Aidan Fischer October 17, 2022

CS 396 Homework 1

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

1. 1. In this context, the two basic security properties that should be considered in the design of a secure protocol to confirm that the secret key held by both Alice and Bob is the same key as in the past, are confidentiality and integrity. They are communicating over an insecure channel, so they risk having unauthorized parties privy to whatever communication they make with each other to confirm the key they have. Confidentiality is important here to make sure those third parties do not have the ability to deduce the secret key(s) from the communications. Integrity is important because malicious modification or interception of communications between Alice and Bob can interfere with their confirmation method and result in false negatives or positives.
   2. No, it does not. Integrity is not achieved over an insecure channel. Alice and Bob make no attempt to confirm the identity of the sender or the authenticity of messages, and therefore attackers can maliciously modify a message or impersonate a sender. Confidentiality is also not achieved. Assuming Alice and Bob indeed share a key, an attacker intercepting communications between Alice and Bob can use this procedure to retrieve the shared secret key k
      1. The attacker has x = ka xor r, y = kb xor x = kb xor ka xor r
      2. Because the procedure succeeds when r = y, and we are assuming the procedure did succeed since we are assuming Alice and Bob share a secret key, then we assume y = r.
      3. Therefore, the attacker has r, and can retrieve k by taking x xor y = x xor r = ka xor r xor r = ka = k
2. 1. Since the attacker knows the possible passwords, we can exploit patterns in the known possible plaintexts to figure out which password is correct. For t = 1 to 4
      1. t = 1: This is very simple. If the ciphertext is subsequent letters (i.e. c = ijkl, c=efgh, etc), then the first password is the correct password. If they aren’t subsequent letters, then the second password is correct. A period of 1 is equivalent to a standard Caesar cipher.
      2. t = 2: Take the pairs of the 1st and 3rd, and 2nd and 4th letters of the ciphertext. Each letter of these pairs would respectively have been encrypted using the same shift letter. However, notice that the letter difference for each pair in each plaintext is the same (a, c; b, d; e, g). All pairs have letters that are two letters apart in the alphabet. This distance is reserved in the ciphertext, and thus cannot be exploited to differentiate plaintext. Since we don’t know the key, we cannot use other letter pairs to determine the plaintext, since the pairs’ letter distance in the ciphertext would be arbitrary. Therefore, the password is secure under t = 2.
      3. t = 3: First and final letter of ciphertext will have the same shift. In this case, the shift between the 1st and 4th letters of the plaintexts are different, and thus we can use the letter distance trick in this case. If the letter distance in this pair is 3 in the ciphertext, then the first password is correct. If it is 5, then the second password is correct.
      4. t = 4: All letters have arbitrary shifts without a known key. We cannot exploit letter patterns or letter pairs in this case, as there are no pairs that have known matching shifts. The password is trivially secure in this case.
   2. The mono-alphabetic substitution cipher is trivial to break under a chosen-plaintext attack since an attacker can just send a message with all 26 letters and get the corresponding replacements immediately. In fact, they don’t even need to use all 26 letters. Since the cipher only replaces a letter with a single other letter, the attacker only needs to send 25 of the 26 letters and can use the process of elimination to get the 26th letter’s replacement. A good short message would be a pangram (sentence that uses each letter of a language at least once) of the English language, but I noted that we only need 25 letters. The shortest example I could find with 25 unique letters, (found [here](https://puzzling.stackexchange.com/questions/102523/what-is-the-shortest-english-language-sentence-which-contains-25-unique-letters)) “Quick jigs vex Nymph of Bad Waltz,” consists of 27 letters (two duplicated), and excludes the letter r. Recovering the key of the 25 unique letters present, we can recover r by assuming the missing member of the key alphabet is the letter assigned to r. Against a ciphertext-only attacker, the mono-alphabetic substitution cipher is perfectly secure if messages do not reveal patterns that could be statistically analyzed. More specifically, ciphertexts that do not contain repeated letters and do not contain spaces, because then letter repeatings cannot be used to determine statistical likelihoods, and short words cannot be used to guess replacements (like the words I and a).
3. 1. An explanation of my strategy and algorithm is in Cryptanalysis.py in my submission. Here are the plaintexts and key  
      ‘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'  
      Key is ‘TheQuickBrownFoxJumpsOverLazyDog!’
   2. Ffe8144a63819ae4ba99c7549fa59776f5f8bbdd2841080f5ba87ce918c3578821. Algorithm in KeyRotation.py
4. 1. The controversy related to Dual\_EC\_DRBG involves organizations such as NIST, ISO, and the NSA. Another stakeholder is the RSA. NIST and ISO certified the algorithm as a valid cryptography algorithm. The NSA created the algorithm. The RSA made it the default in their cryptography software suite. The NSA’s goal, revealed by leaked documents, is to have backdoors into cryptography algorithms in order to have access to private, encrypted communications without the knowledge of the owners of the information. NIST and ISO are organizations that standardize these algorithms and recommend their public use. Their goal is to provide a set of algorithms that should be used, as opposed to unknown algorithms. However, they don’t necessarily verify the security of the algorithms they certify. RSA is a computer security firm that provides software for data security. They supposedly got bought off by the NSA to use Dual\_EC\_DRBG as their software’s default algorithm. However, their main goal is to provide cryptographic software solutions to the public.
   2. A broad ethical concern occurs here where multiple organizations recommended the use of a PRNG with a known potential backdoor, without adequately informing the general users of the algorithm about the possible vulnerability. Cryptographic researchers very quickly discovered the possibility of a potential backdoor, within weeks after the public release of the algorithm, and yet the algorithm went on to be used widely for 7 years before NIST finally retracted its certification and recommended everyone stop using it. Additionally, that only happened AFTER the NSA documents were leaked. If that didn’t happen, who knows how long that algorithm would go on being used.
   3. I will be referring to the ACM code of ethics at acm.org/code-of-ethics. The codes of ethics in this list broken here are 1.1, 1.2, 1.3, 1.6, 1.7, 2.1, 2.2, 2.5, 2.7, 2.9, 3.1, and 3.7. I won’t go into extensive detail about how each of these is broken, as that will take up multiple pages and isn’t the point of the question, but I will list the names of these codes
      1. 1.1 Contribute to society and to human well-being, acknowledging that all people are stakeholders in computing.
      2. 1.2 Avoid harm.
      3. 1.3 Be honest and trustworthy.
      4. 1.6 Respect privacy.
      5. 1.7 Honor confidentiality.
      6. 2.1 Strive to achieve high quality in both the processes and products of professional work.
      7. 2.2 Maintain high standards of professional competence, conduct, and ethical practice.
      8. 2.5 Give comprehensive and thorough evaluations of computer systems and their impacts, including analysis of possible risks.
      9. 2.7 Foster public awareness and understanding of computing, related technologies, and their consequences.
      10. 2.9 Design and implement systems that are robustly and usably secure.
      11. 3.1 Ensure that the public good is the central concern during all professional computing work.
      12. 3.7 Recognize and take special care of systems that become integrated into the infrastructure of society.

When such codes are not applied, we run into a situation where the public trusts and uses systems that are demonstrably untrustworthy. This is exactly what happened here. The public used an algorithm that the NSA supposedly could backdoor into, for seven years, despite the possibility of the backdoor having been known in the cryptography community since the algorithm’s induction. A large amount of data could’ve been stolen, all the while general users of the system were none the wiser.