BabyTimeCapsule

Organization: HackTheBox	Type: online challenge	
Categories: □ Network Security ☑ Cryptography □ Mobile Applications	□ Reverse Engineering □ Web Applications □ Forensics	Difficulty: Easy
Name: Kasper Verhulst	Release date: 16-07-2022 Completing date: 26-12-2022	

1 Understanding the challange

The challenges involves the encryption of the flag with a textbook RSA implementation. An encryption server will listen on a TCP connection. We can establish a TCP conncetion using netcat:

```
    nc localhost 1337
Welcome to Qubit Enterprises. Would you like your own time capsule? (Y/n) Y
{”time_capsule”: ”75FD896DAC110FF4E5B78892EDC7FC7967DDA71E9DF283382D081AD16A1ED239BAE927E693A5
”pubkey”: [”5350F28DD6B164EB7DD56EABBA123CC13A6ADA93C4BC61861E9E29653AEC438786A0C8624790194D
”5”]}
Welcome to Qubit Enterprises. Would you like your own time capsule? (Y/n) Y
{”time_capsule”: ”75FD896DAC110FF4E5B78892EDC7FC7967DDA71E9DF283382D081AD16A1ED239BAE927E693A5
”pubkey”: [”90960896CCC0B2BE51ADDD62C76C0DA6C132EAB09A24292D9CD2B7EB05758E55A0485C25805341B3D
”5”]}
Welcome to Qubit Enterprises. Would you like your own time capsule? (Y/n) n
Thank you, take care
```

Each time we ask for it, the program will re-encrypt the flag. The public key (e,N) is outputted together with the ciphertext. Each iteration, p and q are randomly generated to construct N = p.q, but e=5 always stays the same

2 Solving the challenge

The RSA algorithm seems to be correctly implemented. Furthermore, you can try to break the prompt by not responding with Y(es) or n(o), but the program seems to be handling it correctly. However, there are some inherent vulnerabilities to textbook RSA, even when implemented correctly.

For instance, when encrypting with low encryption exponents (e.g., e = 3) and small values of the m (i.e., m ¡ n1/e), the result of me is strictly less than the modulus n. In this case, no modulus operation is needed and ciphertexts can be decrypted easily by taking the eth root of the ciphertext over the integers. Yet this CTF, the message is sufficiently long.

Another possible attack vector for textbook RSA is the so-called Coppersmith's attack. If the same clear-text message is sent to e or more recipients in an encrypted way, and the receivers share the same exponent e, but different p, q, and therefore n, then it is easy to decrypt the original clear-text message via the Chinese remainder theorem.

$$
\begin{aligned}
C_1 \equiv m^e \quad (mod N_1) \\
C_2 \equiv m^e \quad (mod N_2) \\
... \\
C_e \equiv m^e \quad (mod N_e)
\end{aligned}
\tag{1}
$$

By using the CRT we can find:

$$C \equiv m^e (mod N_1.N_2...N_e) \tag{2}$$

Ten, simply taking the e-th root:

$$m \equiv \sqrt[e]{C} \quad (mod N_1.N_2...N_e) \tag{3}$$