\*prerequisite - RSA,

If you’re not familiar with it, go read [this](https://www.geeksforgeeks.org/rsa-algorithm-cryptography/)

[You’re welcomed to read the original paper](https://archiv.infsec.ethz.ch/education/fs08/secsem/bleichenbacher98.pdf)

PCKS padding -

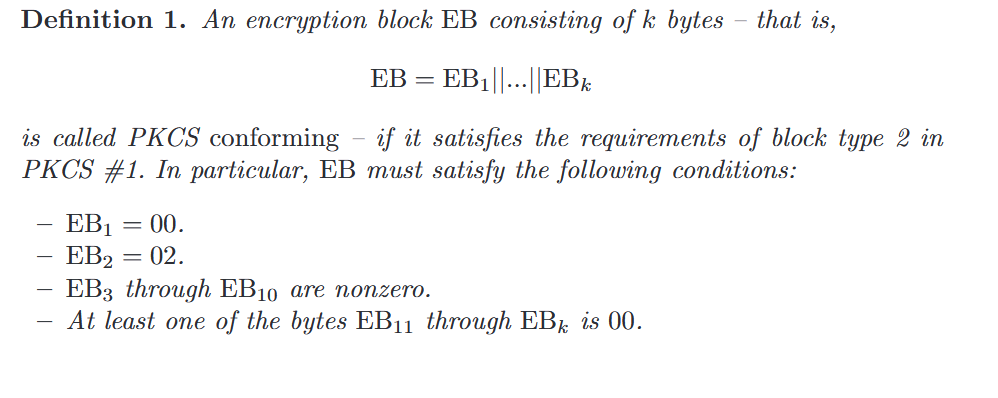
Throughout this CTF our RSA keys are of size 1024 bits.

Denote k as the length of n in bytes (i.e. if n 17 then k = 1, if n is 255 then k = 1, if n = 256 then k = 2), or formally speaking:

We assume our message is a number (e.g. “hello” has binary representation 0110100001100101011011000110110001101111 00001010 and this binary number translates to 114784820031242)  
  
We pad our messages in order to ensure proper block alignment and to enhance security.

On the original message (let’s denote || the number of its bytes), we define where are random non-zero bytes, and || = .

To de-pad the message, find the first null byte after the first bytes, and remove all bytes up to it.



The Bleichenbacher attack breaks RSA, under the assumption of an access to a (side) server (oracle) which leaks whether a message is PCKS conforming - i.e. it can decipher some rsa ciphered message if:

1. We’ve access to (not )

2. We can “spam” an oracle to check whether some message is PCKS conforming, in other words: and we can call .

How? We use the following 2 insights

1. If is then where . (infers from the first 2 bytes, [big endian](https://www.geeksforgeeks.org/little-and-big-endian-mystery/)).

2. (where we determine , so we can weaponize it by using the previous observation.

The main idea -

1. Maintain possible range values where is located, at first the range of values is [2\*B,3\*B-1]

2. By querying the oracle find some for which

3. Based on our finding of last step, narrow down the possible values (i.e. reduce the range)

4. If we’re left with 1 value it’s and we’re done.

Otherwise go back to step 1

You’ll find the attack formally expressed in the next page.

Be prepared, it’s not trivial;

Correctness and analysis are shown later.

They aren’t mandatory for completing the CTF, but they are pretty educational and will deepen your understanding.

:

Isn’t relevant for this CTF.

1. Set ,

\*

For each in :

For all in :

If :

return

Else

Goto step 2

as otherwise thus it can’t be PCKS conf’

We expect to to be PCKS conf’ in probability of as |[2B,3B-1]| = B

So we expect 1 of oracle calls to succeed, which is decent enough.

We prove by induction that

i=0 we’re done as is PCKS conf ‘

Assume for , we prove for :

;

As by induction .

Similarly we get .

From first equation (\*) we get:

So is indeed in one of the ranges.

Why the attack is fast enough stems from the exponential decreasing size of ranges. We’ll not show the proof (well it’s more of an approximation based on heuristic than a rigorous proof).