Bit Flipping Attack on CBC mode (ಠ_ಠ)

Posted on **July 18, 2015**

Okay hôm nay mình sẽ bắt đầu một chủ đề khá thú vị trong lĩnh vực Cryptography như tiêu đề ~

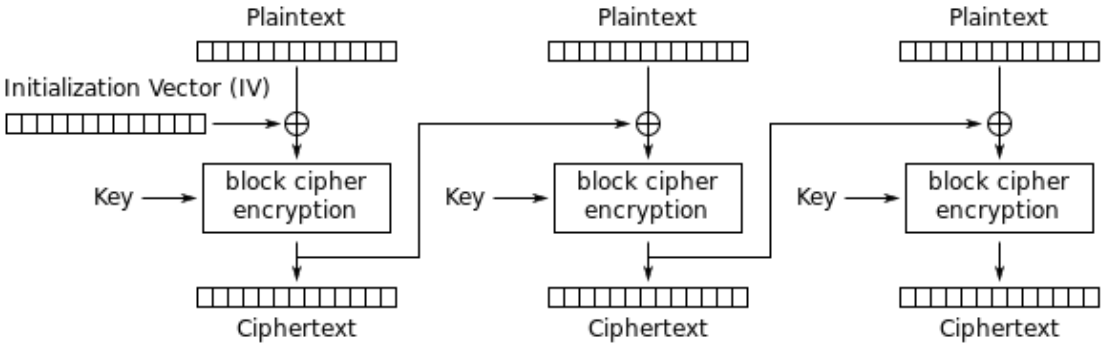
Sau một thời gian ngâm cái đống này thì mình phải nói rằng kỹ thuật này khá đơn giản nhưng ứng dụng thì thật không thể tin được.

Overview: Kỹ thuật này được dùng để thay đổi data gốc chỉ bằng việc thay đổi các bit ở ciphertext của server trả về.

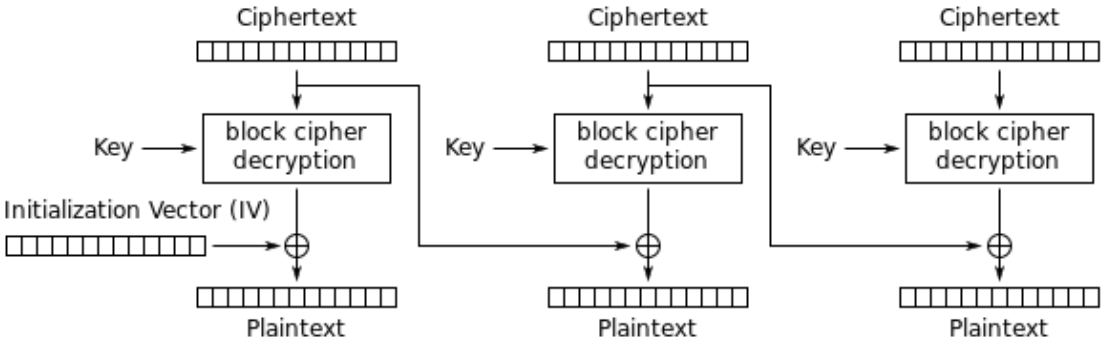
Ví dụ: chỉ bằng việc thay đổi ciphertext của “Tôi nợ bạn 10.000 vnđ”, nó đã trở thành “Tôi nợ bạn 10.000 usd”, rất tuyệt phải không? làm giàu không khó 😀

Okay, chúng ta “go deep” nào

1. CBC là gì nhỉ? Đó là viết tắt của Cipher Block Chaining – một trong những mode được sử dụng để đảm bảo cùng một dữ liệu input thì output sẽ là các cipher khác nhau



Cipher Block Chaining (CBC) mode encryption



Cipher Block Chaining (CBC) mode decryption

Có thể hình dung CBC như sau:

Step1: Dữ liệu sẽ được chia thành các khối nhỏ(block), block đầu tiên sẽ XOR với 1 IV (IV là một thứ ngẫu nhiên để đảm bảo cùng 1 input sẽ cho ra nhiều output tránh bị tấn công relay), sau đó kết quả (block-XOR-IV) sẽ làm input vào một thuật toán mã hóa khối nào đó như AES, và cho ra block ciphertext đầu tiên.

Step2: Ciphertext đó sẽ được tiếp tục sử dụng như một IV của các plaintext block sau (như hình), cứ như vậy cho đến hết

Step3: Nối các cipher block thu được và tạo ra một ciphertext hoàn chỉnh

2. XOR – đơn giản chỉ là một phép toán giống như AND, OR

xúc tích và không có gì để nói thêm **(¬‿¬)/**

3. Nói cho lắm rồi cũng tới phần ứng dụng (—‿‿—)

Notice 1: Đây không phải là kiểu tấn công giải mã trực tiếp, mà chỉ đơn thuần là thay đổi ciphertext để data ban đầu thay đổi
Notice 2: Phần giải mã của CBC

x[1] = D(y[1]) xor IV
x[n] = D(y[n]) xor y[n-1]

Nếu lật bit ở block ciphertext thứ (n-1) thì bit ở block ciphertext thứ (n) cũng bị lật theo

Một ngày đẹp trời bạn vào một trang web bán hàng online, thấy một cái áo ba lỗ rất đẹp nhưng không đủ tiền mua, mò quanh trang web thì bạn thấy khung search sản phẩm, bạn search thử vài cái thì thấy chủ trang web này kĩ tính quá, chơi hẳn CBC cho search function chứ không để ?search= như bình thường

Hình dung như sau:
Câu query có thể là:
Select * from Shop where Sanpham='$search';
Giả sử bạn search **abc**, kết quả trả về

http://victim.comm/search/aaabbbccc111
No result for abc

aaabbbccc111 chính là "abc" đã đi qua CBC
Search **xyz**, kết quả trả về

http://victim.comm/search/bbcacbbba231
No result for xyz

Kệ, ai mà quan tâm tới CBC trên url chứ, Sql injection vào tìm nick ông nào có tiền để mua đồ coi
search **' or 1=1– –**

http://victim.comm/search/cbbaaccba332
No result for \' or 1=1– –

Ây zà, đã bị filter rồi mới encrypt… ┌∩┐(◣_◢)┌∩┐
làm sao để tránh bị add thêm slash vào đầu đây…, đã tới lúc sử dụng Bit Flipping Attack!

search **a or 1=1– –**

> *http://victim.comm/search/aabbbbccc222*
> *No result for a or 1=1– –*

Như vậy ta chỉ cần modify ciphertext “aabbbbccc222” để biến a thành ‘ và câu query sẽ được thực thi.
Vì mỗi Plaintext block ban đầu sẽ cho ra các ciphertext block riêng biệt nên ta chỉ modify vừa đủ tùy theo thuật toán mã hóa khối ta sử dụng

modify thành “ffbbbbccc222”

> *http://victim.comm/search/ffbbbbccc222*
> *No result for ! or 1=1– –*

modify thành “fbbbbbccc222”

> *http://victim.comm/search/fbbbbbccc222*
> *No result for u or 1=1– –*

và cứ như vậy cho đến khi được dấu ‘ :D, server decrypt và sẽ hiểu rằng bạn đã nhập vào là ‘ or 1=1– – , thế là bypass filter thành công 😀

Một ví dụ khác qua video: