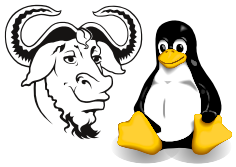

Tutorial on Shell, and the Secure Shell

1 Shell Programming

1. What will this shell script do? Disuss the purpose of this script.

```
#! /bin/sh
if ! grep nickl /etc/passwd > /dev/null 2>&1
then
    useradd -c 'Nick (local)' nickl
fi
if ! grep nickl /etc/sudoers > /dev/null 2>&1
then
    echo 'nickl  ALL=(ALL) ALL' >> /etc/sudoers
fi
```

2. What will the output of this shell script be?

```
#! /bin/sh

i=2
for j in 2 4 6
do
    j=`expr $i \* $j`
    i=`expr $i + 1`
done
echo "i=$i j=$j"
```

```
$ ./for-loop-question
i=5 j=24
```

3. Rewrite the script from question 2 above using a `while` loop.

```
#! /bin/sh

i=2
j=0
k=0
while [ $k -lt 6 ]
do
    k=‘expr $k + 2‘
    j=‘expr $i  $k‘
    i=‘expr $i + 1‘
done
echo "i=$i j=$j"
```

4. Write a shell script to print all its parameters, each one on a line by itself.

```
#! /bin/sh

for parameter
do
   echo $parameter
done
```

5. Write a shell script to read a text file in a format like this:

```
nicku:Nick Urbanik
fred:Freddy Wong
albert:Albert Ho
```

and create user accounts for each user.

Here is a strong hint towards getting a solution:

```
$ cat reading-line-by-line-and-splitting
#! /bin/sh

while read line
do
    echo     "This is a line:           $line"
    IFS=:
    for part in $line
    do
        echo This is part of the line: $part
    done
done

$ reading-line-by-line-and-splitting < ~/account-info.txt
```

```
This is a line:           nicku:Nick Urbanik
This is part of the line: nicku
This is part of the line: Nick Urbanik
This is a line:           fred:Freddy Wong
This is part of the line: fred
This is part of the line: Freddy Wong
This is a line:           albert:Albert Ho
This is part of the line: albert
This is part of the line: Albert Ho
```

```
#! /bin/sh

IFS=:
while read line
do
    i=1
    for part in $line
    do
        [ "$i" -eq 1 ] && uid=$part
        [ "$i" -eq 2 ] && name=$part
        i=‘expr $i + 1‘
    done
    echo useradd -c ¨"$name"¨ $uid
done
```

Here is another solution that uses `set`:

```
#! /bin/sh

IFS=:
while read line
do
    set $line
    uid=$1
    name="$2"
    echo useradd -c ¨"$name"¨ $uid
done
```

Here is another that uses the `awk` programming language:

```
#! /bin/sh

while read line
do
    awk -F: ’print "useradd -c " "’"$2"’ " $1’
done
```

2 Secure Shell

1. The user keys are stored in ∼/.ssh/id_rsa, ∼/.ssh/id_rsa.pub and ∼/.ssh/authorized_keys2.

(a) Are these user keys required for SSH to enable remote log in? If not all, and if some are required, list the ones that are required.

✎

(b) What is the purpose of each one of these key files?

✎

(c) How do you create them, and if we have one client and one server computer, which keys are required where?

✎

2. The host keys are stored in /etc/ssh/ssh_host_rsa_key, /etc/ssh/ssh_host_rsa_key.pub, ∼/.ssh/known_hosts2 and /etc/ssh/ssh_known_hosts2.

(a) Are all these host keys required for SSH to enable remote log in? If not all, and if some are required, list the ones that are required.

✎

(b) List two purposes of all of these key files.

✎

(c) How do you create them, and if we have one client and one server computer, which keys are required where?

✎