more careful analysis reveals that it is again very closely related. Not surprisingly, the Albakri–Harn scheme is also insecure. In particular, if two nodes collude, or if one node gains access to a single key to which it is not entitled, then all keys for all groups, regardless of membership, can be computed.

5.1 | The protocol

We describe the Basic scheme [13]. The protocol involves a set of \( \ell \) sensor nodes \( S = \{S_1, S_2, \ldots, S_\ell\} \) and the KGC. The protocol has two phases:

- **Token generation**: a pre-distribution phase in which the KGC generates a token \( T_i \) for each node \( S_i \in S \), \( 1 \leq i \leq \ell \). Similar to the previous schemes, it is assumed that the KGC distributes the tokens to the nodes using a secure channel, for example, at the time of manufacture/personalisation.

- **Group key establishment**: where all members of a subset of nodes independently generate a group key for use by nodes in that subset.

5.1.1 | Token generation

Again as before, the KGC first chooses an 'RSA modulus' \( N \), i.e. \( N = pq \) where \( p \) and \( q \) are two primes chosen to be sufficiently large to factor \( N \) infeasible. It is also assumed that each node \( S_i \) has a unique identifier \( ID_i \), \( 1 \leq i \leq \ell \), where \( 1 \leq ID_i < N \).

The KGC next chooses a set of \( \ell \) univariate polynomials \( F = \{f_1, f_2, \ldots, f_\ell\} \) over \( \mathbb{Z}_N \), each of degree \( t - 1 \) for some \( t \). In the absence of further information, we suppose here the coefficients of these polynomials are chosen uniformly at random from \( \mathbb{Z}_N \). No explicit guidance is given on the choice of \( t \), but later it is claimed that the scheme is secure if up to \( t - 1 \) nodes are captured (and presumably their secrets are revealed), and that the choice of \( t \) is a trade-off between the computational complexity of key establishment and the security of the scheme. We assume throughout that \( t \geq 2 \), since if \( t = 1 \) all polynomials are of degree zero and all tokens (and group keys) are identical.

For every \( i \) \( (1 \leq i \leq \ell) \), the KGC calculates the token \( T_i \) as the following polynomial in \( \ell - 1 \) variables:

\[
T_i = f_i(ID_i) \prod_{j=1 \atop j \neq i}^{\ell} f_j(x_j) \mod N.
\]

Finally, the KGC equips each node \( S_i \in S \) with the following:

- the token \( T_i \)
- the values of \( N \) and the node’s own identifier \( ID_i \)
- the identifiers \( ID_j \) of all of the other nodes.

It is important to note that some of the steps above are based on the author’s interpretation of the Albakri–Harn scheme [13], as many details are left unclear.

5.1.2 | Group key establishment

Suppose some subset \( S' \subseteq S \) of sensors \( |S'| \geq 2 \) wish to share a group key. The group key \( K_{S'} \in \mathbb{Z}_N \) for \( S' \) is defined to be:

\[
K_{S'} = \prod_{S_i \in S'} f_i(ID_i) \prod_{S_j \in S - S'} f_j(0) \mod N.
\]

Any of the nodes \( S_i \in S' \) can compute \( K_{S'} \) by evaluating the token \( T_i \mod N \) for a particular choice of the indeterminates \( x_j \), namely by setting

\[
x_j = \begin{cases} ID_j & \text{if } S_j \in S' \\ 0 & \text{otherwise.} \end{cases}
\]

However, a node \( S_k \notin S' \) cannot use its token \( T_k \) to compute \( K_{S'} \), at least not in the way described above.

Finally note that it should be clear that the group key \( K_S \) for the set of all sensors is simply

\[
K_S = \prod_{j=1}^{\ell} f_j(ID_j) \mod N.
\]

5.1.3 | An observation

It was stated above that this scheme is closely related to the Harn–Hsu scheme. This is not immediately apparent, as the Harn–Hsu scheme involves a node being given a set of shares each of which is a univariate polynomial, and in the Albakri–Harn scheme a sensor node is given a token consisting of a single multivariate polynomial. However, this token is actually analogous to the product of the Harn–Hsu shares, in the case where the shares polynomials have distinct indeterminates (although the polynomials \( f_i \) in Albakri–Harn scheme are all distinct). Armed with this insight, the protocol then works in an essentially identical way to the Harn–Hsu scheme. It is therefore not surprising that, as we discuss below, closely analogous attacks apply.

5.2 | An attack

5.2.1 | Stage 1: Partial polynomial recovery

In the discussions below, we need notation for the coefficients of the polynomials in \( F \) and, hence, recalling that all these polynomials have degree \( t - 1 \), we suppose that
for every $i$, $1 \leq i \leq \ell$.

We first consider what a single node $S_i$ can learn about the polynomials in $F$ from a single token $T_i$. It follows immediately from the definition that $T_i$ consists of the sum of all terms of the form

$$f_i(ID_i) \prod_{j=1}^{\ell} f_{jk_i} \mod N,$$

where $0 \leq k_j \leq t - 1$ for every $j$. That is, $S_i$ will know the value of

$$f_i(ID_i) \prod_{j=1}^{\ell} f_{jk_i} \mod N$$

for every pair $(i, k_j)$, where $1 \leq j \leq \ell$ ($j \neq i$) and $0 \leq k_j \leq t - 1$.

Given these observations, for any $r$ ($1 \leq r \leq \ell$, $r \neq i$) and any $s$ ($1 \leq s$), and using the known coefficients of the polynomial $T_s$, it is simple to compute the ratio of two such coefficients, namely

$$\frac{f_i(ID_i) \prod_{j=1}^{\ell} f_{jk_i}}{f_i(ID_i) \prod_{j=1}^{\ell} f_{jk_i}} \mod N$$

where $k_j = k_j'$ for every $j$ except for $j = r$ and $k_j = s$, where we put $k_r' = 0$. In this case, all the identical terms will cancel, and the above expressions will equal

$$\frac{f_r(ID_r)}{f_{r0}} \mod N.$$

That is, for every polynomial $f_r$ ($r \neq i$), the ratio of each coefficient in $f_r$ to the constant coefficient $f_{r0}$ can be computed. Hence $S_i$ can readily discover the values $w_1, w_2, ..., w_{\ell-1}$ where

$$f_r = w_r f_{r0}.$$

Moreover, given $ID_r$ (which is known to all nodes), $S_i$ can use the above to deduce that

$$f_r(ID_r) = z_r f_{r0}$$

for some $z_r$ known to $S_i$, for every $r \neq i$.

### 5.2.2 Stage 2: Pairwise collusion to complete the attack

We next describe how, if two nodes collude, for example, by sharing their tokens, they can completely break the system; that is, they will have the means to readily compute every possible group key, including for groups excluding both. That is, as is the case for all three schemes examined, the system is completely insecure if just two nodes collude, regardless of the choice of $t$.

We first need the following simple result (analogous to Lemma 2), which uses the notation of Section 5.1.

**Lemma 3** Suppose the values $z_1, z_2, ..., z_\ell$ satisfy

$$f_r(ID_r) \equiv z_r f_{r0} \pmod{N}$$

for every $r$ ($1 \leq r \leq \ell$). If $S' \subseteq S$ is some non-empty subset of the nodes, the key $K_{S'}$ for the group $S'$ satisfies:

$$K_{S'} = \prod_{r \in S'} z_r \prod_{r \in S \setminus S'} f_{r0} \pmod{N}.$$

**Proof** By definition, we have

$$K_{S'} = \prod_{r \in S'} f_r(ID_r) \prod_{r \in S \setminus S'} f_{r0} \pmod{N}.$$

We assumed that $f_r(ID_r) = z_r f_{r0}$ mod $N$ for every $r$, and it trivially holds that $f_r(0) = f_{r0}$ for every $r$. Hence

$$K_{S'} = \prod_{r \in S'} z_r f_{r0} \prod_{r \in S \setminus S'} f_{r0} \pmod{N}.$$

The result now follows by re-arranging the products. $\square$

The following corollary (analogous to Corollary 1) is immediate.

**Corollary 3** The group key $K_S$ shared by all nodes satisfies

$$K_S = \prod_{r=1}^{\ell} z_r \prod_{r=1}^{\ell} f_{r0} \pmod{N}.$$

We now observe that, from the arguments in Section 5.2.1, if any two users collude then they can learn the complete set of values $z_1, z_2, ..., z_\ell$ satisfying $f_r(ID_r) = z_r f_{r0}$ mod $N$, $1 \leq r \leq \ell$.

Next observe that, since they can both readily compute the group key $K_S$ shared by all nodes, they can obtain

$$\prod_{r=1}^{\ell} f_{r0} \pmod{N}$$

since it follows from Corollary 3 that

$$\prod_{r=1}^{\ell} f_{r0} = \frac{K_S}{\prod_{j=1}^{\ell} z_j} \pmod{N}.$$
Armed with knowledge of \( \prod_{\ell=1}^{\ell} f_{\ell} \mod N \) together with the complete set of values \( z_1, z_2, \ldots, z_{\ell} \), the colluding nodes can use Lemma 3 to immediately compute any group key \( K_{S_i} \), regardless of whether either of the colluding nodes are members of the group \( S' \). This completes the attack.

5.2.3 | An alternative to collusion

Just as previously, even in this case it remains possible to completely break the system if a node (by some means) learns a single key for a group of which the node is not a member. Suppose node \( S_i \) knows the key \( K_{S'} \) for the group \( S' \), where \( S_i \notin S' \). By Lemma 3, \( S_i \) knows that

\[
K_{S'} = \prod_{j \in S'} z_j \prod_{\ell=1}^{\ell} f_{\ell} \mod N.
\]

But \( S_i \) knows all the values \( z_j \) for \( S_j \in S' \), since \( S_i \notin S' \). Hence \( S_i \) can compute \( \prod_{\ell=1}^{\ell} f_{\ell} \mod N \) as

\[
\prod_{\ell=1}^{\ell} f_{\ell} = \frac{K_{S'}}{\prod_{j \in S'} z_j} \mod N.
\]

Knowledge of \( \prod_{\ell=1}^{\ell} f_{\ell} \mod N \) together with the (almost) complete set of values \( z_1, z_2, \ldots, z_{\ell} \) (except for \( z_i \)), enables \( S_i \) to now compute any group key \( K_{S'} \) for any group \( S' \) for which \( S_i \notin S' \). This completes the attack.

5.3 | A simplified attack

Using the same notation as before, we now present an even simpler version of the attack approaches described in Section 5.2. This attack does not seek to recover information about the polynomials in \( F \), but instead recovers just sufficient information to be able to recover all group keys. We use the notation established in Section 5.2.

5.3.1 | Learning a set of ratios

We start with another simple observation, analogous to Corollary 2.

**Lemma 4** Suppose \( S_i = S - S_i \). Then

\[
K_S / K_{S_i} = z_i \mod N
\]

for every \( i, 1 \leq i \leq \ell \).

**Proof** By Lemma 3,

\[
K_S = \prod_{j=1}^{\ell} z_j \prod_{\ell=1}^{\ell} f_{\ell} \mod N.
\]

Similarly, by Corollary 3,

\[
K_{S_i} = \prod_{j \neq i}^{\ell} z_j \prod_{\ell=1}^{\ell} f_{\ell} \mod N.
\]

The result follows immediately. □

From Lemma 4, it follows that \( S_i \) can learn the values of \( z_r \), for every \( r \neq i \).

5.3.2 | Completing the attack

Armed with the values of \( z_r, 1 \leq r \leq \ell \), for every \( r \neq i \), the attacks described in Sections 5.2.2 and 5.2.3 now work exactly as previously described.

6 | OTHER OBSERVATIONS

6.1 | But what about the security analyses?

On the face of it, the above analyses are rather surprising in the light of apparently robust claims made in the Harn–Hsu [11], Harn–Gong [12], and Albakri–Harn [13] papers. In all three schemes, there are apparently ‘theorems’ proving the security of the respective schemes. However, in every case closer examination reveals that the ‘proofs’ are not in any way rigorous arguments.

For example, Theorem 1 of Albakri and Harn [13] states that ‘The adversaries cannot obtain any information of secret polynomials selected by KG’. The ‘secret polynomials’ referred to here are the polynomials in \( F \), and the set of possible adversaries includes parties with knowledge of one or more tokens \( T_i \). However, we have shown in Section 5.2 that knowledge of a single token is sufficient to learn all but one of the polynomials in \( F \) up to a constant term. How can this be, given Theorem 1? Examination of the ‘proof’ of Theorem 1 reveals that it is by no means a rigorous proof—it is more a series of unsubstantiated claims. For example, the proof starts with the following statement ‘Capturing one sensor—It is obvious that by capturing any single sensor \( S_i \), and obtaining the token \( T_i \), the adversary cannot recover information of any individual polynomial \( f_{\ell} \), nor the product of all individual polynomials’. That is, the ‘proof’ of the main claim seems to amount to a statement that it is ‘obvious’.

Sadly, the claimed result is clearly not as obvious as the authors hoped.

More seriously, this highlights the need for newly proposed cryptographic protocols to be provided with robust and rigorous evidence of their security. Indeed, this has been the
state of the art for a couple of decades, as has been very widely discussed—see, e.g. Boyd et al. [7].

6.2 | Three almost identical schemes

Quite apart from the lack of security, it is unfortunate that, given all three schemes we considered [11–13] share one author, three such similar schemes have been published separately. Moreover, the authors make no attempt to explain the close relationships between the three schemes.

6.3 | An application

To make matters worse, some authors have sought to build broader security schemes on top of one of the schemes considered here. For example, Harn et al. [14] describe a secure routing protocol for WSNs, of which the Harn–Hsu scheme [11] forms an integral part; indeed, for some reason the authors have chosen to describe the Harn–Hsu scheme again in some detail. This, of course, means that the routing protocol, regardless of its design, is inherently insecure. It would, of course, have been good design practice to describe the routing protocol in terms of a ‘black box’ technique for key pre-distribution and then to mention possible candidates for this black box, since there is no inherent reason to couple the two techniques. This would have avoided the main problem.

7 | SUMMARY AND CONCLUSIONS

As we have demonstrated, the Harn–Hsu, Harn–Gong and Albakri–Harn schemes all possess fundamental flaws (as does the derived Cheng–Hsu–Xia–Harn scheme). Given the nature of these flaws, it is difficult to imagine how the schemes could be rescued. Indeed, there is no reason to believe that a secure scheme can be designed using the underlying approach adopted in all three cases. As observed in Section 1, many related polynomial-based group key distribution schemes have been shown to be flawed [2–6]. Again as observed above, there are many existing schemes which achieve the same objectives in an efficient way and which have rigorous proofs of security—see, e.g. Boyd et al. [7] and Blundo et al. [1].

Fundamentally, the fact that the authors have not provided rigorous proofs of security for the various schemes means that attacks such as those described here remain possible. It would have been more prudent to follow established wisdom and only publish a scheme of this type if a rigorous security proof had been established. Similar remarks apply to the all-too-often misconceived attempts to fix broken schemes, unless a proof of security can be devised for a revised scheme. Achieving this seems very unlikely for variants of the schemes we have examined.

ORCID
Chris J Mitchell  https://orcid.org/0000-0002-6118-0055

REFERENCES
1. Blundo, C., Santis, A.D., Herzberg, A., Kutten, S., Vaccaro, U., Yang, M.: Perfectly secure key distribution for dynamic conferences. Inf. Comput. 146(1), 1–23 (1998). https://doi.org/10.1006/inco.1998.2717
2. Liu, J., Wu, Y., Liu, X., Zhang, Y., Xue, G., Zhou, W., et al.: On the (in) security of recent group key establishment protocols. Comput. J. 60, 507–526 (2017)
3. Mitchell, C.J.: The Hsu–Harn–Mu–Zhang–Zhu group key establishment protocol is insecure. arXiv:1803.05365 [cs.CY] (2018). http://arxiv.org/abs/1803.05365
4. Mitchell, C.J.: The Saeed–Liu–Tian–Gao–Li authenticated key agreement protocol is insecure. arXiv:1906.09330 [cs.CY] (2019). http://arxiv.org/abs/1906.09330
5. Mitchell, C.J.: Security issues in a group key establishment protocol. Comput. J. 62(3), 373–376 (2019). https://doi.org/10.1093/comjnl/bxy058
6. Mitchell, C.J.: Yet another insecure group key distribution scheme using secret sharing. arXiv:2003.14356 [cs.CY] (2020). https://arxiv.org/abs/2003.14356
7. Boyd, C., Mathuria, A., Stelia, D.: Protocols for Authentication and Key Establishment. In: Information Security and Cryptography, 2nd ed., Springer, Berlin (2020). https://doi.org/10.1007/978-3-662-58146-9
8. Cheng, Q., Hsu, C., Xia, Z., Harn, L.: Fast multivariate-polynomial-based membership authentication and key establishment for secure group communications in WSN. IEEE Access. 8, 71833–71839 (2020). https://doi.org/10.1109/ACCESS.2020.2987978
9. Rivest, R.L., Shamir, A., Adleman, L.: A method for obtaining digital signatures and public-key cryptosystems. Commun. ACM. 21, 120–126 (1978)
10. Menezes, A.J., van Oorschot, P.C., Vanstone, S.A.: Handbook of Applied Cryptography. CRC Press, Boca Raton (1997)
11. Harn, L., Hsu, C.F.: Pre-distribution scheme for establishing group keys in wireless sensor networks. IEEE Sens. J. 15(9), 5103–5108 (2015)
12. Harn, L., Gong, G.: Conference key establishment protocol using a multivariate polynomial and its applications. Secur. Commun. Netw. 8(9), 1794–1800 (2015). https://doi.org/10.1002/sec.1143
13. Albakri, A., Harn, L.: Non-interactive group key pre-distribution scheme (GKPS) for end-to-end routing in wireless sensor networks. IEEE Access. 7, 31615–31623 (2019). https://doi.org/10.1109/ACCESS.2019.2900390
14. Harn, L., Hsu, C.F., Ruan, O., Zhang, M.Y.: Novel design of secure end-to-end routing protocol in wireless sensor networks. IEEE Sens. J. 16(6), 1779–1785 (2016)

How to cite this article: Mitchell CJ. How not to secure wireless sensor networks: A plethora of insecure polynomial-based key pre-distribution schemes. IET Inf. Secur. 2021;15:223–230. https://doi.org/10.1049/ise2.12016