Thomas Byrnes & Harris Spahic
CS-396 Homework 1
10/25/2022

1)

1) Integrity and Confidentiality are the two basic security properties you would want to consider. You want to ensure that the asset is only viewed and/or modified by authorized users. This is relevant because you don't want any unauthorized parties to be able to modify off view Ka or Kb.

2) The protocol satisfies integrity because an unauthorized party would not be able to modify x and y being sent over an insecure channel. However, this protocol does not obey confidentiality since an unauthorized party would be able to view x and y, and thus be able to determine the key by xoring the exchanged messages. (if Ka = Kb, x^y = Ka = Kb)

2)

1) When the period is 1, the cipher is essentially a caesar cipher and you can tell if the letters in the ciphertext are successive, if so, then the user's password is p1. When the period is 2, since the first and third characters are an equal distance apart, and the second and fourth characters are also an equal distance apart in p1 and p2. In this case you cannot determine which password the user has used. Because for any encryption, any cipher that encrypts the first password then there is another cipher that encrypts the second password to the same value. When the period is 3, then you can compare the first and fourth characters in the ciphertext. If they are a distance 3 apart, then you can determine that the user has used password p1. When the period is 4, you are essentially implementing a One Time Pad which is perfectly secret and can't be decrypted, assuming the key is transmitted securely and never reused.

2) To recover the secret key you only need one character of the plaintext to determine how far the cipher is shifting the plaintext. The shortest single message plaintext you can use to recover the key would be 'a'. If we have no plaintext available to us, and the message being shifted is an incoherent term with no discernable patterns then the substitution cipher is perfectly secure.

3)

1)
Plaintext:
testing testing can you read this
yep I can read you perfectly well
awesome one time pad is working
yay we can make fun of Abrar now

i hope no student can read this
that would be quite embarrassing
luckily OTP is perfectly secret
didnt Abrar say there was a catch
maybe yet I didnt pay attention
we should really listen to Abrar
nah we are doing well without her

To obtain the key we used python to brute force decrypt the message. Once we had the key we used another python script to xor the ciphertext and the key to obtain the plaintext. Our code is available in the zip file.

2) To find the generated key after 21 days (October 4 - October 25), we again used a python script to take the key we found in part 1 and applied key = sha256(key) + '21'(00100001 in hexadecimal) twenty one times to find the key used on October 25:

f5e0a3dad5d9f4064ddfb7ec86ce195d74ed0e00ce2b99b6e8275dcb9269dcde21

4)

1) The NIST(National Institute of Standards and Technology) released Dual_EC_DRBG, a pseudorandom number generator in 2006. Shortly after, a pair of Microsoft researchers found vulnerabilities in the number generator that would potentially give NIST or the NSA (National Security Agency) a backdoor. If the NIST/NSA can rig the number generator, then they would be able to decrypt any type of encryption scheme that uses the Dual_EC_DRBG random number generator.

2) It is unethical for a government agency to have the ability to access all secret communications generated via a certain algorithm. This would allow the NSA/NIST to essentially spy on not just all Americans, but all people who encrypt messages using Dual_ECRBG. The security weaknesses of Dual_EC_DRBG were known before the algorithm was made a standard, and once it was made a standard, it remained a standard for eight years before it was withdrawn. This means that the United States government knowingly endorsed a flawed random number generator in order to increase its surveillance power over an unknowing population.

3) When the ethical codes of data privacy and security are not applied, you essentially allow agencies to collect personal data that was expected to be kept confidential. What these organizations do with the data is irrelevant. Even if it is harmless, which it likely isn't, they violate their users' trust by accessing information they shouldn't. By putting a backdoor in Dual_EC_DRBG, then paying RSA Security 10 million dollars to use that algorithm in their encryption scheme, the government essentially allowed anybody who knows how to use that backdoor access to your information. Typically this would be the NSA/NIST but there is nothing stopping bad actors within those organizations from using that data for nefarious purposes.

I pledge my honor that I have abided by the Stevens Honor System.