Sreeram Danda
CS577 Homework #3
Problem #4

#4a
Since the adversary knows a *v* they can determine a solution to the expression (ax + b) but cannot determine the exact values of *a* and *b* themselves. Both of these variables can be any number of combinations of primes. Furthermore, since *x* is a large prime and *a*, *b* are also prime integers it will take an unreasonable amount of time to determine the exact *a* and *b* used to get the result by brute force regardless of the fact they know the corresponding (*x, v*) pair. Therefore, the adversary must make a completely random guess with only the knowledge that v' must be between 0 and *p*, due to the definition of modulus. Since, the range of the guess is (0, *p*). The probability that the adversary correctly guesses the verification value for x' is 1/*p*.

#4b
If the adversary knows two pairs of (message, verification), then solving for *a* and *b* becomes trivial. Using simple substitution, the adversary can generate two equations where the only unknown is *a* and *b* by using the two messages as *x* and *y,* using their corresponding *v* and *w* as h(x) and the known public prime *p*. Thus, very easily solving for *a* and *b*. Once the adversary knows the two primes *a* and *b*. They can generate any message *x*' and find its corresponding verification value by using the public has function. Thus, they have successfully generated a valid (message, verification) pair that can be used maliciously.

#4c
A scheme that would allow the adversary to know two (message, verification) pairs and still be unable to imitate either person would be to use a hash function in the form:

$$h(x) = ((ax + b) mod\, p)\; mod\, m$$

In this function, the newly added m represents the number of slots in the hash table, unknown to the adversary and is such that *p* > *m*. The new family of hash functions representing this form is:

$$H_{ab} = \{h:\ a, b \in Z_p,\ a \neq 0\}$$

Assume that the adversary has the two message pairs (x, v) and (y, w) from part 4b. If the adversary wants to send a new message *z* with either *v* or *w* as the verification tag then the given hash functions are:

$$v' = (az + b) mod\, p \qquad\qquad w' = (az + b) mod\, p$$
$$v = (ax + b)\, mod\, p \qquad\qquad w = (ay + b) mod\, p$$

Note that: $v' - v \equiv a(z - x)\, mod\, p$ and $w' - w \equiv a(z - y) mod\, p$

Therefore, v' cannot equal v and similarly w' cannot equal w, because *p* is a large prime and z, x, y are non-zero messages that are not equal. Thus, the only way the equation is true is if *a* = 0 which is a contradiction with the definition of the hash function family set up earlier. Furthermore, attempting to solve for the values of *a* and *b* will not be deterministic; there are *p(p-1)* choices for each pair of *a*,*b* values. So for any input message, randomly picking *a*,*b*, there is an equal probability for any of those pairs of *a*,*b* to be the resultant hash value h(x). Therefore the probability that a message, verification pair made by the adversary is legitimate is still 1/p.