CS6600 Homework 3

Matthew Whitesides

I. Chapter 3 Problems

1) $A[s_1, o_1]\ and\ A[s_1, o_2] = A[s_1, o_2] \cup A[s_2, o_2]$ essentially says that the first two subjects created are able to give the entire scope of rights available to the system as not other subjects with lesser rights would be able to be created after that. This works because whatever initial rights *s1* had over *o2* would give the rights of any object not explicitly created for that object, plus the rights of a subject (*s2*) has over that object. If one could test for the absence of rights, this statement would still be valid; however, you wouldn't need to use both objects. You could do something like $A[s_1, o_1] \cup (\forall r \notin A[s_1, o_1])$ check for all rights and rights not in that to get the entire scope, but that would be more operations.
2) We can omit the delete and destroy commands because they inherently cannot "add" rights; they can only take them away. Therefore no leak could occur from those commands, only the opposite of a leak which may not be optimal for a user but is secure. This situation would be different if we could test for the absence of a right. However, we could use the test for absence to determine if a right has been deleted from a subject making the minimal set of operations reduced given we currently have to check each right.
3)
   a) Modifying the definition to say leaks occur beyond the initial state of the cell, then the delete and destroy commands would affect eh ability to leak a right. If so, we cannot get rid of them because we now have a definition where you can delete rights potentially all of the rights of a subject, then any addition of any basic right would cause a leak.
   b) If there is no create command no additional objects or subjects are added, therefore the number of *k* operations is unchanged and $(k \leq n|S_0||O_0| + 0)$.
   c) If we have created subject *s* where *s* is in the initial set, the total number of subjects and objects will not change. However, two new commands will be executed to delete and create the subject and modify an object. Therefore $k \leq n|S_0||O_0| + 2$.
   d) If the subject is not in the initial set then we have in the worst case one new subject $|S_0 + 1|$ and $|O_0 + 1|$ objects, and we continue to execute the two operations like in part b, giving us $(k \leq n|S_0 + 1||O_0 + 1| + 2)$.
   e) If we are creating a new object that was in the initial set, however, we have a total of 4 new commands that must be executed the delete and create the object and the subjects with rights on the object. Then since the rights must be reentered into each object we must have $|S_0+1|$ and $|O_0+1|$. Therefore we end up with $(k \leq n|S_0+1||O_0+1|+4)$.
   f) Finally, if the create object is not in the initial set, we have everything from part d that needs to be done plus the addition of the new object the number of rights must execute, leaving us with $(k \leq n|S_0 + 1||O_0 + 2| + 4)$.
4) The change in determining if a right has leaked to any addition of a right compared to the previous state would not change the theorem 3.2 much. The state machine setup and the problem would remain the same, and we'd still be looking for state q_f, but instead of comparing it to state q_0, we'd be looking for q_{f-1} however it's still undeterminable if we will enter state q_f.
5) Just because the HRU model proves that any given right is undeterminable if it leaks does not mean it's not worth studying the security of a system. The HRU proof is very general, and we can break down the system into mono-operational commands that create a decidable NP-complete problem. Also, on a practical scale like a UNIX system, even if you can't prove fully that a system is secure, you can simulate most standard use cases, which provides an ever closer to complete level of security. Another aspect is that just because it's unprovable, a right can "leak" isn't always bad. Systems such as UNIX are based upon the idea that subjects and grant rights to other subjects. At the same time, it's still true if you narrow the problem down to unauthorized subject right allocation. It's not as general as the HRU model would show.

Acknowledgment

The author would like to thank Professor Bruce McMillin with the Department of Computer Science, Missouri University of Science and Technology.

Matthew Whitesides Master's Student at Missouri University of Science and Technology.