3 Operating Systems (rmm1002)

A UNIX system has three accounts: the standard *root* account plus three user accounts, *alice*, *bob*, and *chris*. The *access matrix* describes the various operations that different *domains* (also termed *subjects* or *principals*) can invoke on individual *objects*.

(*a*) In a large system, the access matrix may be very large but also very sparse. State the two common representations of the access matrix, and explain which you would use to represent access rights for a modern personal laptop.

[4 marks]

(*b*) The above UNIX system has four peripherals attached: a printer, a removable hard disk used for backups, a web camera with integral microphone, and a set of speakers. Give the access matrix corresponding to the following set of policy statements for the *read* and *write* operations on the four peripherals:

- *alice* may use the printer fully but *chris* may not use the printer at all, and *bob* may only check the printer's status.

- Only *root* may create backups and only *bob* may recover files from a backup.

- *alice* and *bob* are both allowed to participate fully in video-conferences, while *chris* may only play music.

[5 marks]

(*c*) Describe how an administrator might configure users, groups and file permissions on the system to create files with the following permissions:

(*i*) A file readable and writeable by *root* and *alice* and readable by *bob*.

(*ii*) A file readable and writeable by *root* and *bob* and readable by any user.

(*iii*) A file owned by *alice* but readable and writeable only by *root*.

[7 marks]

(*d*) A tidy-minded system administrator decides they would prefer all user IDs to count up from 1000. They start by changing the user ID of the *root* user from 0 to 1000. How does this change the access matrix for the files described in (*c*)? How will this impact administration of the system subsequently?   [4 marks]