Introduction to IT Security

Homework 5 – Authentication and Authorisation

Abraham Murciano

September 22, 2020

1 Access Control Lists and Capabilities

There are two prominent methods which are used to control access to objects (resources such as files and processes). One is called access control list (ACL) and the other is called capability list (C-list).

An ACL is a data structure that indicates which subjects have which permissions for each object. Conversely, a C-list is a data structure which indicates which permissions on which objects each subject has.

One advantage of using access control lists is that permissions for any given object are all stored together, so it is easy to view who has which permissions for an object. Another advantage is that if there are more resources than users (this is often the case) then it is a lot quicker to check if a specific user has access to a specific resource, than if we were using C-lists. This is because access control lists at least as shallow as the number of users, whereas the shallowness of C-lists are on the order of the number of resources. Additionally, with ACLs it is very easy to remove (or add) objects, since each list is only related to one single object, so it would only require deleting a single list.

However, C-lists also have their advantages. Firstly, it is very easy to remove (or add) users from a system which implements C-lists, since all of that user's permissions are stored together in one list, and there would be no need to traverse multiple lists in search of references to a specific user. Secondly, C-lists are not vulnerable to confused deputy attacks in the way which ACLs are. A confused deputy attack is when a program is acting on a resource on behalf of a user, and the user provides (as raw data) an identifier to a resource which they themselves do not have permission to use, but the program which is acting as an agent does. This is a problem in ACLs because when the access control check is performed, the list belonging to the resource whose identifier was passed is checked, and since the agent program is in that list, the access control check passes. However when using C-lists, in order for the access control check to be performed, the list of everything the user has permission to do must be passed

along to the check. Therefore the correct list (that of the user, not the agent program) would be passed, and the access control check would correctly fail.

2 SELinux in Android

Security-Enhanced Linux in Android uses a mandatory access control (MAC) access policy to manage access to all processes running on the operating system. When a system enforces MAC, it means that the security policy is controlled by a security policy administrator, and they are the only ones who can allow or deny permissions to users. This is in contrast to discretionary access control (DAC) where any user or process can grant any permissions which they have to other users or processes.

3 John the Ripper

We are tasked with using the program John the Ripper to crack the password encrypted in the following extract from an unshadowed password file.

```
userChallenge:$6$Lz6Uh2V9$L5Uqlh7ML66JtLAyX3ZKaEIhCF2QLbvJ02KZdMn
rXj.hmKpBLMBhy3g.B24R9r9iq.5omXUia.FISArUJMJEn/:1001:1001::/home/
```

Firstly, we will create a file with these contents and call it `password`. Then we modify the conf file located in `/usr/share/john/john.conf` by appending our rules, since we know that the third letter is uppercase and the password ends in 777.

```
# ...
[List.Rules:hw5]
T2Az"777"
```

Here, `T2` tells John to convert the character at index 2 (starting from index 0) to uppercase. Then `Az"777"` tells John to insert the string "777" to the end of each password.

Then we run the following command command, and obtain the displayed output.

```
$ john --wordlist --rules=hw5 ./password
Using default input encoding: UTF-8
Loaded 1 password hash (sha512crypt, crypt(3) $6$ [SHA512 128/128
AVX 2x])
Cost 1 (iteration count) is 5000 for all loaded hashes
```

```
Will run 8 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
drAgon777        (userChallenge)
1g 0:00:00:00 DONE (2020-09-22 00:02) 4.166g/s 2133p/s 2133c/s
2133C/s 123456777..crYstal777
Use the "--show" option to display all of the cracked passwords
reliably
Session completed
```

Finally, we can run this command to display the password, which tells us that the password was in fact `drAgon777`.

```
$ john ./password --show
userChallenge:drAgon777:1001:1001::/home/userChallenge:/bin/sh

1 password hash cracked, 0 left
```