LOTR

Cryptanalysis, 936 points

$k = 1024$
$b = 2k + 128$
$e = 0x10001$

This problem instantiates an insecure version of the *ring signature* scheme presented here. The goal of the task is to forge a signature even though you don't have access to any of the signing keys.

In the Keygen phase, 243 pairs of RSA public and private keys are generated (N_i, d_i)(the public exponent e is the same for all users). To generate a valid signature corresponding to the challenge message $m =$ FAKE NEWS, an attacker must be able to produce $2^{b-1} + 2^{2k} < x_i < 2^b - 2^{2k}$ for $i \in [0, 242]$, such that

$$\bigoplus_{i=0}^{242} \mathsf{RSA}_{\mathsf{permutation}}(x_i) = \mathsf{sha256}(m) \bmod 2^{256}$$

.

The function $\mathsf{RSA}_{\mathsf{permutation}}()$ is used to transform the usual RSA encryption/decryption, that acts on all the numbers from $[0, N-1]$, into a permutation that acts on any integer whose binary representation has length at most b. This ensures that any user that holds a secret key d is always able to produce a valid signature.

Using the notations: $y_i := \mathsf{RSA}_{\mathsf{permutation}}(x_i)$ and $z_i := y_i \bmod 2^{256}$, notice that is enough for the attacker to produce the x_i's that satisfy

$$\bigoplus_{i=0}^{242} z_i = \mathsf{sha256}(m)$$

Observe that it is really easy to find a subset $W \subset \{0, 1, 2 \ldots, 242\}$ such that $\bigoplus_{i \in W} z_i = \mathsf{sha256}(m)$. To do this just consider the matrix $\mathbf{Z} \in GF(2)^{256 \times 243}$ whose columns are given by the binary representation of the z_i's. To find the subset W it's enough to solve the system $\mathbf{Z} \cdot \mathbf{w} = \mathsf{sha256}(m)$, for $\mathbf{w} \in GF(2)^{243}$. The above system has a solution with probability roughly 2^{-13} when the x_i are uniformly random. So we just generate enough matrices until the system has a solution.

Unfortunately it is not enough to find a subset of values that xor to the hash of the message. The problem asks that the xor of *all* the values is equal to the hash of the message. In order to do this we can use a similar linear algebra trick as follows:

- we generate uniformly random values x_i, x_i' for $i \in [0, 242]$ and generate the corresponding matrices $\mathbf{Z} \in GF(2)^{256\times 243}$, $\mathbf{Z}' \in GF(2)^{256\times 243}$ as we did before.

- we can use linear algebra to find $\mathbf{w} \in GF(2)^{243}$ such that

$$(\mathbf{Z}+\mathbf{Z}')\mathbf{w} = \mathsf{sha256}(m) + \mathbf{Z}\cdot\mathbf{1}, \text{where } \mathbf{1} = (1, 1, \ldots, 1)^\top \in GF(2)^{256}$$

As before, the system has a solution with probability roughly 2^{-13}. So if we repeat this enough times we end up with a solvable linear system.

Notice that the above system is equivalent to $\mathbf{Z}(\mathbf{1}-\mathbf{w}) + \mathbf{Z}'\mathbf{w} = \mathsf{sha256}(m)$, over $GF(2)$. From this we can easily pick 243 vectors that xor to the hash of the message: if $\mathbf{w}[i] = 0$ pick z_i , else pick z_i'. Written in a more compact way, a valid signature is given by the values $(1-w[i])\cdot x_i + w[i]\cdot x_i'$ for $i \in [0, 242]$. Since all the x_i and x_i''s were uniformly sampled, they satisfy the verification bounds with overwhelming probability.