292 – Homework Assignment 3

Fall 2021

Due: Monday, December 6th at noon

Do not discuss the problems with anyone other than the instructor.

1. Consider the following code for checking if a 3 character password is entered correctly (`key` is the password, `input` is the user input):

```
check1(key,input) {
  for (int i = 0; i < 3; i++)
    if (key[i] != input[i])
      return 0;
  grant_access();
  return 1;
}
check2(key,input) {
  if (key == input) {
    grant_access();
    return 1;
  }
  else
    return 0;
}
```

Assume that each password character and each input character is a lowercase alphabet character. Also assume that `key == input` checks if string `key` is equal to string `input`. **(a)** Assuming that the symbolic execution framework tracks the number of instructions executed using a variable O, generate the observation constraints for the above procedures. **(b)** Use model counting the determine the probabilities for each path constraint and compute the information leakage using entropy assuming that the value of O is observable.

2. Consider the following procedure where S is a (uniformly distributed) secret value and I is a public input.

```
P(S, I) {
  if (S < I)
    sleep(1);
  else
    sleep(2);
}
```

Assume that S and I are both 4 bit values (ranging from 0 to 15) where S is the secret and I is the public input. Generate the path constraints for this procedure corresponding to different observables (assume that the observable is the execution time and different sleep times are observable).

Represent the amount of information leakage in terms of entropy for a single execution of P as a function of input I (first write a function of I that represents the number of models for a given I). Which value of I maximizes the information leakage?

3. Consider the following procedure:

```
P(H) {
  if (H > 0)
    result = 1;
  else
    result = 0;
  return result;
}
```

Use the quantitative information flow analysis based on bounded model checking to check if the channel capacity for this procedure is 1. First, show the template code calling this procedure (which should include an assertion that limits the channel capacity to 1). Show the formula that bounded model checking technique will generate based on the template code. Is the generated formula satisfiable and what can you conclude from the satisfiability of the generated formula?

4. Consider the following procedures where H and L are high and low security variables, respectively.

```
P1(H) {
  if (H)
    L = 0;
  else
    L = 1;
}

P2(H) {
  if (H)
    X = 0;
  else
    X = 1;
  L = X & H;
}
```

Use self-composition to transform these procedures so that the non-interference property can be expressed as a safety property. State the non-interference property for the transformed procedures. Do these procedures satisfy the non-interference property?

5. (a) Write an HyperLTL property that represents non-interference. Show a set of traces that satisfy the property and a set of traces that violate the property. **(b)** Write an HyperLTL property that limits the channel capacity of the leakage to 2 bits. Show a set of traces that satisfy the property and a set of traces that violate the property.