- The `chmod` command changes permissions on files (the permission descriptions can be provided by symbolic descriptions, or by octal numbers). Only the owner of the file (and the su) may change permissions on a file.

- If a directory is writable, people can remove files in it regardless of the permissions on the files themselves. Remove write permission from directory to make sure no one can delete files from it.

Inodes

- A file has several components: a name, contents, adm info, etc. The adm info is store in the inode (originally, "i-node"), along with essential system data such as how long it is, where on the disc the contents of the file are stored, and so on.

- There are three times in the inode:
  - time when the contents of the file were last modified
  - time when the file was last used
  - time the inode itself was last changes (e.g., to reset permissions)

- The system's internal name for a file is its i-number: the number of the inode holding the file's information. `ls -i` reports the i-numbers in decimal.

- The `ln` command makes a link to an existing file, with the syntax: `ln old-file new-file`. The purpose of a link is to give two names to the same file, often so that it can appear in two different directories. Two links to a file point to the same inode, and hence have the same i-number:

  ```
  $ ls -li junk
  1531917 -rw-rw-r-- 1 you you 8 Mar 12 08:30 junk
  $ ln junk trash
  $ ls -li
  1531917 -rw-rw-r-- 2 you you 8 Mar 12 08:30 junk
  1531917 -rw-rw-r-- 2 you you 8 Mar 12 08:30 trash
  ```

  Each link is equally important, and there is no difference between the first link, and subsequent ones (the "first link" is junk!).

- On the other hand:

  ```
  $ ls -li
  1531917 -rw-rw-r-- 1 you you 8 Mar 12 08:30 junk
  $ cp junk trash
  $ ls -li
  1531917 -rw-rw-r-- 1 you you 8 Mar 12 08:30 junk
  1531916 -rw-rw-r-- 1 you you 8 Mar 12 08:36 trash
  ```

Devices

- One of the nicest ideas in the UNIX system is the way it deals with peripherals—discs, tape drives, printers, terminals, etc. Rather than having special routines to, for example, read magnetic tape, there is a file called `/dev/mt0`. Inside the kernel, references to that file are converted into hardware commands to access the tape, so if a program reads `/dev/mt0`, the contents of a tape mounted on the drive are returned.

- For example,

  ```
  $ cp /dev/mt0 junk
  ```

  copies the contents of the tape to a file called `junk`. `cp` has no idea there is anything special about `/dev/mt0`; it is just a file—a sequence of bytes.

```
69445 crw-rw---- 1 root uucp  4, 66 Aug 30  2001 ttyS2
70258 lrwxrwxrwx 1 root root      8 Nov 15 06:36 cdrom -> /dev/hdc
65683 brw------- 1 root disk 22,  0 Aug 30  2001 hdc
65649 brw-rw---- 1 root disk  3,  0 Aug 30  2001 hda
66109 crw-r----- 1 root kmem  1,  2 Aug 30  2001 kmem
65242 crw-r----- 1 root kmem  1,  1 Aug 30  2001 mem
69064 crw--w---- 1 root root  4,  0 Mar 12 07:17 tty0
69065 crw------- 1 root root  4,  1 Mar 12 07:18 tty1
```

For a device file, the inode contains the internal name for the device, which consists of:

- Its type: character (c) or block (b). For example, disks and tapes are block devices, and terminals and modems are character devices.

- A pair of numbers, called the major and minor device numbers. The major number encodes the type of device, while the minor number distinguishes different instances of the device.

- All files accessible in a UNIX system are arranged in one big tree, the file hierarchy, rooted at /. These files can be spread out over several devices. The `mount` command serves to attach the file system found on some device to the big file tree. Conversely, the `umount` command will detach it again.

- For example:

  ```
  mount -t iso9660 /dev/cdrom /mnt/cdrom
  ```

Review of Cryptography

In the RSA scheme, Alice constructs functions E and D as follows:

- She chooses two distinct (odd) primes p, q, and computes $n = p \cdot q$.
- Alice keeps p, q secret, while n is given to the public.
- Alice also chooses an element $k \in \mathbb{Z}^*_{\phi(n)}$ with $k > 1$, and advertises k along with n. (Note that $\phi(n) = \phi(pq) = (p-1)(q-1)$, so it is easy to compute.)
- The encoding function is $E(M) = M^k \pmod{n}$, assuming that messages correspond to the elements of $\mathbb{Z}_n$ (if messages are bigger, then divide them into portions, each portion in $\mathbb{Z}_n$, and encode each portion separately).
- Alice can easily compute $l = k^{-1}$, that is, the multiplicative inverse of k in the group $\mathbb{Z}^*_{\phi(n)}$, so that her decoding function is $D(C) = C^l \pmod{n}$.
- Note that if $C = E(M)$, then $D(C) = M^{kl} = M \pmod{n}$ since $kl = 1 \pmod{\phi(n)}$.

Example

Alice chooses the following two (odd) primes: $17, 19$. (In real life, the primes would be **much much bigger**!)

So $n = 17 \cdot 19 = 323$.

In this case, $\phi(n) = \phi(323) =$ the number of integers i such that $\gcd(i, 323) = 1$, i.e., the number of integers relatively prime to 323, so
$\phi(323) = (17 - 1)(19 - 1) = 288$.

Alice must now choose a k in $\mathbb{Z}_{288}^*$, meaning that she must choose a k such that k has an inverse in $\mathbb{Z}_{288}$, meaning that there exists an l such that

$k \cdot l = 1 \pmod{288}$.

Let's suppose that she chooses $k = 5$, and she computes $l = 173$. Note that
$5 \cdot 173 = 865 = 3 \cdot 288 + 1$.

Therefore, Alice's **Public Key** is $\langle n, k \rangle = \langle 323, 5 \rangle$.

On the other hand, Alice's **Secret Key** is
$\langle n, l \rangle = \langle 323, 173 \rangle$.

Alice now advertises her Public Key $\langle 323, 5 \rangle$ to the world.

Suppose that Bob wishes to send the following message to Alice: "I LOVE YOU," using her Public Key.

First, Bob "translates" the message into numbers, following some established convention; for example, direct ASCII into integers translation:

I	73
	32
L	76
O	79
V	86
E	69
	32
Y	89
O	79
U	85

And now Bob sends the message letter by letter (of course, in real life, he would send much bigger chunks).

First he sends "I"

He takes 73, and computes

$$73^k \pmod{n} = 73^5 \pmod{323} = 99$$

and now Bob sends 99 to Alice. He does the same with each letter in our scheme.

Alice receives 99, and now wants to decrypt it. So she computes:

$$99^l \pmod{n} = 99^{173} \pmod{323} = 73$$

so she knows that the first letter of the message has ASCII number 73, and therefore it is an "I".