Assignment 1

# Part 1: Short answer questions

1. **Determine the entropy associated with the following method of generating a password.**

Choose and place in this order one lowercase letter followed by one upper case letter, followed by two digits, followed by @, followed by two letters, each upper or lower case, and then followed by four symbols drawn from the set {$,7,3,v,w,J,z,T}. Finally, apply the hash function Tiger to give an output string in hex, which will be used as a password.

Step 1: 1 lowercase letter.

* There are 26 lowercase letters, Entropy

Step 2: One uppercase letter

* There are 26 uppercase letters. Entropy is

Step 3: Two digits

* There are ten digits (0-9)
* Entropy for each digit is
* For two digits, the entropy is

Step 4: Symbol ‘@’

* There is one symbol ‘@’
* Entropy: (since there is only one choice, no randomness)

Step 5: Two letters, each upper or lower case

* There are 52 possible letters (26 lowercase + 26 uppercase)
* Entropy for each letter is
* For two letters, the entropy is

Step 6: Four symbols are drawn from the set {$, 7, 3, v, w, J, z, T}

* There are eight possible symbols
* Entropy for each symbol is
* For four symbols, the entropy is

Step 7: Applying the Tiger hash function

* Hashing does not add entropy to the password; it simply transforms it. Therefore, I don’t need to add entropy for this step.

**Total Entropy Calculation**

First, compute the password space.

* There are 26 lowercase characters a-z: the first character has choices
* There are 26 uppercase characters A-Z: The next character has choices
* There are ten numbers from 0-9: the following two characters have choices
* There is one symbol, ‘@’ following character will have choices
* There are two letters, each upper or lower case from a-zA-Z: The next two characters have choices
* There are four symbols drawn from the set {$, 7, 3, v, w, J, z, T}: the following four characters will have choices

The total number of possible passwords is:

Hence the entropy is

1. **For the following collection of statements, describe the sets of actions, objects, and subjects; and draw an access control matrix to represent the scenario.**

* Alice can climb trees and eat apples.
* Bob can climb fences, eat apples, and wave flags.
* Trees can hurt apples.
* Carol can jump waves, eat apples, and wave flags.

**Subjects**

* Alice
* Bob
* Carol
* Trees

**Actions**

* Climb
* Eat
* Wave
* Hurt
* Jump

**Objects**

* Apples
* Flags
* Waves
* Fences
* Trees

**Access Control Matrix**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Subjects** | **Trees** | **Apples** | **Fences** | **Flags** | **Waves** |
| **Alice** | Climb | Eat |  |  |  |
| **Bob** |  | Eat | Climb | Wave |  |
| **Carol** |  | Eat |  | Wave | Jump |
| **Trees** |  | Hurt |  |  |  |

1. **Assume an application requires access control policies based on the applicant’s age and the type of funding to be provided. Using an ABAC (attribute-based access control) approach, write policy rules for each of the following scenarios:**
   1. If the applicant is older than 35, only “Research Grants (RG)” can be provided.

* **Attributes:**
  + Subject (Applicant): `age`
  + Object (Resource): `” Research Grants (RG)”`
  + Action: `provide`
* **Policy Model:**
  + Condition: `IF applicant.age > 35`
  + Rule: `THEN permit access to “Research Grants (RG)” `
  + Relationship: Age attribute greater than 35 allows the provision of “Research Grants (RG)” only.
* **Architecture**
  + Policy Management: Evaluates the applicant’s `age` attribute during the request.
  + Enforcement Point: Ensures that only “Research Gate (RG)” is provided if the applicant’s `age` exceeds 35.
  + Attribute Source Interaction: To enforce this policy, the system queries the applicant’s `age` attribute.
  1. If the applicant’s age is less than or equal to 35, both “RG and Travel Grants (TG)” can be provided.
* **Attributes:**
  + Subject (Applicant): `age`
  + Object (Resource):
    - `Research Grants (RG)`
    - `Travel Grants (TG)`
  + Action: `provide`
* **Policy Model:**
  + Condition: `IF applicant.age <= 35`
  + Rule: `THEN permit access to “Research Grants (RG)” AND “Travel Grants (TG)”`
  + Relationship: Age attribute less than or equal to 35 allows the provision of both “Research Grants (RG) and “Travel Grants (TG)”.
* **Architecture:**
  + Policy Management: Evaluates the applicant’s `age` attribute during the request.
  + Enforcement Point: Ensures that both “Research gate (RG)” and “Travel Grants (TG)” are provided if the applicant’s `age` is less than or equal to 35.
  + Attribute Source Interaction: The system queries the applicant’s `age` attribute to enforce this policy.