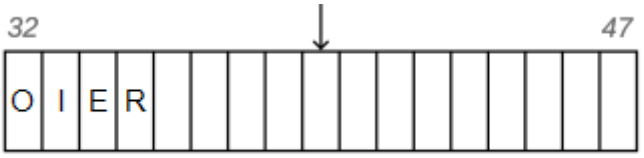
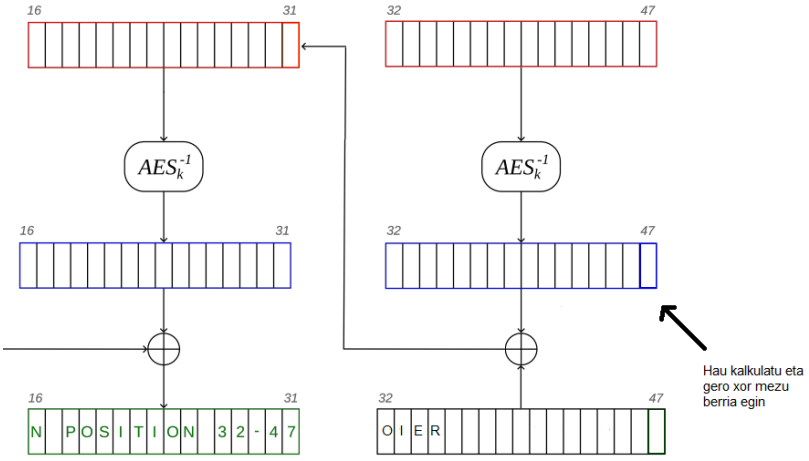
PADDING

Padding oracle praktikan mezu zifratu bat aldatu behar dugu gure izena azaltzeko mezu deszifratuaren bukaeran, mezu hau AES-CBC bitartez zifratuak izan dira.



Azken blokearen ciphertexta kalkulatzeko plaintext-a xor ciphertext-arekin (aurreko blokea) egiten da eta honek tarteko balio bat ematen digu non geroago aes_cbc zifratuz ciphertext-aren azken blokea lortzen dugu. Orduan plaintextaren azken blokea aldatu nahi baldin badugu azkeneko blokearen tarteko balioa lortu eta mezu berria xor tarteko balioa ciphertextaren aurreneko blokean egin beharko dugu.



Hau lortu dezakegu padding gehitzeko teknika oker bat erabiltzen denean, kasu honetan azkeneko blokea 16 bytekoa ez bada, falta zaizkion byte-ak betetzen ditu falta zaizkion byte kopurua ezarriz. Adibidez, 3 byte falta badira blokea bukatzeko, azkeneko 3 byteak auek izango dira: 0x03, 0x03 eta 0x03. Hau jakinda padding ezberdinak erabiliz intermediate bectorearen azkeneko zatia kalkulatu dezakegu gutxinaka. Hortaz hurrengo programa egin dugu.

Gure programan lehenengo for begizta (i begizta) zifratutako blokearen byte bakoitza aztertzen du (16tik 31ra), iterazio bakoitzean padding zuzena emanten duten byte balioak bilatzen dira. Bigarreneko begiztan (j begizta) ciphertext 31-i posizioan izan ditzakeen 255 balioak aztertzen ditugu eta iterazio bakoitzean decipher_AES_CBC funtzioari deituz padding-a zuzena den ala ez egiaztatzen dugu. Paddin-a zuzena denean orduan tarteko balioa kalkulatzen dugu (47-i posizioan) ciphertext[31-i] ^(i+1) eginez eta azkenik ciphertext-aren padding-a egokituko dugu urrengo iteraziorako.

```
for (i = 0; i < 16; i++){
        for (j = 0; j < 256; j++){
                if (i == 0 && (uint8_t) j == ignore) continue;
                ciphertext[31-i] = (uint8_t) j;
                plen = decipher_AES_CBC_PO(plaintext, ciphertext, clen);

                if (plen!=0){
                        intermediate[47-i] = ciphertext[31-i] ^ (i+1);
                        if (i < 15)
                        for (k = 0; k < i+1; k++){
                                ciphertext[31-k] = (i+2) ^ intermediate[47-k];
                        }
                        break;
                }
        }
}
```

Intermediate guztia lortu ondoren ordezkatu nahi dugun textuarekin xor egin eta ciphertext-en ordezkatu behar dugu.

```
for (i = 0; i < 16; i++){
        ciphertext[i+16] = desired_p[i] ^ intermediate[32+i];
}
```

Hemen exekutatuko programaren emaitza Oier eta Gaizka izenekin:

```
Ciphertext (hex): 60592ff65e192e29a29be678fc8873cd0aabea229e2d4521568b1fa32712a1fd8037b482bbc8f3bc523ad5e2e2fd0868
Plaintext (plen = 39): INTRODUCE NAME IN POSITION 32-47 PLEASE
Intermediate[47] f4
Intermediate[46] a8
Intermediate[45] 1b
Intermediate[44] 2e
Intermediate[43] aa
Intermediate[42] 16
Intermediate[41] 82
Intermediate[40] 5f
Intermediate[39] 28
Intermediate[38] 0
Intermediate[37] 7e
Intermediate[36] df
Intermediate[35] 67
Intermediate[34] a6
Intermediate[33] fb
Intermediate[32] 2a
Your name: OIER
```

```
Ciphertext (hex): 60592ff65e192e29a29be678fc8873cd0aabea229e2d4521568b1fa32712a1fd8037b482bbc8f3bc523ad5e2e2fd0868
Plaintext (plen = 39): INTRODUCE NAME IN POSITION 32-47 PLEASE
Intermediate[47] f4
Intermediate[46] a8
Intermediate[45] 1b
Intermediate[44] 2e
Intermediate[43] aa
Intermediate[42] 16
Intermediate[41] 82
Intermediate[40] 5f
Intermediate[39] 28
Intermediate[38] 0
Intermediate[37] 7e
Intermediate[36] df
Intermediate[35] 67
Intermediate[34] a6
Intermediate[33] fb
Intermediate[32] 2a
Your name: GAIZKA CEREZO
```

Aurkitu dugun arazo nagusia mezu berria sartzean beste bloke bat gehitu beharko genuke “Oier Barrutiabengoa” izena sartzeko. Paddinga oker ez erabiltzeko, aurresan ezin daiteken forma batean aplikatu beharko genuke.