ACCESS 2013 a.m.

Group Project - Cryptography Week

Due Friday, July 12, before midnight

RSA Public Key Lab: Set up, use, and explain an RSA public-key cryptography system for the 7 ACCESS groups. Follow the steps below carefully!!!

(1) Each team should choose two primes p,q between 10^30 and 10^31, so that the modulus N=pq is greater than 10^60 - and thus will have at least 61 digits. This means that when you send people messages you can use up to 60 digits in each number packet , **before** encoding with their public key. This forces each packet number to be safely in the residue range for the recipient's modulus before you encrypt it, so that when the recipient decrypts it, they will recover your original message. (After you encode a packet it will quite likely have 61 or 62 digits, but it will still be in the recipient's residue range because the encryption function transforms residue numbers to residue numbers. Real RSA systems use much larger primes, but with these lengths each residue number fits onto a single text line).

(2) After picking your primes find a suitable encryption power e. Use e and the auxillary modulus to compute your secret decryption power d. Make sure e and d are multiplicative inverses mod N2, check that you can successfully encrypt and decrypt messages using your own public and private information, and verify that your N really is bigger than 10^60, before proceding to step (3) - This double-checking should prevent errors which have occurred in some unfortunate ACCESS groups.

(3) As a further check that your data is correct, please use the provided link to submit your group's choice of the numbers p,q,N,N2,e. If these numbers are OK, send the result of the verification step to Fernando by email. The Cryptography ACCESS website
http://www.math.utah.edu/~fguevara/ACCESS2013
will act as "Certificate Authority" and will have a list of the public keys from all groups.

(4a) Create a favorite secret message such that it is no longer than 90 characters long (including spaces). Split your message into two or three packets of 30 characters or less. By using the conversion table in page 9 of Davis' notes (or an automated version that is posted in the class website) convert these two or three packets into two or three integers with at most 60 digits each. (Make sure that no packet starts with the digit "0", since this "0" at the start of a number wouldn't be visible to any unfortunate ACCESS group who correctly decrypted the encrypted message you sent them!)

(4b) Create any plaintext signature which identifies your group, using at most 30 characters. Convert your signature into a number with at most 60 digits, using Davis' table. We are using the secure signature feature, so decrypt your signature using your own (secret) decryption power. This will create a long sequence of digits, a number less than your groups' modulus but probably with 61 or 62 digits and potentially larger than the moduli of groups you're sending to. This means you'll need to break your decrypted signature into two packets of at most 60 digits each (and so that the second one doesn't have a lead "0")

(4c) You now have up to 5 packets of at most 60 digits: 3 from your plaintext secret message and 2 from your decrypted signature. None of these packets should have a leading 0. If you are group x then you will be sending messages to groups (x+1) and (x-1), mod 7. For example, group 3 sends messages to groups 2 and 4; group 7, also known as group 0, sends messages to groups 6 and 1. Encrypt the (up to) 5 packets you've created in parts (4a),(4b), using the public encryption keys for the two groups you're

sending to, and then email out your encrypted packets to the *accessam-2013@lists.utah.edu* mailing list **and** *fguevara@math.utah.edu*. Do not forget to specify to which group is your message intended to.

(5) Use your private key and reverse the encryption process to decode the messages you receive from your two neighbor groups. After you use your decryption key to decrypt the signature part of their messages, you'll need to glue the last two packets back together, and then encrypt with the sender's public key, in order to verify their plain text signature - because you need to undo everything they did.

(6) Create a lab report for this experiment: Describe the process you went through to set up the ACCESS RSA system. Exhibit your public and private key information, your original plain text message and signature, and the various transformations of your message and signature as you prepared them for transmission to your two target groups. Exhibit the encrypted messages you received, explain how you decrypted them, and exhibit the final results. Make sure all numerical representations of your messages are numbers and not pictures, so that we will have an easy time checking your work (which we will do!). Present this data in a careful, organized way. Explain well. In case there are either sending or receiving issues with the messages, work with the other group involved to trouble shoot errors in the various steps, without giving away your secret messages or signatures.

Paper submission: Please submit your group work to me (Fernando, fguevara@math.utah.edu) and Cheryl (zapata@math.utah.edu), as one single Maple Worksheet file. This project is **due Friday July 6**, before midnight. Help each other - you're all on the same ACCESS team! If you think a group sent you a defective message, contact them and explain what isn't working, as a prelude to both sides trying to troubleshoot the problem. If you can't unstick each other please contact Cheryl or Fernando. You can also come to my office (LCB 212) and I'll be happy to help. Have fun!!