CSCI262 – System Security

Assignment 1

Part One Question 1

aA11@aA$$$

1 lower case letter = 26
1 upper case letter = 26
2 digits = 10 x 10
@ = 1
2 letters (upper or lower) = 52 x 52
3 symbols drawn from {$, 9, 5, v, w, l} = 6 x 6 x 6

Total strength = 26 x 26 x 10 x 10 x 1 x 52 x 52 x 6 x 6 x 6 = 39 482 726 400

Applying Tiger hash function (192-bits) = cf84d0a641166ec281e3faf58f1701bcaab3e4fc4a766fc8

What is the entropy of the password?

Since the hash output is only made up of lower-case alphabets and digits with a length of 48
Entropy = 48 x log base 2 (36) = 248.156 (to 3d.p)

Did entropy of password increase after applying the hash function?

Strength of password (aA11@aA$$$) = 39 482 726 400
Entropy of password (aA11@aA$$$) = 1 log base 2 (39 482 726 400) = 35.201 (to 3d.p)
Yes, the entropy of password increased after applying the hash function from 35.201 to 248.156

Did the strength of password increase after applying the hash function?

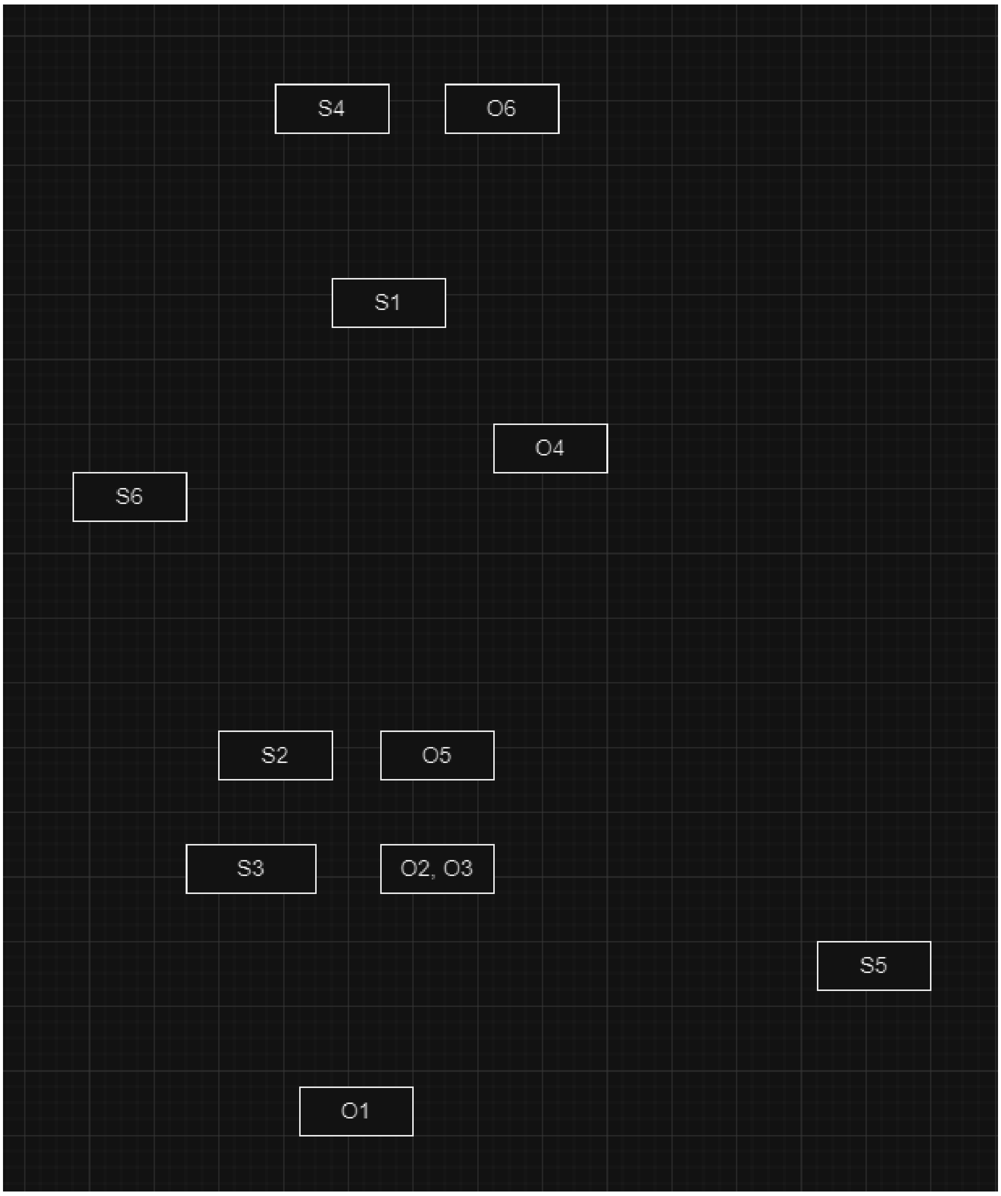
Strength of password (aA11@aA$$$) = 39 482 726 400
Strength of password after hashing = 2^248.156
Yes, the strength of the password also increases after applying the hash function.

Part One Question 1

From the start, base on the access control matrix, I started to put each object into 1 individual box and start to level them based on their dominance.

Example:
Since S3 can both read and write O2 and O3, I placed them on the save level.
Since S2 can both read and write O5, I placed them on the same level.
Since S4 can both read and write O6, I placed them on the same level.

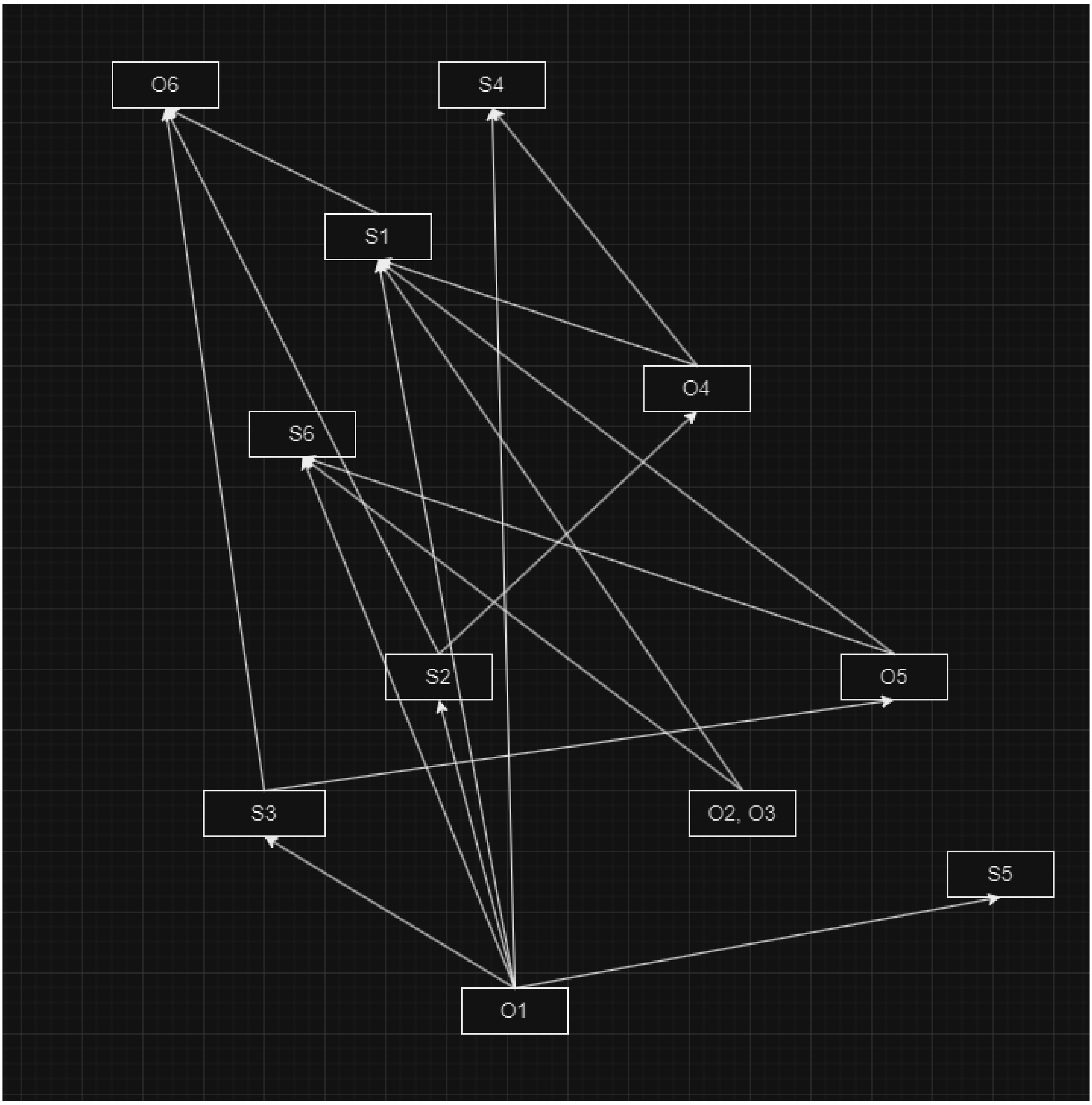
Since S1 can read O1 to O5 and can read O6, I placed S1 above O1 to O5 but below O6 and so on. Below is the result.



Then I started to draw the relationship between the Subject and Object.

Example:
Since O1 can be read by all S1 to S6, I draw arrow from O1 to all S1 to S6.
Since O2 and O3 can be read by S1 and S6, I draw arrow from O2 and O3 to S1 and S6.
Since S3 can write to O5, I draw arrow from S3 to O5 and so on.
I did not draw relation between S and O at the same level since they can both read and write and also could be merge later on.

Below is the result.

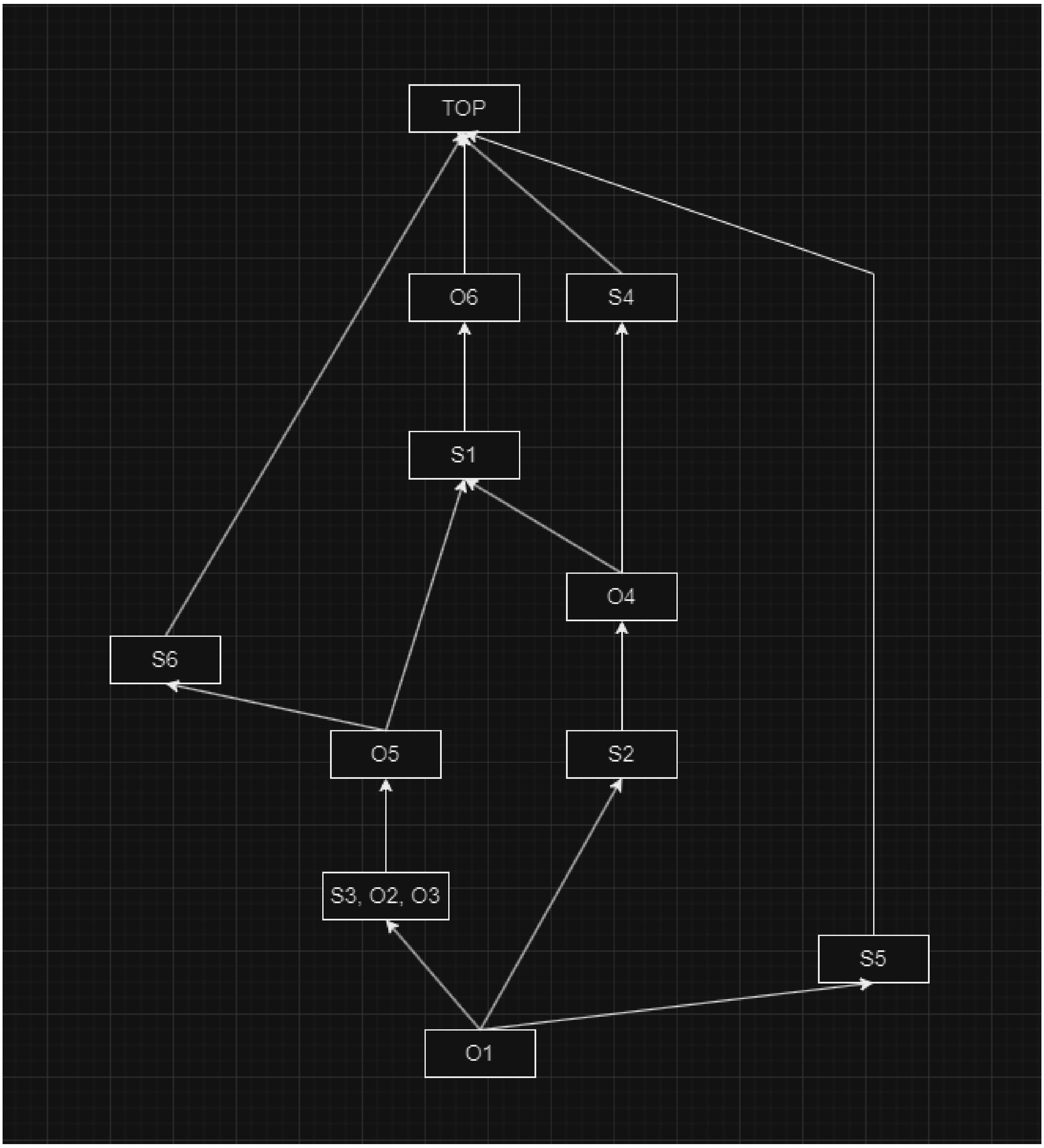


Lastly, I will remove the transitive relationship.

Example:
Since O2 and O3 can be mapped to S1 and S6, and S3 is already mapped to S1 and S6 via O5, I can merge O2 and O3 together with S3 on the same level and remove redundant relationships.
Since O1 can be mapped to S4, and S2 is already mapped to S4 via O4, I can remove the transitive relationship that goes directly from O1 to S4 and so on.

Lastly after removing all redundant and transitive relationships arrows, I organised them by shifting them around to make it look neater and ensure the arrows does not cross each other.

Below is my final BLP lattice-structured system.



Part One Question 3

-Alice can climb trees and eat apples

Actions: climb, eat
Subject: Alice
Object: trees, apples

-Bob can climb fences, eat apples, and wave flags

Actions: climb, eat, wave
Subject: Bob
Object: fences, apples, flags

-Trees can hurt apples

Actions: hurt
Subject: Trees
Object: apples

-Carol can jump waves and wave flags

Actions: jump, wave
Subject: Carol
Object: waves, flags

Access Control Matrix

	Trees	Apples	Fences	Flags	Waves
Alice	climb	eat			
Bob		eat	climb	wave	
Tress		hurt			
Carol				wave	jump