CS 798

Assignment 4

due 11:30 AM, Tuesday, December **3**

1. (a) Suppose there exists a PHF$(N; n, m, w)$. By constructing a suitable incidence matrix, prove that there exists a $(1, w-1)$-CFF(Nm, n).

   Answer:

   Let F be the transpose of the matrix which represents the PHF$(N; n, m, w)$. F is an N by n matrix, whose entries come from the set $\{1, \ldots, m\}$, say.

   We construct M, an Nm by n incidence matrix in which the rows are labelled by the elements of $\{1, \ldots, N\} \times \{1, \ldots, m\}$. The entry in row (r, x) and column c, which we denote as $m_{(r,x),c}$, is defined as follows:

$$m_{(r,x),c} = \begin{cases} 1 & \text{if } F(r,c) = x \\ 0 & \text{otherwise.} \end{cases}$$

   We claim that M is the incidence matrix of a $(1, w-1)$-CFF(Nm, n). Clearly there are Nm points and n blocks, so we need only prove the cover-free property.

   Let B be a block, and let $B_1, \ldots, B_{w-1}$ be $w-1$ other blocks. These w blocks correspond to w columns, say c and $c_1, \ldots, c_{w-1}$. There exists a row of F, say row r, in which the entries in these w columns are all different. Denote $x = F(r, c)$. Then $m_{(r,x),c} = 1$ and $m_{(r,y),c} = 0$ for all $y \neq x$. Therefore the point (r, x) is in the block B, but not in any of the blocks $B_1, \ldots, B_{w-1}$, and hence we have a $(1, w-1)$-CFF.

   (b) The following array is the *transpose* of a PHF$(4; 9, 3, 3)$:

0	0	0	1	1	1	2	2	2
0	1	2	0	1	2	0	1	2
0	1	2	1	2	0	2	0	1
0	1	2	2	0	1	1	2	0

   Construct a $(1, 2)$-CFF$(12, 9)$ from this PHF using the technique you described above. Present the CFF both in the form of an incidence matrix and in the form of a set system.

Answer:

The incidence matrix is as follows:

$$\left(\begin{array}{ccccccccc}
1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\
\hline
1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\
0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\
0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\
\hline
1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\
0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\
\hline
1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\
0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0
\end{array}\right)$$

The set system is as follows:

$$\begin{aligned}
X &= \{1, \ldots, 12\}, \\
\mathcal{A} &= \{\{1,4,7,10\}, \{1,5,8,11\}, \{1,6,9,12\}, \\
&\quad \{2,4,8,12\}, \{2,5,9,10\}, \{2,6,7,11\} \\
&\quad \{3,4,9,11\}, \{3,5,7,12\}, \{3,6,8,10\}\}.
\end{aligned}$$

2. PHF can be used for broadcast encryption. The following description summarizes how this can be done.

step 1. Given a PHF$(N; n, m, w)$, construct an Nm by n incidence matrix, M (this is similar to question 1).

step 2. Use M to set up Nm Fiat-Naor 1-KDPs, as described in the "general" BES. Denote these schemes as $\mathcal{F}_{i,j}$, $1 \le i \le N$, $1 \le j \le m$.

step 3. Split the secret $K \in \mathbb{Z}_p$ into N shares, using the (N, N) threshold scheme in which the secret is the modulo p sum of the shares. Denote these shares as $s_1, \ldots, s_N$.

step 4. Let P be the subset to which K is being broadcast. For all i, j, let $k_{i,j}$ be the group key for $P \cap users(\mathcal{F}_{i,j})$. Use $k_{i,j}$ to encrypt s_i, for all i, j.

step 5. Broadcast the Nm encryptions of the shares.

(a) Describe how each user in P can decrypt the broadcast, finding K.

Answer: Suppose that user $U_c \in P$ (this user corresponds to column c of the incidence matrix M). For all i, $1 \le i \le N$, there exists a j_i such that user U_c is in the scheme $\mathcal{F}_{i,j_i}$) (to be specific, $j_i = F(i, c)$, where F is the matrix of the transposed PHF). Since U_c is in the scheme $\mathcal{F}_{i,j_i}$, this means that U_c can decrypt the share s_i using the group key for the set of users $P \cap users(i, j_i)$, for $1 \le i \le n$. Therefore U_c can decrypt all N shares, and reconstruct the secret K.

(b) Explain why the scheme is secure against coalitions of size w.

Answer:

Consider any coalition C of size w. The users in C correspond to w columns of the transposed PHF matrix, F, say columns $c_1, \ldots, c_w$. There exists a row of F, say row r, in which the entries $F(r, c_1), \ldots, F(r, c_w)$ are all different. Therefore, $|C \cap users(r, j)| \leq 1$ for $1 \leq j \leq m$. This means that C cannot compute any of the group keys used to encrypt the share s_r. Since all N shares are required to decrypt K, it follows that C cannot decrypt K.

(c) Given the PHF from question 1, describe in detail how the scheme is set up (and list the keys held by each user); how the group keys would be constructed for the set $P = \{1, 2, 3, 4\}$; and how the broadcast is formed.

Answer:

In the set-up phase, a total of twelve keys will be given to each user, as indicated below:

U_1	U_2	U_3	U_4	U_5	U_6	U_7	U_8	U_9
$\ell_{1,1}$	$\ell_{1,1}$	$\ell_{1,1}$	$\ell_{1,2}$	$\ell_{1,2}$	$\ell_{1,2}$	$\ell_{1,3}$	$\ell_{1,3}$	$\ell_{1,3}$
$\ell_{1,1,2}$	$\ell_{1,1,1}$	$\ell_{1,1,1}$	$\ell_{1,2,5}$	$\ell_{1,2,4}$	$\ell_{1,2,4}$	$\ell_{1,3,8}$	$\ell_{1,3,7}$	$\ell_{1,3,7}$
$\ell_{1,1,3}$	$\ell_{1,1,3}$	$\ell_{1,1,2}$	$\ell_{1,2,6}$	$\ell_{1,2,6}$	$\ell_{1,2,5}$	$\ell_{1,3,9}$	$\ell_{1,3,9}$	$\ell_{1,3,8}$
$\ell_{2,1}$	$\ell_{2,2}$	$\ell_{2,3}$	$\ell_{2,1}$	$\ell_{2,2}$	$\ell_{2,3}$	$\ell_{2,1}$	$\ell_{2,2}$	$\ell_{2,3}$
$\ell_{2,1,4}$	$\ell_{2,2,5}$	$\ell_{2,3,6}$	$\ell_{2,1,1}$	$\ell_{2,2,2}$	$\ell_{2,3,3}$	$\ell_{2,1,1}$	$\ell_{2,2,2}$	$\ell_{2,3,3}$
$\ell_{2,1,7}$	$\ell_{2,2,8}$	$\ell_{2,3,9}$	$\ell_{2,1,7}$	$\ell_{2,2,8}$	$\ell_{2,3,9}$	$\ell_{2,1,4}$	$\ell_{2,2,5}$	$\ell_{2,3,6}$
$\ell_{3,1}$	$\ell_{3,2}$	$\ell_{3,3}$	$\ell_{3,2}$	$\ell_{3,3}$	$\ell_{3,1}$	$\ell_{3,3}$	$\ell_{3,1}$	$\ell_{3,2}$
$\ell_{3,1,6}$	$\ell_{3,2,4}$	$\ell_{3,3,5}$	$\ell_{3,2,2}$	$\ell_{3,3,3}$	$\ell_{3,1,1}$	$\ell_{3,3,3}$	$\ell_{3,1,1}$	$\ell_{3,2,2}$
$\ell_{3,1,8}$	$\ell_{3,2,9}$	$\ell_{3,3,7}$	$\ell_{3,2,9}$	$\ell_{3,3,7}$	$\ell_{3,1,8}$	$\ell_{3,3,5}$	$\ell_{3,1,6}$	$\ell_{3,2,4}$
$\ell_{4,1}$	$\ell_{4,2}$	$\ell_{4,3}$	$\ell_{4,3}$	$\ell_{4,1}$	$\ell_{4,2}$	$\ell_{4,2}$	$\ell_{4,3}$	$\ell_{4,1}$
$\ell_{4,1,5}$	$\ell_{4,2,6}$	$\ell_{4,3,4}$	$\ell_{4,3,3}$	$\ell_{4,1,1}$	$\ell_{4,2,2}$	$\ell_{4,2,2}$	$\ell_{4,3,3}$	$\ell_{4,1,1}$
$\ell_{4,1,9}$	$\ell_{4,2,7}$	$\ell_{4,3,8}$	$\ell_{4,3,8}$	$\ell_{4,1,9}$	$\ell_{4,2,7}$	$\ell_{4,2,6}$	$\ell_{4,3,4}$	$\ell_{4,1,5}$

The following will be the keys used in the twelve Fiat-Naor schemes:

(i,j)	$users(i,j)$	$users(i,j) \cap P$	$k_{(i,j)}$
$(1,1)$	$\{U_1, U_2, U_3\}$	$\{U_1, U_2, U_3\}$	$\ell_{1,1}$
$(1,2)$	$\{U_4, U_5, U_6\}$	$\{U_4\}$	$\ell_{1,2} + \ell_{1,2,5} + \ell_{1,2,6}$
$(1,3)$	$\{U_7, U_8, U_9\}$	$\emptyset$	$\ell_{1,3} + \ell_{1,3,7} + \ell_{1,3,8} + \ell_{1,3,9}$
$(2,1)$	$\{U_1, U_4, U_7\}$	$\{U_1, U_4\}$	$\ell_{2,1} + \ell_{2,1,7}$
$(2,2)$	$\{U_2, U_5, U_8\}$	$\{U_2\}$	$\ell_{2,2} + \ell_{2,2,5} + \ell_{2,2,8}$
$(2,3)$	$\{U_3, U_6, U_9\}$	$\{U_3\}$	$\ell_{2,3} + \ell_{2,3,6} + \ell_{2,3,9}$
$(3,1)$	$\{U_1, U_6, U_8\}$	$\{U_1\}$	$\ell_{3,1} + \ell_{3,1,6} + \ell_{3,1,8}$
$(3,2)$	$\{U_2, U_4, U_9\}$	$\{U_2, U_4\}$	$\ell_{3,2} + \ell_{3,2,9}$
$(3,3)$	$\{U_3, U_5, U_7\}$	$\{U_3\}$	$\ell_{3,3} + \ell_{3,3,5} + \ell_{3,3,7}$
$(4,1)$	$\{U_1, U_5, U_9\}$	$\{U_1\}$	$\ell_{4,1} + \ell_{4,1,5} + \ell_{4,1,9}$
$(4,2)$	$\{U_2, U_6, U_7\}$	$\{U_2\}$	$\ell_{4,2} + \ell_{4,2,6} + \ell_{4,2,7}$
$(4,3)$	$\{U_3, U_4, U_8\}$	$\{U_3, U_4\}$	$\ell_{4,3} + \ell_{4,3,8}$

The broadcast consists of

$$\begin{array}{ccc} e_{k_{(1,1)}}(s_1) & e_{k_{(1,2)}}(s_1) & e_{k_{(1,3)}}(s_1) \\ e_{k_{(2,1)}}(s_2) & e_{k_{(2,2)}}(s_2) & e_{k_{(2,3)}}(s_2) \\ e_{k_{(3,1)}}(s_3) & e_{k_{(3,2)}}(s_3) & e_{k_{(3,3)}}(s_3) \\ e_{k_{(4,1)}}(s_4) & e_{k_{(4,2)}}(s_4) & e_{k_{(4,3)}}(s_4) \end{array}$$

3. (a) Suppose that the following are the nine shares in a $(6, 9)$ Shamir threshold scheme implemented in $\mathbb{Z}_{1993}$:

i	x_i	y_i
1	1	187
2	2	1547
3	3	498
4	4	1407
5	5	1564
6	6	1176
7	7	795
8	8	185
9	9	603

Exactly one of these shares is defective (i.e., incorrect). Your task is to determine which share is defective, and then figure out its correct value, as well as the value of the secret. The "primitive operations" in your algorithm are polynomial interpolations and polynomial evaluations. Try to minimize the number of polynomial interpolations you perform. Hint: The question can be solved using only three polynomial interpolations.

Answer:

If we interpolate shares $1, \ldots, 6$, then we obtain the polynomial

$$f_1(x) = 889 + 1017x + 994x^2 + 1455x^3 + 698x^4 + 1113x^5.$$

However, $f_1(7) = 1403$, $f_1(8) = 401$ and $f_1(9) = 1323$, so f_1 is not the correct polynomial. If we interpolate shares $1, 2, 3, 7, 8, 9$, then we obtain the polynomial

$$f_2(x) = 1556 + 117x + 599x^2 + 471x^3 + 1392x^4 + 38x^5.$$

However, $f_2(4) = 549$, $f_2(5) = 479$ and $f_2(6) = 890$, so f_2 is not the correct polynomial. If we interpolate shares $4, 5, 6, 7, 8, 9$, then we obtain the polynomial

$$f_3(x) = 371 + 54x + 789x^2 + 1002x^3 + 1556x^4 + 401x^5.$$

Then, $f_3(1) = 187$, $f_3(2) = 1547$ and $f_3(3) = 1070$, so the third share is incorrect and f_3 is the correct polynomial. The secret is 371.

Optional: The method used was guaranteed to find the correct polynomial. One of $f_1(x), f_2(x), f_3(x)$ is the correct polynomial, say $f_i(x)$. $f_i(x)$ will agree with eight of the nine shares. However, if $j \neq i$, then $f_j(x)$ will agree with at most *six* shares. ($f_j(x)$ agrees with at most five of the eight good shares, because six good shares interpolate to a unique polynomial, namely, $f_i(x)$. Hence $f_j(x)$ agrees with at most six shares in total.) Therefore the correct polynomial is the unique polynomial, among the three polynomials tested, that agrees with eight of the nine shares.

(b) Suppose that a (k, n) Shamir threshold scheme has exactly one defective share, and suppose that $n \geq 2k$. Describe how it is possible to determine which share is defective using only two polynomial interpolations.

Answer: Interpolate the first k shares, to obtain a polynomial $f_1(x)$. Then check to see if at most one of the remaining $n - k$ shares does not lie on the polynomial f_1. If so, then f_1 is the correct polynomial, K is the constant term of f_1, and the algorithm terminates. Otherwise, interpolate the next set of k shares (shares $k+1, \ldots, 2k$) to obain a polynomial f_2. Under the given hypotheses, f_2 must be the correct polynomial and K is the constant term of f_2.

We now give a brief proof sketch that the algorithm described above will yield the correct value of the secret. Let $f(x)$ denote the polynomial used to generate the shares. We observe the following:

- If $f = f_i$ $(i = 1, 2)$, then at most one of the n shares will not lie on f_i.
- If $f \neq f_i$ $(i = 1, 2)$, then at least $n - k$ of the n shares will not lie on f_i.
- If $n \geq k + 2$, then there exists a value i $(i = 1, 2)$ such that at most one of the n shares does not lie on f_i, and $f = f_i$ for any such i.
- $n \geq 2k$ and $k \geq 2$, so $n \geq k + 2$, as desired.

(c) More generally, suppose that a (k, n) Shamir threshold scheme has exactly τ defective shares, and suppose that $n \geq (\tau + 1)k$. Describe how it is possible to determine which shares are defective using only $\tau + 1$ polynomial interpolations.

Answer:

For $i \leftarrow 1$ to $\tau + 1$, perform the following steps:

(1) interpolate shares $(i-1)k+1, \ldots, ik$ to yield a polynomial $f_i(x)$

(2) determine the set $B_i = \{j : f_i(x_j) \neq y_j\}$

(3) if $|B_i| \leq \tau$, then K is the constant term of f_i

We now give a brief proof sketch that the algorithm described above will yield the correct value of the secret. Let $f(x)$ denote the polynomial used to generate the shares. We observe the following:

- If $f = f_i$ $(i = 1, 2, \cdots, \tau + 1)$, then at most τ of the n shares will not lie on f_i.
- If $f \neq f_i$ $(i = 1, 2, \cdots, \tau + 1)$, then at least $n - k - \tau + 1$ of the n shares will not lie on f_i.
- If $n \geq k + 2\tau$, then there exists a value i $(i = 1, 2)$ such that at most τ of the n shares does not lie on f_i, and $f = f_i$ for any such i.
- $n \geq (\tau + 1)k$ and $k \geq 2$, so $n \geq k + 2\tau$, as desired.

4. Suppose we want to revoke r users, say $U_{i_1}, \ldots, U_{i_r}$, at the same time in the LKH scheme. Assuming that the tree depth d and the nodes are labelled as described in class, we can assume that $2^d \leq U_{i_1} < \cdots < U_{i_r} \leq 2^{d+1} - 1$.

(a) Informally describe an algorithm that can be used to determine which keys in the tree need to be updated.

Answer: For a node that is not a leaf node, let $\mathsf{left}(\cdot)$ and $\mathsf{right}(\cdot)$ denote the left and right children (respectively) of the given node.

Define a set $\mathcal{D}$ recursively, as follows:

$$U_{i_j} \in \mathcal{D}, 1 \leq j \leq r$$
$$\text{for all non-leaf nodes } X, \mathsf{left}(X), \mathsf{right}(X) \in \mathcal{D} \Rightarrow X \in \mathcal{D}.$$

The set $\mathcal{D}$ contains all nodes in the tree for which keys are no longer required. It can easily computed in a bottom-up fashion, as follows:

(1) $\mathcal{D} \leftarrow \{U_{i_j} : 1 \leq j \leq r\}$
(2) for $X \leftarrow 2^d - 1$ **downto** 1
  do if $\{2X, 2X+1\} \subseteq \mathcal{D}$
  then $\mathcal{D} \leftarrow \mathcal{D} \cup \{X\}$

The keys to be updated are those in the set

$$\mathcal{Q} = \left(\bigcup_{j=1}^{r} \mathcal{P}(U_{i_j}) \right) \setminus \mathcal{D},$$

where, for any leaf node U, $\mathcal{P}(U)$ denotes the set of nodes in the unique path from the node U to the root node R.

(b) Describe the broadcast that is used to update the keys. Which keys are used to encrypt the new, updated keys?

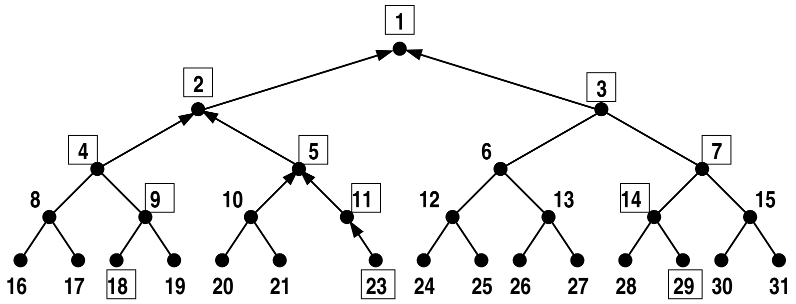
Answer: Define $\mathcal{Q}$ and $\mathcal{D}$ as above. For all nodes $X \in \mathcal{Q}$ and for $Y \in \{\mathsf{left}(X), \mathsf{right}(X)\}$, the following item is broadcast by the *TA*:

$$\begin{array}{ll} e_{k'(Y)}(k'(X)) & \text{if } Y \in \mathcal{Q} \\ e_{k(Y)}(k'(X)) & \text{if } Y \notin \mathcal{D} \cup \mathcal{Q} \\ \emptyset & \text{if } Y \in \mathcal{D} \end{array}$$

(c) Illustrate (a) and (b) by describing the updated keys and the broadcast if users 18, 23 and 29 are to be revoked in a tree with depth $d = 4$. How much smaller is the broadcast in this case, as compared to the three broadcasts that would be required to revoke these three users one at a time?

Answer: Here, $\mathcal{Q} = \{1, 2, 3, 4, 5, 7, 9, 11, 14\}$ and $\mathcal{D} = \{18, 23, 29\}$.

The nodes in $\mathcal{Q} \cup \mathcal{D}$ are boxed in the following diagram:



The following items are broadcast:

$$\begin{array}{ll} e_{k'(2)}(k'(1)) & e_{k'(3)}(k'(1)) \\ e_{k'(4)}(k'(2)) & e_{k'(5)}(k'(2)) \\ e_{k(6)}(k'(3)) & e_{k'(7)}(k'(3)) \\ e_{k(8)}(k'(4)) & e_{k'(9)}(k'(4)) \\ e_{k(10)}(k'(5)) & e_{k'(11)}(k'(5)) \\ e_{k'(14)}(k'(7)) & e_{k(15)}(k'(7)) \\ e_{k(19)}(k'(9)) & \\ e_{k(22)}(k'(11)) & \\ e_{k(28)}(k'(14)) & \end{array}$$

The size of the broadcast is reduced from $3 \times 7 = 21$ to 15 in this example.

5. (a) Consider the $(3, 3r^2, r^2+2r)$ 2-IPP code $\mathcal{C}$ that was described in class. Give a complete description of a $O(1)$ time algorithm TRACE, which takes as input a triple $\mathbf{f} = (f_1, f_2, f_3)$ and attempts to determine an identifiable parent of $\mathbf{f}$. If $\mathbf{f} \in \mathcal{C}$, then TRACE$(\mathbf{f}) = \mathbf{f}$; if $\mathbf{f} \in \mathsf{desc}_2(\mathcal{C}) \backslash \mathcal{C}$, then TRACE$(\mathbf{f})$ should find an identifiable parent of $\mathbf{f}$; and TRACE$(\mathbf{f})$ should return the output "fail", if $\mathbf{f} \notin \mathsf{desc}_2(\mathcal{C})$.

   Answer:

   Let (x_1, x_2, x_3) be the input to the tracing algorithm. If $(x_1, x_2, x_3) \in \mathsf{desc}_2(\mathcal{C})$, then there are six cases that can occur:

   (1) If $x_1 \in L$, compute $m_2 \in M$ such that $m_2 \equiv x_1 \pmod n$. Then compute $m_1 = (x_1 + r^2 - m_2)/r$. The codeword (x_1, m_1, m_2) is a parent of (x_1, x_2, x_3).

   (2) If $x_2 \in L$, compute $m \in M$ such that $m \equiv x_2 \pmod n$. Then compute $s = (x_2 - m)/r$. The codeword (m, x_2, s) is a parent of (x_1, x_2, x_3).

   (3) If $x_3 \in L$, compute $s_2 \in S$ such that $s_2 \equiv x_3 \pmod n$. Then compute $s_1 = (x_3 - s_2 - r)/r$. The codeword (s_1, s_2, x_3) is a parent of (x_1, x_2, x_3).

   (4) If $x_1, x_2 \in S$, let $s_1 = x_1$ and $s_2 = x_2$. The codeword $(s_1, s_2, rs_1 + s_2 + r) = (x_1, x_2, rx_1 + x_2 + r)$ is a parent of (x_1, x_2, x_3).

   (5) If $x_2, x_3 \in M$, let $m_1 = x_2$ and $m_2 = x_3$. The codeword $(rm_1 + m_2 - r^2, m_1, m_2) = (rx_2 + x_3 - r^2, x_2, x_3)$ is a parent of (x_1, x_2, x_3).

   (6) If $x_1 \in M$ and $x_3 \in S$, let $m = x_1$ and $s = x_3$. The codeword $(m, sr + m, s) = (x_1, rx_3 + x_1, x_3)$ is a parent of (x_1, x_2, x_3).

   If none of these six cases arise, then $(x_1, x_2, x_3) \notin \mathsf{desc}_2(\mathcal{C})$ and the tracing algorithm fails.

   (b) Illustrate the execution of your algorithm in the case $r = 10$ ($q = 120$) for the following triples: $(13, 11, 17)$; $(44, 9, 14)$; $(18, 108, 9)$.

   Answer: When $(x_1, x_2, x_3) = (13, 11, 17)$, we have $x_2, x_3 \in M$, so we are in case (5). $10 \times 11 + 17 - 10^2 = 27$, and hence $(27, 11, 17)$ is a parent of $(13, 11, 17)$.

   When $(x_1, x_2, x_3) = (44, 9, 14)$, we have $x_1 \in L$, so we are in case (1). We have $m_2 = 14$ and $m_1 = (44 + 100 - 14)/10 = 13$, and hence $(44, 13, 14)$ is a parent of $(44, 9, 14)$.

   When $(x_1, x_2, x_3) = (18, 108, 9)$, we have $x_2 \in L$, so we are in case (2). We have $m = 18$ and $s = (108 - 18)/10 = 9$, and hence $(18, 108, 9)$ is a parent of $(18, 108, 9)$. (In this example, (x_1, x_2, x_3) is already a codeword.)

Note: You can write a computer program to answer this question, but it is not required. If you do not write a program, give a detailed pseudo-code description of your algorithm.