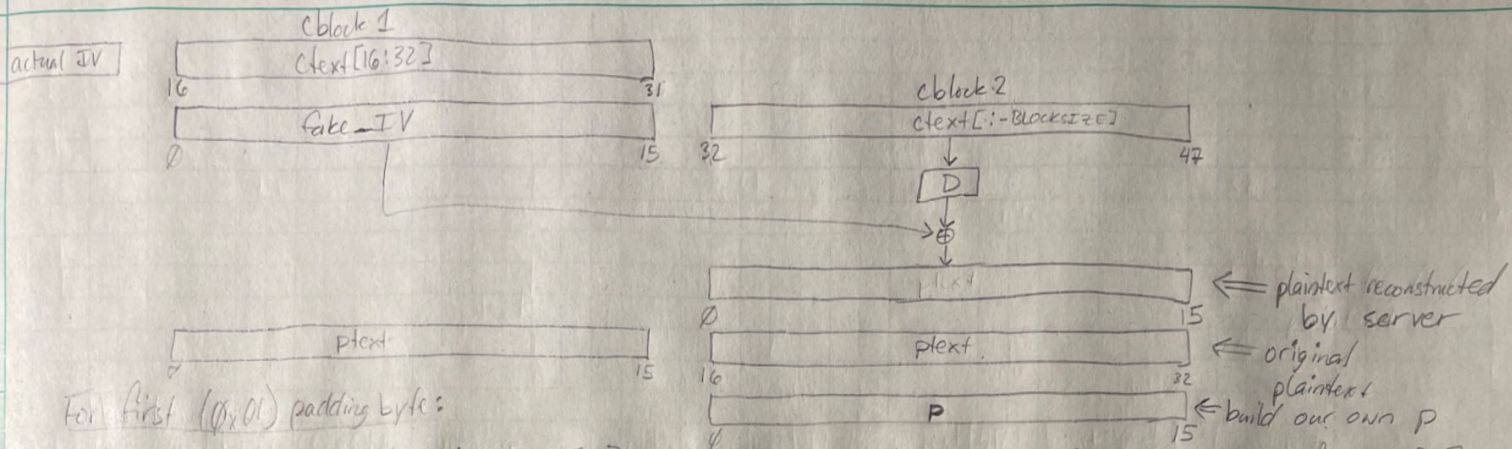

For first (0x01) padding byte:

When padding valid, fake_IV[15] ⊕ Decrypt(ctext[47]) = 0x01 ⇒ Decrypt(ctext[47]) = fake_IV[15] ⊕ 0x01

Since we know originally, ctext[31] ⊕ Decrypt(ctext[47]) = ptext[31],

we can now get ptext[31] = ctext[31] ⊕ fake_IV[15] ⊕ 0x01 ⇒ set as p[15]

To set up ctext for next round (i.e. set last byte of ctext to 0x02):

We want fake_IV[15] ⊕ D(ctext[47]) = 0x02 ⇒ p[15] ⊕ fake_IV[15] = D(ctext[47]) = ctext[31] ⊕ ~~fake_IV[15]~~ ⊕ 0x01 ⊕ fake_IV[15] (Cancel)

⇒ fake_IV[15] ⊕ (ctext[31] ⊕ 0x01) = 0x02

⇒ fake_IV[15] = ctext[31] ⊕ 0x01 ⊕ 0x02

For second (0x02) padding byte:

ptext[30] = ctext[30] ⊕ fake_IV[14] ⊕ 0x02 ⇒ p[14]

General Rule (for 1 block of ctext attempting to pad w/ "target" # bytes): p[16-target] = ctext[16-target] ⊕ fake_IV[16-target] ⊕ target

For prepping for 3rd round: We want fake_IV[15] ⊕ D(ctext[47]) = fake_IV[14] ⊕ D(ctext[46]) = 0x03

right now: fake_IV[15] ⊕ D(" ") = 0x02 ⇒ fake_IV[15] = D(" ") ⊕ 0x02 ⇒ we want fake_IV = fake_IV[15] ⊕ 0x02 ⊕ 0x03

⇒ (fake_IV ⊕ 0x02 ⊕ 0x03) ⊕ D(" ") = 0x02

⇒ fake_IV[15] ⊕ D(" ") = 0x02 ⊕ (0x02 ⊕ 0x03) = 0x03

General Rule: To prep for "target" padding, set fake_IV[x] = fake_IV[x] ⊕ (target - 1) ⊕ target