CS 458 Information Security

Homework 3: Cryptographic Hash and MAC, Asymmetric Cryptography

1. Crypto Hashes and MACs

(a) What property must a good cryptographic hash function have?

Collision resistant. There are three levels of collision resistance (see slides); a good hash function satisfies all of them.

(b) CRC, or cyclic redundancy check, is a message digest that is used to detect transmission errors. Why shouldn't we use CRCs as MAC?

CRCs are sent in the clear, and are designed to detect random errors instead of malicious errors. For a MAC to be useful, it has to be keyed.

(c) Name one situation where integrity is of utmost importance but confidentiality is not of concern.

Distribution of open source software (e.g. Linux). We want to make sure end users only download what we want them to, but the content of the files don't need to be kept secret.

(d) **Bonus**: $r_1, r_2, \ldots, r_n$ are independent random variables uniformly distributed among $\{1, 2, \ldots, B\}$. In terms of B, what's the smallest n such that $\exists i{\ne}j, r_i = r_j$ with probability >1/2?

About $1.2 \times B^{1/2}$. Note that as long as $\{r_i\}$ are independent and identically distributed, n will be no larger than this number for a collision to be more likely than not.

(e) **Bonus**: given collision resistant hash function H, how is HMAC constructed? What are *ipad* and *opad*? What do they stand for?

$\mathrm{HMAC}(k, m) = \mathrm{H}(\, k \oplus opad \,||\, I\,)$ where $I = \mathrm{H}(k \oplus ipad \,||\, m)$

ipad = 0x36363636... is the inner pad. (0x36 = 00110110_2)

opad = 0x5c5c5c5c... is the outer pad. (0x5c = 01011100_2)

2. Public Key Crypto

(a) Unlike symmetric ciphers, an asymmetric cipher uses two different keys for encryption and decryption: a public key and a private key. Which key is used to encrypt and which to decrypt? (Hint: *hint hint.*)

Both can be used to encrypt or decrypt. Messages encrypted with one key can be decrypted with the other key.

(b) A good asymmetric cipher should make the following tasks easy (E) or hard (H)?

[Easy] Generate a key pair
[Easy] Encrypt a message with a public key
[Easy] Decrypt a message with corresponding private key
[Hard] Compute private key given public key
[Hard] A message is encrypted with a public key: decrypt this message given this public key

(c) Asymmetric cryptography is generally slower than symmetric cryptography

(d) What can we do with asymmetric cryptography that we can't with symmetric cryptography?

- Establish session keys
- Authenticate origin of message
- Encrypt message non-interactively (DH)
- etc.

3. Diffie-Hellman and RSA

(a) Which protocol would you use to send an encrypted email to your friend, DH or RSA? Justify your answer.

Diffie-Hellman.
No interaction is required between the two parties to establish a secret key, which is perfect for emails because they bounce between numerous relay servers before reaching their destination.

(b) What hard problem does Diffie-Hellman protocol depend on? What hard problem does RSA depend on?

DH: Discrete Logarithm
RSA: Factorization of big numbers

(c) **Bonus**: $\varphi(n)$ is the number of integers in $\{1, 2, ..., n\}$ relatively prime to n. Prove if p, q are prime, then $\varphi(pq) = (p-1)(q-1)$.

In $\{1, 2, \ldots, pq\}$, q numbers are multiples of p, and p numbers are multiples of q, but we counted pq twice. Therefore $\varphi(pq) = pq - q - p + 1 = (p-1)(q-1)$.