

## 4

## Software

**In this chapter you will learn about:**

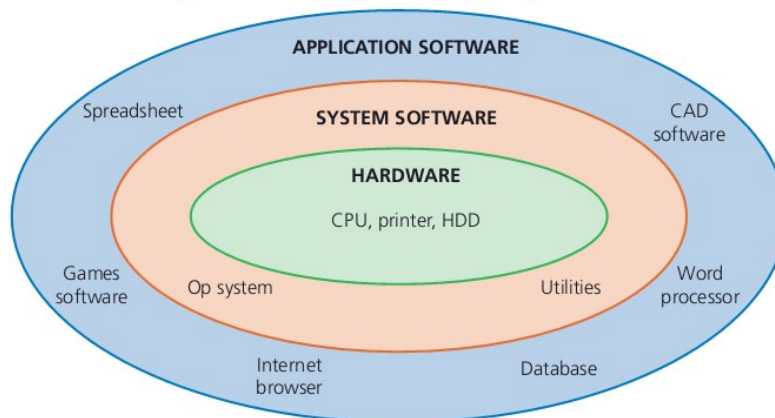
- ★ types of software and interrupts
  - the differences between systems software and applications software (including the operating system, utility programs and application software)
  - the role and basic functions of an operating system
  - the need for hardware, firmware and operating systems when running application software
  - role and operation of interrupts
- ★ types of programming language, translators and IDEs
  - advantages and disadvantages of high-level and low-level languages
  - assembly language is a low-level language that uses mnemonics and assemblers
  - the operation of compilers and interpreters for high-level languages
  - advantages and disadvantages of compilers and interpreters
  - the role and functions of integrated development environment (IDEs) when writing code.

In this chapter you will learn about some of the key software used in computer systems. The chapter will consider essential software (such as an operating system) all the way through to application software (such as word processors). The first part of the chapter will cover how the software is used, while the second part will cover how software is translated so that a computer can carry out the software's instructions.

## 4.1 Types of software and interrupts

### 4.1.1 System software and application software

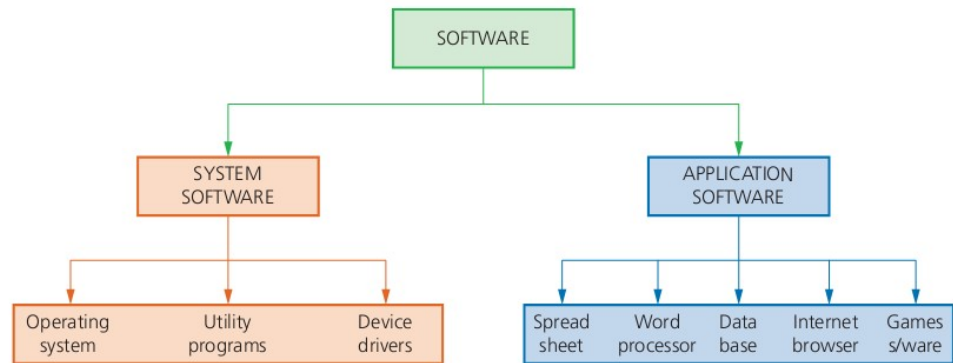
All computers begin life as a group of connected hardware items. Without software, the hardware items would be useless. This section considers the link between hardware and software. Figure 4.1 summarises the hierarchy of software and hardware.



▲ **Figure 4.1** Software and hardware hierarchy

## 4 SOFTWARE

You will notice from Figure 4.1 that there are two types of software: system software and application software:



▲ **Figure 4.2** Software types

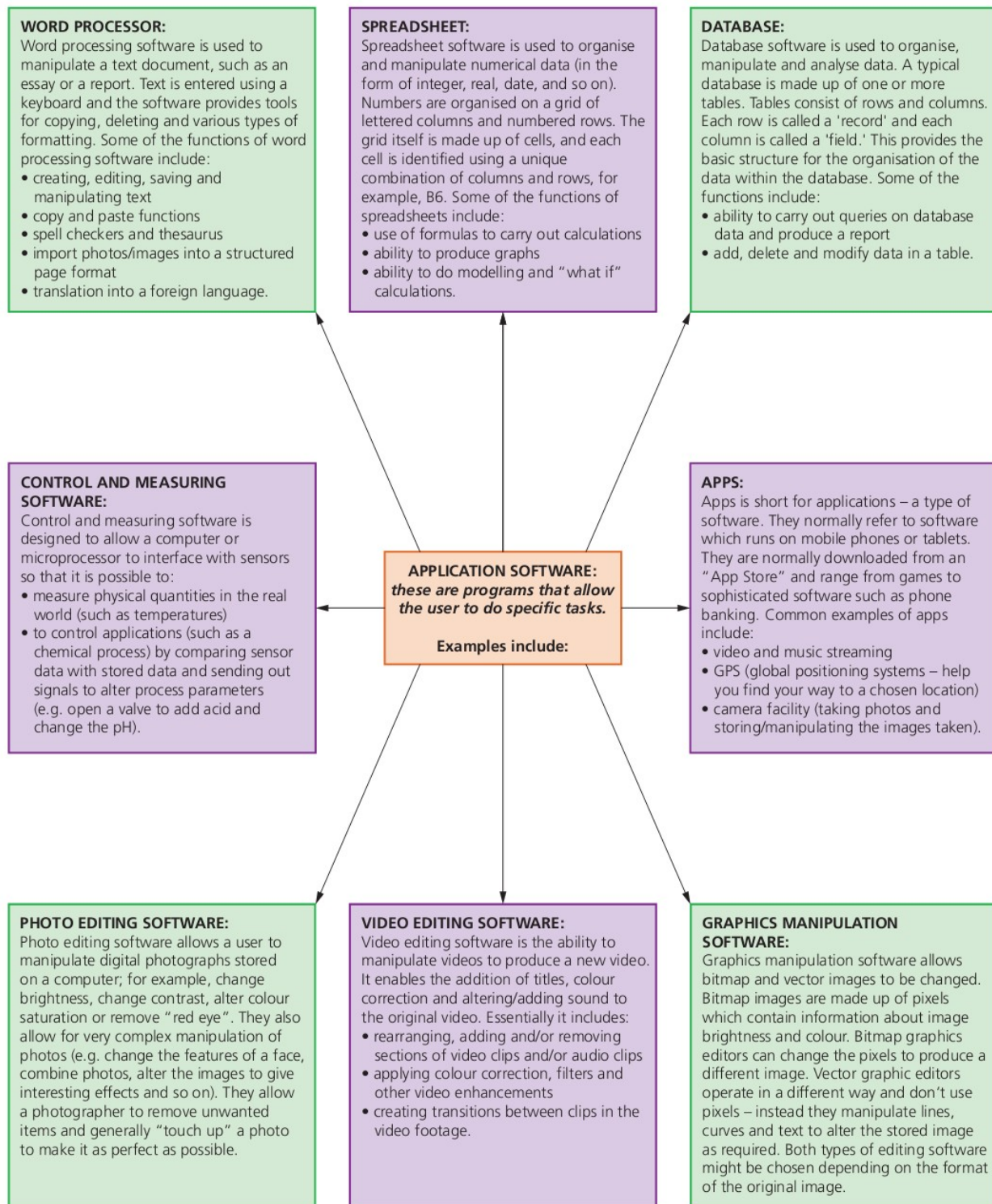
### General features of system software

- » set of programs to control and manage the operation of computer hardware
- » provides a platform on which other software can run
- » required to allow hardware and software to run without problems
- » provides a human computer interface (HCI)
- » controls the allocation and usage of hardware resources.

### General features of application software

- » used to perform various applications (apps) on a computer
- » allows a user to perform specific tasks using the computer's resources
- » may be a single program (for example, NotePad) or a suite of programs (for example, Microsoft Office)
- » user can execute the software as and when they require.

## Examples of typical application software

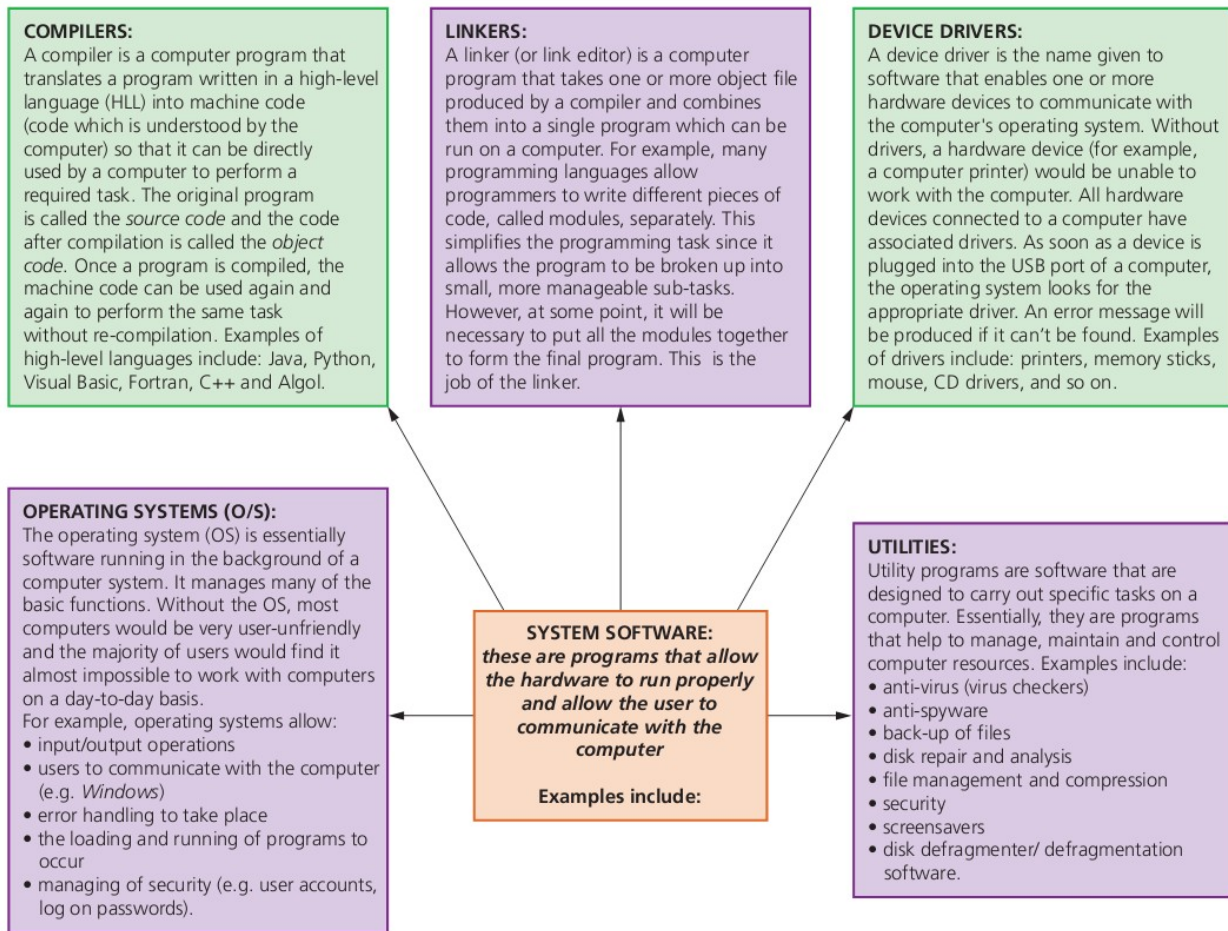


▲ Figure 4.3 Application software



## 4 SOFTWARE

### Examples of typical system software



▲ **Figure 4.4** System software

### Link

Refer to Section 4.1.3 on the running of apps on a computer.

The remainder of this section considers the role of the operating system, utility programs and device drivers in much more depth. Compilers and linkers will be considered later on in this book.

### Utility software (utilities)

Computer users are provided with a number of utility programs (often simply referred to as utilities) that are part of the system software.



Utility programs are often initiated by the user, but some, notably virus checkers, often just run in the background without the need for any user input. Utility programs offered by most computer system software include:

- » virus checkers
- » defragmentation software
- » disk contents analysis and repair
- » file compression and file management
- » back-up software
- » security
- » screensavers.

Virus checkers (anti-virus software)

Any computer (including mobile phones and tablets) can be subject to a virus attack. Operating systems offer virus checkers, but these must be kept thoroughly up to date and should run in the background to maintain their ability to guard against being infected by such **malware**. There are many ways to help prevent viruses (such as being careful when downloading material from the internet, not opening files or websites given in emails from unknown senders or by not using non-original software). However, virus checkers still afford the best defence against such malware.

Running **anti-virus software** in the background on a computer will constantly check for virus attacks. Although various types of anti-virus software work in different ways they all have the following common features:

- » they check software or files before they are run or loaded on a computer
- » anti-virus software compares a possible virus against a database of known viruses
- » they carry out **heuristic checking** – this is the checking of software for types of behaviour that could indicate a possible virus; this is useful if software is infected by a virus not yet on the database
- » any possible files or programs which are infected are put into **quarantine** which:
  - allows the virus to be automatically deleted, or
  - allows the user to make the decision about deletion (it is possible that the user knows that the file or program is not infected by a virus – this is known as a **false positive** and is one of the drawbacks of anti-virus software)
- » anti-virus software needs to be kept up to date since new viruses are constantly being discovered
- » full system checks need to be carried out once a week, for example, since some viruses lie dormant and would only be picked up by this full system scan.

### Link

See Section 5.3 for more on computer viruses.

### Link

Refer to Chapter 3 for more detail on how data is stored on a hard disk drive (HDD).

**Defragmentation software**

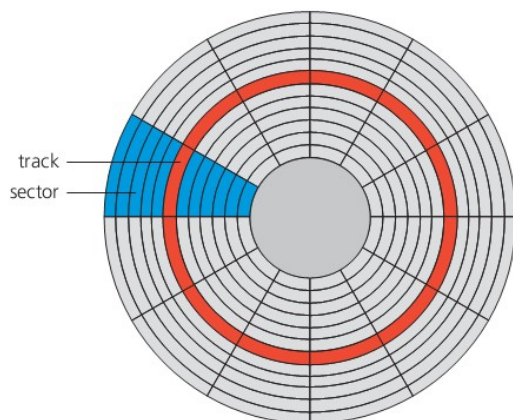
As a HDD becomes full, blocks used for files will become scattered all over the disk surface (in potentially different sectors and tracks as well as different surfaces). This will happen because files will become deleted, partially-deleted, extended and so on over time. The consequence of this is slower data access time; the HDD read-write head will now require several movements just to find and retrieve the data making up the required file.

It would obviously be advantageous if files could be stored in **contiguous** sectors considerably reducing HDD head movements. (Note that due to the different operation of SSDs when accessing data, this is not a problem when using solid state devices.)

contiguous means 'next to each other'

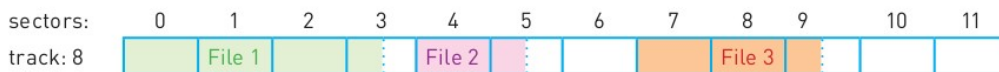
4 SOFTWARE

Consider the following scenario using a disk with 12 (numbered 0 to 11) sectors per surface:



▲ **Figure 4.5** Hard disk drive tracks and sectors

In this example we have three files (1, 2 and 3) stored on track 8 of a disk surface covering all 12 sectors:



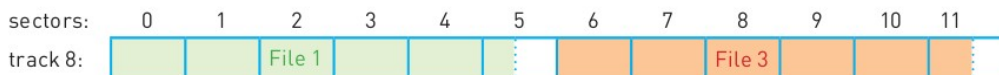
Now file 2 is deleted by the user and file 1 has data added to it; however, the file 2 sectors which become vacant are not filled up straight away by new file 1 data since this would require 'too much effort' for the HDD resources; we now get the following (file 1 is now stored in sectors 0, 1, 2, 3, 10 and 11):



File 1 has now been extended to write data in sectors 10 and 11; now suppose file 3 is extended with the equivalent of 3 blocks of data; this now requires filling up sector 9 and then finding some empty sectors to write the remainder of the data – suppose the next free sectors are on track 11:



If this continues, the files just become more and more scattered throughout the disk surfaces. It is possible for sectors 4, 5 and 6 (on track 8) to eventually become used if the disk starts to fill up and it has to use up whatever space is available. A **disk defragmenter** will rearrange the blocks of data to store files in **contiguous** sectors wherever possible. After defragmentation Track 8 would now become:



This obviously allows for much faster data access and retrieval since the HDD will now require fewer read-write head movements to access and read files 1 and 3. Track 11 would be empty after the defragmentation process.

Back-up software

While it is sensible to take manual back-ups using, for example, a memory stick or portable HDD, it is also good practice to use the operating system **back-up utility**. This utility will:

- » allow a schedule for backing up files to be made
- » only carry out a back-up procedure if there have been any changes made to a file.

For total security there should be three versions of a file:

- 1 the current (working) version stored on the internal HDD or SSD
- 2 a locally backed up copy of the file (stored on a portable SSD, for example)
- 3 a remote back-up version stored well away from the computer (for example, using cloud storage).

The Microsoft Windows environment offers the following facilities using the back-up utility:

- » restore data, files or the computer from the back-up (useful if there has been a problem and files have been lost and need to be recovered)
- » create a restore point (this is basically a kind of 'time machine' where your computer can be restored to its state at this earlier point in time; this can be very useful if a very important file has been deleted and can't be recovered by any of the other utilities)
- » options of where to save back-up files; this can be set up from the utility to ensure files are automatically backed up to a chosen device.

Windows uses **File History**, which takes snapshots of files and stores them on an external HDD at regular intervals. Over a period of time, **File History** builds up a vast library of past versions of files – this allows a user to choose which version of the file they want to use. **File History** defaults to backing up every hour and retains past versions of files for ever unless the user changes the settings.

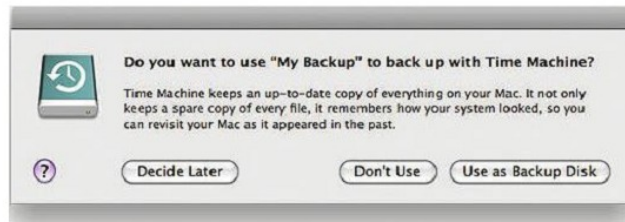
Mac OS offers the **Time Machine** back-up utility. This erases the contents of a selected drive and replaces them with the contents from the back-up. To use this facility, it is necessary to have an external HDD or SSD (connected via USB port) and ensure that the Time Machine utility is installed and activated on the selected computer. Time machine will automatically:

- » back-up every hour
- » do daily back-ups for the past month, and
- » weekly back-ups for all the previous months.

(Note: once the back-up HDD or SSD is almost full, the oldest back-ups are deleted and replaced with the newest back-up data.) The following screen shows the Time Machine message:



## 4 SOFTWARE



▲ **Figure 4.6** Time machine message on Mac OS

### Link

Refer back to Section 2.3 for more information on encryption and decryption.

### Security software

Security software is an over-arching utility that:

- » manages access control and user accounts (using user IDs and passwords)
- » links into other utility software, such as virus checkers and spyware checkers
- » protects network interfaces (for example, through the use of firewalls)
- » uses encryption and decryption to ensure any intercepted data is meaningless without a decryption key
- » oversees the updating of software (does the update request come from a legitimate source, for example).

### Screensavers

**Screensavers** are programs that supply moving and still images on the monitor screen after a period of inactivity by the computer. They were originally developed to protect older CRT (cathode ray tube) monitors which would suffer from 'phosphor burn' if the same screen image remained for any length of time. With modern LCD and OLED screens, this problem no longer exists; consequently, screensavers are now mostly just a way of customising a device. However, many screensavers are also used as part of the computer's security system. If a computer is unused for five minutes, for example, and hasn't been logged out, this will trigger the screensaver to be loaded. The computer user will then be automatically logged out and a screensaver will indicate that the computer is now locked. This gives an extra layer of security for computers used in an office environment, for example.

Some screensavers are often used to activate useful background tasks that can only go on when the computer is in an 'idle' state. For example:

- » virus scans
- » distributed computing applications – these allow apps to use the computer's resources only when it is idle (for example, an online gaming app).

### Device drivers

**Device drivers** are software that communicate with the operating system and translate data into a format understood by a hardware peripheral device. Without device drivers, a hardware device would be unable to work with a computer – a message such as 'device not recognised' would appear on the screen. As soon as a device is plugged into a USB port (for example, a memory stick, printer or camera), the operating system looks for the appropriate device driver.

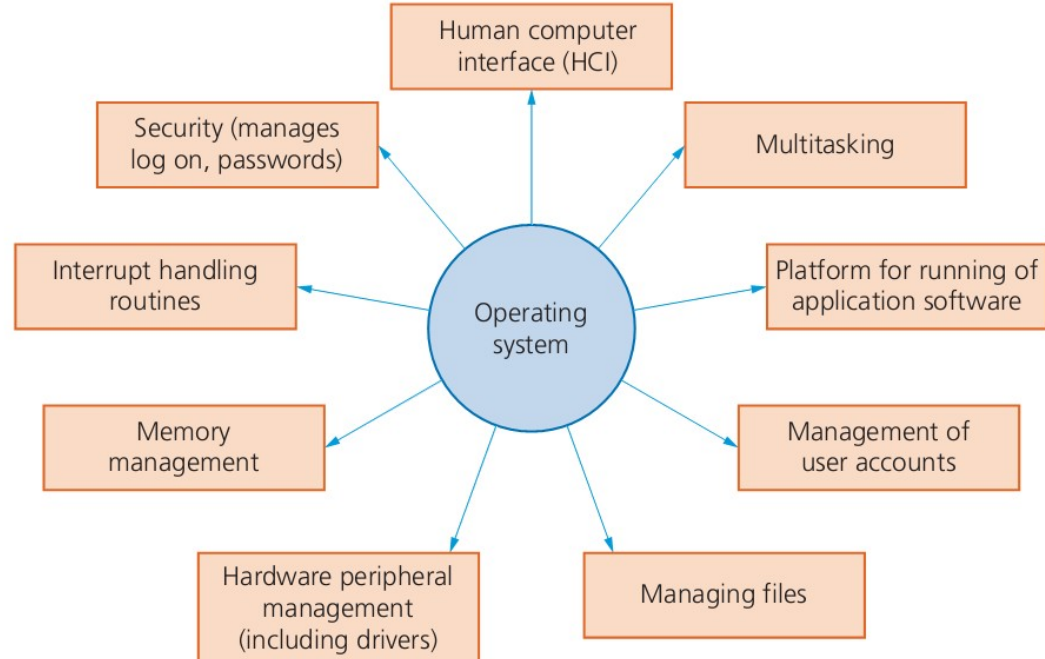
All USB device drivers contain a collection of information about devices called **descriptors**; this allows the USB bus to ask a newly connected device what it is. Descriptors include vendor id (VID), product id (PID) and unique serial numbers. If a device has no serial number associated with it, the operating system will treat the device as new every time it is plugged into a USB port. Serial numbers must be unique since this could prove rather interesting if two different devices with the same serial number were plugged into a computer at the same time.

### 4.1.2 Operating systems

To enable computer systems to function correctly and allow users to communicate with computer systems, software known as an **operating system** needs to be installed. An operating system provides both the environment in which applications can be run and a useable interface between humans and computer. An operating system also disguises the complexity of computer software and hardware. Common examples of operating systems include: Microsoft Windows, Apple Mac OS, Google Android and Apple IOS (the latter two being used primarily on tablets and smartphones).

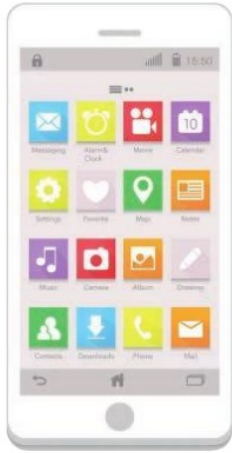
Most computers store the operating system on a hard disk drive (HDD) or solid state drive (SSD) since they tend to be very large programs. Mobile phones and tablets store the operating system on a solid state device since they are too small to accommodate an HDD.

Figure 4.7 summarises some of the functions in a typical operating system.



▲ **Figure 4.7** Operating system functions

## 4 SOFTWARE



▲ **Figure 4.8** GUI icons on a mobile phone

The next section describes each of the nine functions shown in Figure 4.7.

### Human computer interface (HCI)

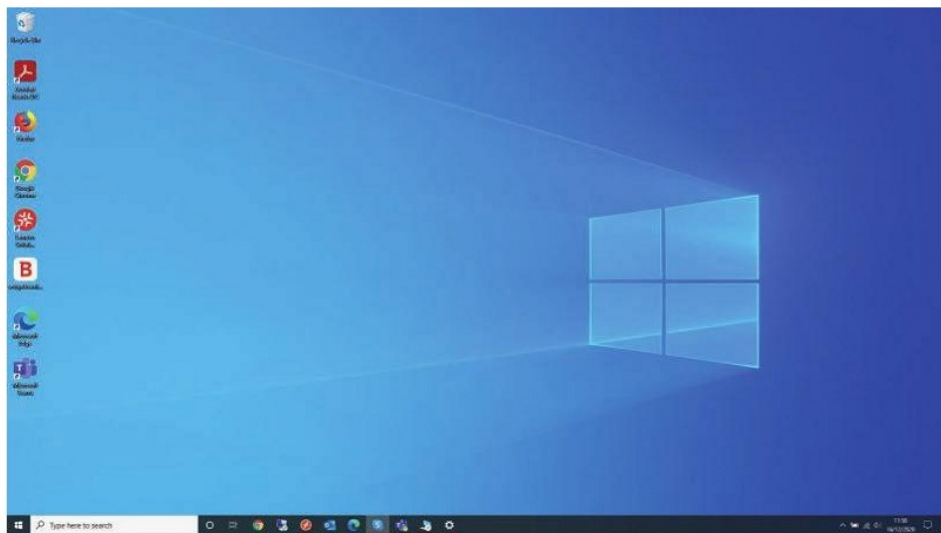
The **human computer interface (HCI)** is in the form of a **Command Line Interface (CLI)** or a **Graphical User Interface (GUI)**.

A CLI requires a user to type in instructions in order to choose options from menus, open software, etc. There are often a number of commands that need to be typed in, for example, to save or load a file. The user has to therefore learn a number of commands just to carry out basic operations. It is also slow having to key in these commands every time an operation has to be carried out. However, the advantage of CLI is that the user is in direct communication with the computer and is not restricted to a number of pre-determined options.

A GUI allows the user to interact with a computer (or MP3 player, gaming device, mobile phone, etc.) using pictures or symbols (icons) rather than having to type in a number of commands. Figure 4.8 shows a mobile screen with a number of GUI icons.

Simply selecting any of the icons from the screen would automatically load the application into the phone ready to be used. There is no need to type in anything.

GUIs use various technologies and devices to provide the user interface. One of the most common is **WIMP (windows icons menu and pointing device)**, which was developed for use on personal computers (PC). Here a mouse is used to control a cursor and icons are selected to open/run windows. Each window contains an application and modern computer systems allow several windows to be open at the same time. An example is shown below (here a number of icons can be seen on the left-hand side and on the bottom):



▲ **Figure 4.9** Windows screen showing icons

A windows manager looks after the interaction between windows, the applications, the pointing devices and the cursor's position.



More recently, devices such as mobile phones and tablets increasingly use touch screens and use **post-WIMP** interactions. With this system, fingers are in contact with the screen allowing actions such as pinching and rotating, which would be difficult to do using a single pointer and a device such as a mouse.

▼ **Table 4.1** Differences between GUI and CLI interfaces

Interface	Advantages	Disadvantages
command line interface (CLI)	<ul style="list-style-type: none"> <li>the user is in direct communication with the computer</li> <li>the user is not restricted to a number of pre-determined options</li> <li>it is possible to alter computer configuration settings</li> <li>uses a small amount of computer memory</li> </ul>	<ul style="list-style-type: none"> <li>the user needs to learn a number of commands to carry out basic operations</li> <li>all commands need to be typed in which takes time and can be error-prone</li> <li>each command must be typed in using the correct format, spelling, and so on</li> </ul>
graphical user interface (GUI)	<ul style="list-style-type: none"> <li>the user doesn't need to learn any commands</li> <li>it is more user-friendly; icons are used to represent applications</li> <li>a pointing device (such as a mouse) is used to click on an icon to launch the application – this is simpler than typing in commands or a touch screen can be used where applications are chosen by simply touching the icon on the screen</li> </ul>	<ul style="list-style-type: none"> <li>this type of interface uses up considerably more computer memory than a CLI interface</li> <li>the user is limited to the icons provided on the screen</li> <li>needs an operating system, such as Windows, to operate, which uses up considerable memory</li> </ul>

Who would use each type of interface?

- » CLI: a programmer, analyst or technician; basically somebody who needs to have a direct communication with a computer to develop new software, locate errors and remove them, initiate memory dumps (contents of the computer memory at some moment in time), and so on
- » GUI: the end-user who doesn't have or doesn't need to have any great knowledge of how the computer works; a person who uses the computer to run software or play games or stores/manipulates photographs, for example.

## Memory management

**Memory management** carries out the following functions:

- » manages the primary storage (RAM) and allows data to be moved between RAM and HDD/SSD during the execution of programs
- » keeps track of all the memory locations
- » carries out memory protection to ensure that two competing applications cannot use the same memory locations at the same time. If this wasn't done the following might happen:
  - data would probably be lost
  - applications could produce incorrect results (based on the wrong data being in memory locations)
  - potential security issues (if data is placed in the wrong location, it might make it accessible to other software, which would be a major security issue)
  - in extreme cases, the computer could crash.

## 4 SOFTWARE

### Security management

**Security management** is another part of a typical operating system; the function of security management is to ensure the integrity, confidentiality and availability of data. This can be achieved as follows (many of these features are covered in more depth elsewhere in this book):

#### Link

See Section 5.3 for more on cyber security.

- » by carrying out operating system updates as and when they become available
- » ensuring that anti virus software (and other security software) is always up to date, preserving the integrity, security and privacy of data
- » by communicating with, for example, a firewall to check all traffic to and from the computer
- » by making use of privileges to prevent users entering 'private areas' on a computer that permits multi-user activity (this is done by setting up user accounts and making use of passwords and user IDs); this helps to ensure the privacy of data
- » by maintaining access rights for all users
- » by offering the ability for the recovery of data (and system restore) when it has been lost or corrupted
- » by helping to prevent illegal intrusion into the computer system (also ensuring the privacy of data).



#### Find out more

By checking out the remainder of this chapter and Chapter 5, find out the methods available to ensure the security, privacy and integrity of data and how these link into the operating system security management. It is important to distinguish between what constitutes security, privacy and integrity of data.

### Hardware peripheral management

**Hardware management** involves all input and output peripheral devices. Hardware management:

#### Link

See Section 4.1.1 for more information on drivers.

- » communicates with all input and output devices using device drivers
- » uses a device driver to take data from a file (defined by the operating system) and translates it into a format that the input/output device can understand
- » ensures each hardware resource has a priority so that they can be used and released as required
- » manages input/output devices by controlling queues and **buffers**; consider the role of the printer management when printing out a document:
  - first of all, the printer driver is located and loaded into memory
  - then the data is sent to a printer buffer ready for printing
  - if the printer is busy (or the printing job has a low priority) then the data is sent to a printer queue before it can be sent to the printer buffer
  - it will send various control commands to the printer throughout the printing process
  - it receives and handles error messages and interrupts from the printer.

**Find out more**

Find out about the tasks carried out by a Keyboard Manager when a user is typing in the text to a word processor. Consider the use of buffers and queues in your answer.

You may need to do some research throughout this book to find out how the Keyboard Manager works.

**File management**

The main tasks of **file management** include:

file name

extension

- » file naming conventions which can be used i.e. **filename.docx** (where the extension can be .bat, .htm, .dbf, .txt, .xls, etc.)
- » performing specific tasks (for example, create, open, close, delete, rename, copy, and move)
- » maintaining the directory structures
- » ensuring access control mechanisms are maintained (for example, access rights to files, password protection, or making files available for editing or locking them)
- » ensuring memory allocation for a file by reading it from the HDD/SSD and loading it into memory.

**Interrupts**

Please refer to Section 4.1.4 for a discussion on interrupts.

**Platform for running of application software**

Please refer to Section 4.1.3 for a discussion on the running of application software.

**Multitasking**

**Multitasking** allows computers to carry out more than one task (i.e. a process) at a time. Each of the processes will share the hardware resources under the control of the operating system software. To make sure that multitasking operates correctly (in other words, the processes don't clash with each other), the operating system needs to constantly monitor the status of each of the processes under its control:

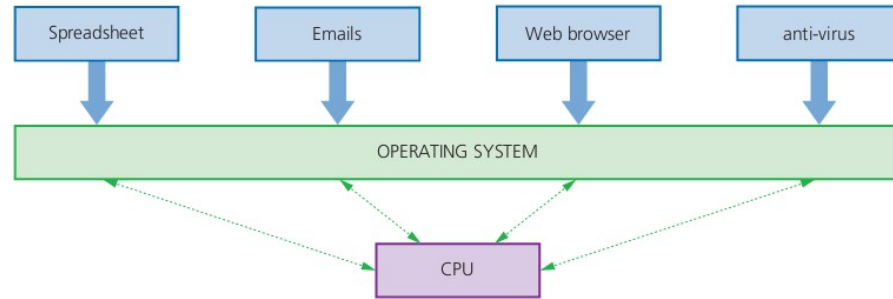
- » resources are allocated to a process for a specific time limit
- » the process can be interrupted while it is running
- » the process is given a priority so it can have resources according to its priority (the risk here is that a low priority process could be starved of resources).

these three bullet points are called pre-emptive multitasking

Using multitasking management, main memory, HDD/SSD and virtual memory are better managed making the most effective use of CPU time.



## 4 SOFTWARE



▲ **Figure 4.10** Multitasking diagram

### Management of user accounts

Computers allow more than one user to log onto the system. It is therefore important that users' data is stored in separate parts of the memory for security reasons (also refer to security management earlier in this section). Each person logging onto the computer will be given a user account protected by a user name and password. The operating system is given the task of managing these different user accounts. This allows each user to:

- » customise their screen layout and other settings
- » use separate folders and files and to manage these themselves.

Very often an **administrator** oversees the management of these user accounts. The administrator can create accounts, delete user accounts and restrict user account activity. On large university or industrial computers, part of the operating system's tasks will be to oversee several users' accounts, since a complex multi-user system may be in place. The operating system has to maintain accounts for several users, managing data that may range from personal data and technical research work down to the ordering of stationery. Multi-access levels permit this control to take place. For example, a clerk in the office may have access to ordering stationery but can't have access to any personal data.

### 4.1.3 Running of applications

This section will bring together some of the topics covered in Section 4.1.2. As mentioned earlier, application software requires the operating system to provide a platform on which the software can run successfully.

When a computer starts up, part of the operating system needs to be loaded into RAM – this is known as **booting up** the computer (or a **bootstrap loader**). The start-up of the computer's motherboard is handled by the basic input/output system (BIOS). The BIOS tells the computer where the storage device that holds the operating system can be found; it then loads the part of the operating system that is needed and executes it.

The BIOS is often referred to as **firmware**. Firmware is defined as a program that provides low level control for devices.

**Advice**

EEPROM is included to fully explain the BIOS, but details about EEPROM go beyond the requirements of the syllabus.

The BIOS program is stored in a special type of ROM, called an **EEPROM** (Electrically Erasable Programmable ROM). EEPROM is a flash memory chip, which means its contents remain even when the computer is powered down. However, it also means the BIOS can be rewritten, updated or even deleted by a user.

However, while the BIOS is stored on an EEPROM, the BIOS **settings** are stored on a CMOS chip (Complementary Metal Oxide Semi-conductor). The CMOS is powered up at all times via a rechargeable battery on the motherboard. Therefore, the BIOS settings would be reset if the battery was removed or disconnected for some reason. Once the CMOS is re-started, it will access the same BIOS program from EEPROM, but the settings will now be the default factory settings. Consequently, if a user has changed the BIOS settings (for example, the clock speed), the settings will revert to those settings made at the factory once power is restored to the CMOS.



▲ **Figure 4.11** Firmware interface between OS and hardware

The application software will be under the control of the operating system and will need to access system software such as the device drivers while it is running. Different parts of the operating system may need to be loaded in and out of RAM as the software runs.

### 4.1.4 Interrupts

An **interrupt** is a signal sent from a device or from software to the microprocessor. This will cause the microprocessor to temporarily stop what it is doing so that it can service the interrupt. Interrupts can be caused by:

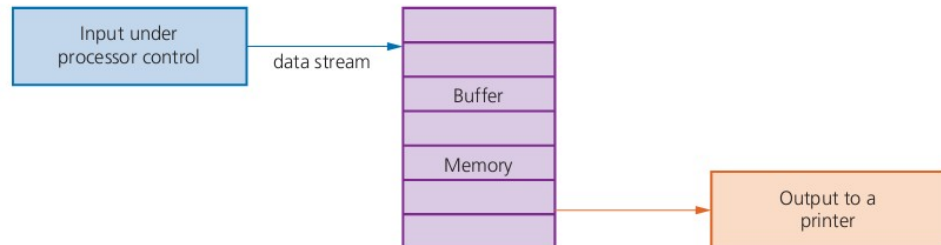
- » a timing signal
- » an input/output process (for example, a disk drive or printer requiring more data)
- » a hardware fault (for example, a paper jam in the printer)
- » user interaction (for example, the user presses a key (or keys) on a keyboard, such as <CTRL><ALT><BREAK>, which causes the system to be interrupted)
- » software errors that cause a problem (for example, an .exe file that cannot be found to initiate the execution of a program, two processes trying to access the same memory location, or an attempt to divide by zero).

Once the interrupt signal is received, the microprocessor either carries on with what it was doing or stops to service the device or program that caused the interrupt. The computer needs to identify the interrupt type and also establish the level of **interrupt priority**.

Interrupts allow computers to carry out many tasks or to have several windows open at the same time. An example would be downloading a file from the internet at the same time as listening to some music from a library. Interrupts allow these two functions to co-exist and the user has the impression that both functions are being carried out simultaneously. In reality, data is being passed in and out of memory very rapidly allowing both functions to be serviced. This can all be achieved by using an area in memory known as a **buffer**. A buffer is a memory area that stores data temporarily (see Figure 4.12). For example, buffers

## 4 SOFTWARE

are used when downloading a movie from the internet to compensate for the difference between download speeds and the data requirements of the receiving device. The data transmission rate of the movie file from the web server to the buffer must be greater than the rate at which data is transmitted from buffer to media player. Without buffers, the movie would frequently 'freeze'.



▲ **Figure 4.12** Use of a buffer when sending data to a printer (buffer used to store data temporarily since printer speed is much slower than microprocessor speed)



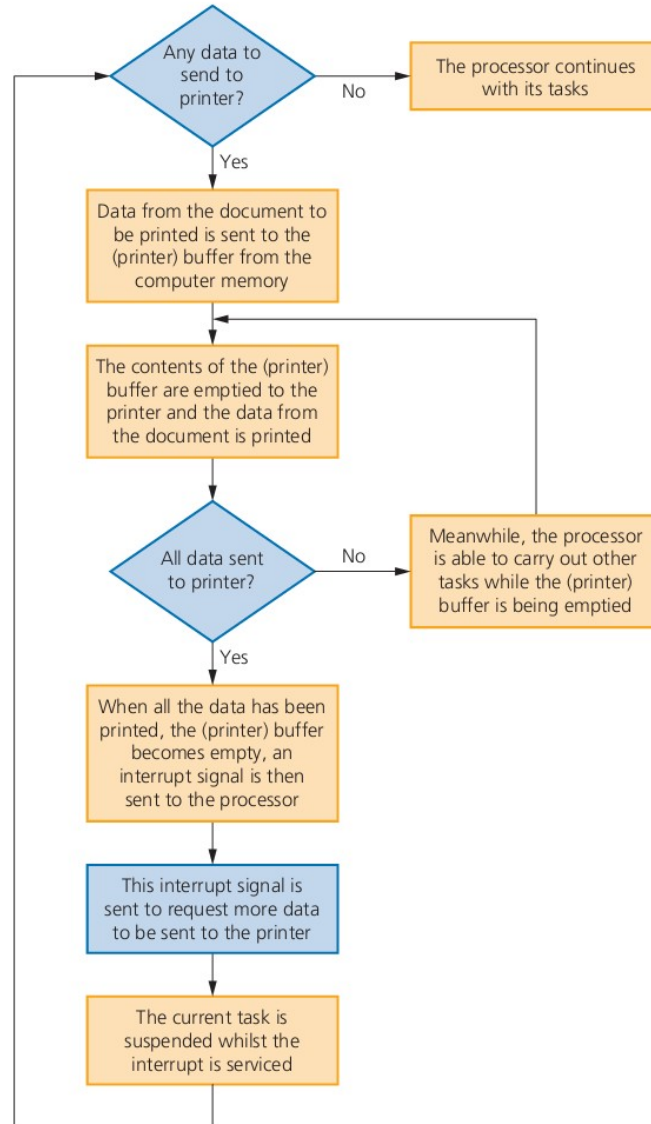
### Find out more

Find out how buffers are used to stream movies from the internet to a device (such as a tablet).

Whenever an interrupt is received it needs to be **serviced**. The status of the current task being run first needs to be saved. The contents of the Program Counter (PC) and other registers are saved. Then the **interrupt service routine (ISR)** is executed by loading the start address into the Program Counter (PC). Once the interrupt has been fully serviced, the status of the interrupted task is reinstated (the contents of all the saved registers are then retrieved) and the process continues.

Buffers and interrupts are often used together to allow standard computer functions to be carried out. These functions are often taken for granted by users of modern computer systems. For example, the following diagram (Figure 4.13) shows how buffers and interrupts are used when a document is sent from memory to a printer. The important thing to remember here is the time taken to print out a document is **much** longer than the time it takes for the microprocessor to send data to the printer. Without buffers and interrupts, the microprocessor would remain idle waiting for a document to be printed. This would be an incredible waste of microprocessor time; the buffers and interrupts allow the microprocessor to carry on with other tasks while the document is being printed thus maximising its processing power and speed.





▲ **Figure 4.13** Use of interrupts and buffers when printing a document

### ➔ Find out more

Try to produce a flow chart (similar to Figure 4.13) that shows the role of buffers and interrupts when the memory sends data to a disk drive for storage.

Remember that the time to write data to disk is much longer than the time it takes for the microprocessor to carry out its tasks.

## 4 SOFTWARE

### Activity 4.1

- 1 Tick (✓) the appropriate column in the following table to indicate whether the named software is system software or application software.

Software	System software	Application software
screensaver		
anti-virus software		
control and measurement software		
printer driver		
video editing software		
compiler		
QR code reader		
on-screen calculator		
operating system software		

- 2 Mike is downloading a video from the internet to his laptop. The speed of data transfer from the internet is slower than the speed at which data is being sent to the media player.
- What could be used to stop the video constantly freezing while Mike is watching it on his laptop?
  - While watching the video, Mike is meanwhile printing a 160-page document on his inkjet printer. Describe how interrupts could be used to allow him to watch his movie at the same time as the printing is being done. The printer memory can store up to 20 pages at a time.
  - Describe what happens if the inkjet printer runs out of black ink during the printing process.
- 3 Choose four features of an operating system and describe their function.
- 4 What is meant by a **descriptor** in a device driver? What role does the descriptor play when a new memory stick, for example, is plugged into a USB port of a computer for the first time?
- 5 **a** Describe what is meant by a **BIOS** and state its function. What is the task of a BIOS when a computer is first powered up?
- b** BIOS software and BIOS settings are different. Describe the different types of memory needs for both the software and its settings. In your explanation state why both types of memory are used.
- 6 Seven descriptions are shown on the left and seven computer terms are shown on the right. Draw lines to connect each description to the correct computer term.

when a computer starts up, OS is loaded into RAM

software that communicates with the OS and translates data into a format that can be understood by an I/O device

computer carrying out many different processes at the same time

program that provides low level control for devices including embedded systems

program that supplies static or moving images on a monitor when a computer has been idle for a period of time

signal from a device or software sent to a microprocessor to temporarily halt the process currently being carried out

memory area used to hold data temporarily

firmware

printer driver

bootstrap loader

interrupt

screensaver

buffer

multitasking

## 4.2 Types of programming language, translators and integrated development environments (IDEs)

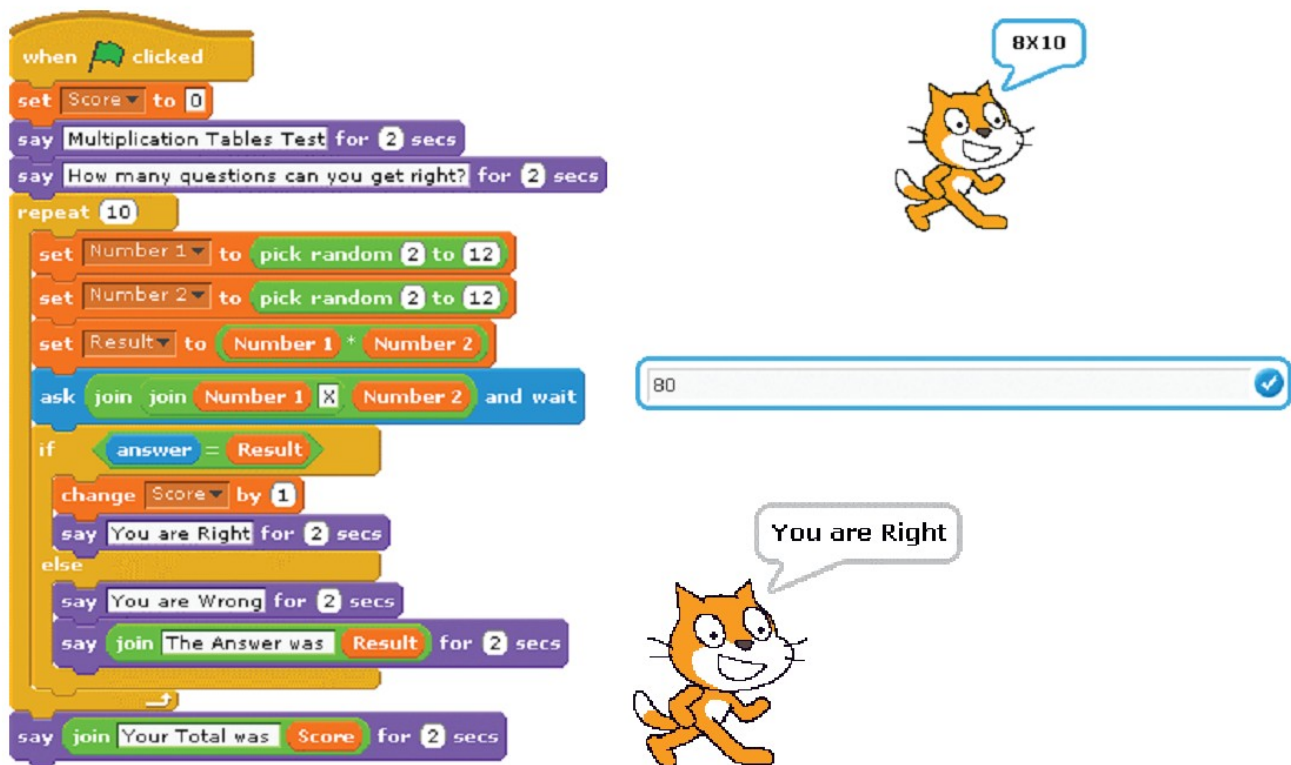
### 4.2 Types of programming language, translators and integrated development environments (IDEs)

People use many different languages to communicate with each other. In order for two people to understand each other they need to speak the same language or another person, an interpreter, is needed to translate from one language to the other language.

Programmers use many different programming languages to communicate with computers. Computers only 'understand' their own language, called **machine code**. A program needs to be translated into machine code before it can be 'understood' by a computer.

Programs are our way of telling a computer what to do, how to do it and when to do it. This enables a single computer to perform many different types of task. A computer can be used to stream videos, write reports, provide weather forecasts and many, many other jobs.

Here is an example of a simple task that can be performed by a computer:



▲ Figure 4.14 A Scratch multiplication table test program

#### Find out more

Find at least ten different tasks that computer programs perform in your school.



## 4 SOFTWARE

A **computer program** is a list of instructions that enable a computer to perform a specific task. Computer programs can be written in **high-level languages** and **low-level languages** depending on the task to be performed and the computer to be used. Most programmers write programs in high-level languages.

### 4.2.1 High-level languages and low-level languages

#### High-level languages

High-level languages enable a programmer to focus on the problem to be solved and require no knowledge of the hardware and instruction set of the computer that will use the program. Many high-level programming languages are portable and can be used on different types of computer.

High-level languages are designed with programmers in mind; programming statements are easier to understand than those written in a low-level language. This means that programs written in a high-level language are easier to:

- » read and understand as the language used is closer to English
- » write in a shorter time
- » debug at the development stage
- » maintain once in use.

The following snippet of program to add two numbers together is a single program statement written in a typical high-level language. It shows how easy it is to understand what is happening in a high-level language program:

```
Sum := FirstNumber + SecondNumber
```

There are many different high-level programming languages in use today including C++, Delphi, Java, Pascal, Python, Visual Basic and many more. Once a programmer has learned the techniques of programming in any high-level language, these can be transferred to writing programs in other high-level languages.



#### Find out more

High-level programming languages are said to be 'problem oriented'. What type of problems are the languages named above used for? Find out about five more high-level languages. Name each programming language and find out what it is used for.

#### Low-level languages

Low-level languages relate to the specific architecture and hardware of a particular type of computer. Low-level languages can refer to **machine code**, the binary instructions that a computer understands, or **assembly language** that needs to be translated into machine code.

Machine code

Programmers do not usually write in machine code as it is difficult to understand, and it can be complicated to manage data manipulation and storage.

The following snippet of program to add two numbers together is written in typical machine code, shown in both hexadecimal and binary, and consists of three statements:

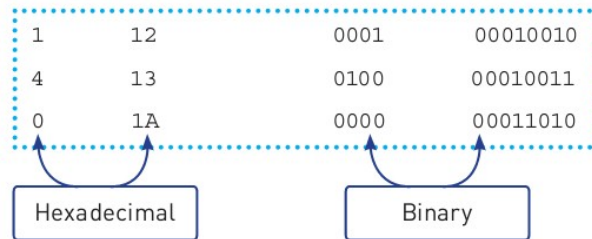
#### Link

For more on instruction sets, see Section 3.1.4.

#### Link

For more on hexadecimal see Section 1.1.2.

## 4.2 Types of programming language, translators and integrated development environments (IDEs)



As you can see, this is not easy to understand in binary! Machine code is usually shown in hexadecimal.



### Find out more

Find out about two different types of machine code. Name each chip set the machine code is used for and find the codes for load, add and store.

▼ **Table 4.2** Differences between high-level and low-level languages

Language	Advantages	Disadvantages
<b>High-level</b>	<ul style="list-style-type: none"> <li>independent of the type of computer being used</li> <li>easier to read, write and understand programs</li> <li>quicker to write programs</li> <li>programs are easier and quicker to debug</li> <li>easier to maintain programs in use</li> </ul>	<ul style="list-style-type: none"> <li>programs can be larger</li> <li>programs can take longer to execute</li> <li>programs may not be able to make use of special hardware</li> </ul>
<b>Low-level</b>	<ul style="list-style-type: none"> <li>can make use of special hardware</li> <li>includes special machine-dependent instructions</li> <li>can write code that doesn't take up much space in primary memory</li> <li>can write code that performs a task very quickly</li> </ul>	<ul style="list-style-type: none"> <li>it takes a longer time to write and debug programs</li> <li>programs are more difficult to understand</li> </ul>

### 4.2.2 Assembly languages

Fewer programmers write programs in an assembly language. Those programmers who do, do so for the following reasons:

- » to make use of special hardware
- » to make use of special machine-dependent instructions
- » to write code that doesn't take up much space in primary memory
- » to write code that performs a task very quickly.

The following snippet of program to add two numbers together is written in a typical assembly language and consists of three statements:

```
LDA      First
ADD      Second
STO      Sum
```

## 4 SOFTWARE

In order to understand this program, the programmer needs to know that:

- » **LDA** means load value of the variable (in this case, **First**) into the accumulator
- » **ADD** means add value of variable (in this case, **Second**) to the value stored in the accumulator
- » **STO** replace the value of the variable (in this case, **Sum**) by the value stored in the accumulator

Assembly language needs to be translated into machine code using an assembler in order to run. See the next section for more details.



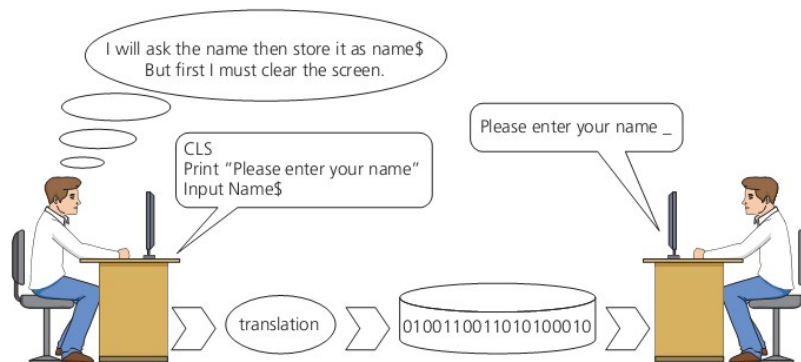
### Find out more

Find out about two assembly languages. Name each assembly language and find out what type of computer it is used for.

## 4.2.3 Translators

Computer programs can exist in several forms.

Programs are written by humans in a form that people who are trained as computer programmers can understand. In order to be used by a computer, programs need to be translated into the binary instructions, machine code, that the computer understands. Humans find it very difficult to read binary, but computers can only perform operations written in binary.



▲ **Figure 4.15** Translation

A program must be translated into binary before a computer can use it; this is done by a utility program called a **translator**. There are several types of translator program in use; each one performs a different task.

### Compilers

A **compiler** is a computer program that translates an entire program written in a high-level language (HLL) into machine code all in one go so that it can be directly used by a computer to perform a required task. Once a program is compiled the machine code can be used again and again to perform the same task without re-compilation. If errors are detected, then an error report is produced instead of a compiled program.

The high-level program statement:

```
Sum := FirstNumber + SecondNumber
```



## 4.2 Types of programming language, translators and integrated development environments (IDEs)

becomes the following machine code instructions when translated:

```
0001      00010010
0100      00010011
0000      00011010
```

### Interpreters

An **interpreter** is a computer program that reads a statement from a program written in a high-level language, translates it, performs the action specified and then does the same with the next statement and so on. If there is an error in the statement then execution ceases and an error message is output, sometimes with a suggested correction.

A program needs to be interpreted again each time it is run.

### Assemblers

An **assembler** is a computer program that translates a program written in an assembly language into machine code so that it can be directly used by a computer to perform a required task. Once a program is assembled the machine code can be used again and again to perform the same task without re-assembly.

The assembly language program statements:

```
LDA      First
ADD      Second
STO      Sum
```

become the following machine code instructions when translated:

```
0001      00010010
0100      00010011
0000      00011010
```

▼ **Table 4.3** Translation programs summary

Compiler	Interpreter	Assembler
Translates a high-level language program into machine code.	Executes a high-level language program one statement at a time.	Translates a low level assembly language program into machine code.
An executable file of machine code is produced.	No executable file of machine code is produced.	An executable file of machine code is produced.
One high-level language statement can be translated into several machine code instructions.	One high-level language program statement may require several machine code instructions to be executed.	One low-level language statement is usually translated into one machine code instruction.
Compiled programs are run without the compiler.	Interpreted programs cannot be run without the interpreter.	Assembled programs are used without the assembler.
A compiled program is usually distributed for general use.	An interpreter is often used when a program is being developed.	An assembled program is usually distributed for general use.

## 4 SOFTWARE

### 4.2.4 Advantages and disadvantages of compilers and interpreters

The advantages and disadvantages of compilers and interpreters are compared in Table 4.4.

▼ **Table 4.4** Comparing translators

Translators	Advantages	Disadvantages
<b>Interpreter</b>	easier and quicker to debug and test programs during development easier to edit programs during development	programs cannot be run without the interpreter programs can take longer to execute
<b>Compiler</b>	a compiled program can be stored ready for use a compiled program can be executed without the compiler a compiled program takes up less space in memory when it is executed a compiled program is executed in a shorter time	it takes a longer time to write, test and debug programs during development

### 4.2.5 Integrated Development Environment (IDE)

An Integrated Development Environment (IDE) is used by programmers to aid the writing and development of programs. There are many different IDEs available; some just support one programming language, others can be used for several different programming languages. You may be using PyCharm (for Python), Visual Studio (for Visual Basic) or BlueJ (for Java) as your IDE.



#### Find out more

Find out which programming language and IDE you are using in school and see if there are any other IDEs available for your programming language.

IDEs usually have these features:

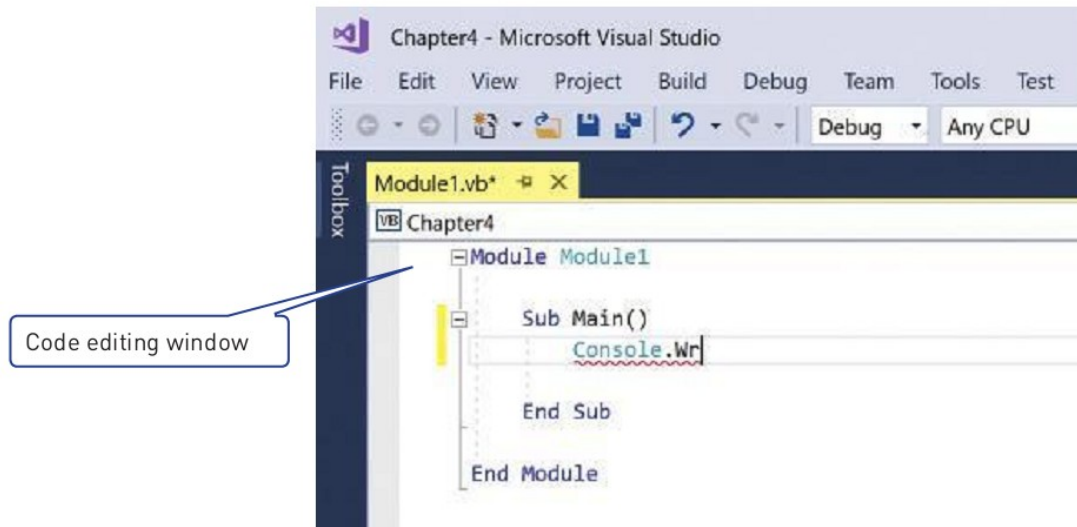
- » code editors
- » a translator
- » a runtime environment with a debugger
- » error diagnostics
- » auto-completion
- » auto-correction
- » an auto-documenter and prettyprinting.

Let's look at each of these features in turn and see how they help the development process.

## 4.2 Types of programming language, translators and integrated development environments (IDEs)

### Code editor

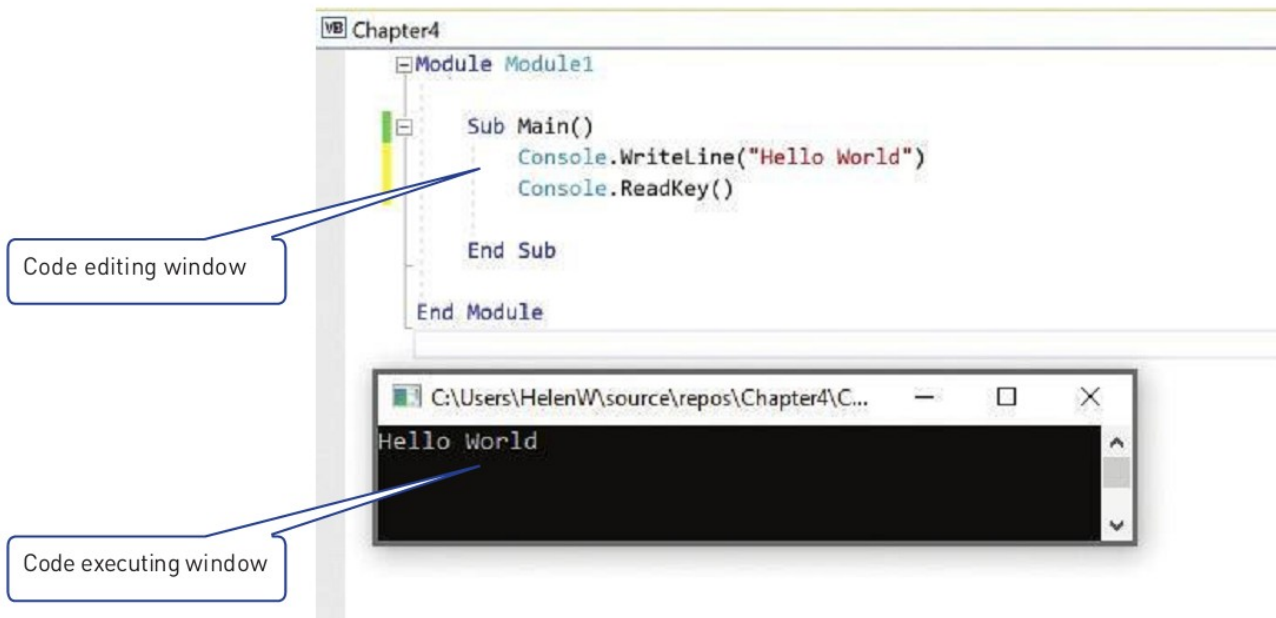
A code editor allows a program to be written and edited without the need to use a separate text editor. This speeds up the program development process, as editing can be done without changing to a different piece of software each time the program needs correcting or adding to.



▲ Figure 4.16 Visual Studio code editor

### Translator

Most IDEs usually provide a translator, this can be a compiler and/or an interpreter, to enable the program to be executed. The interpreter is often used for developing the program and the compiler to produce the final version of the program to be used.



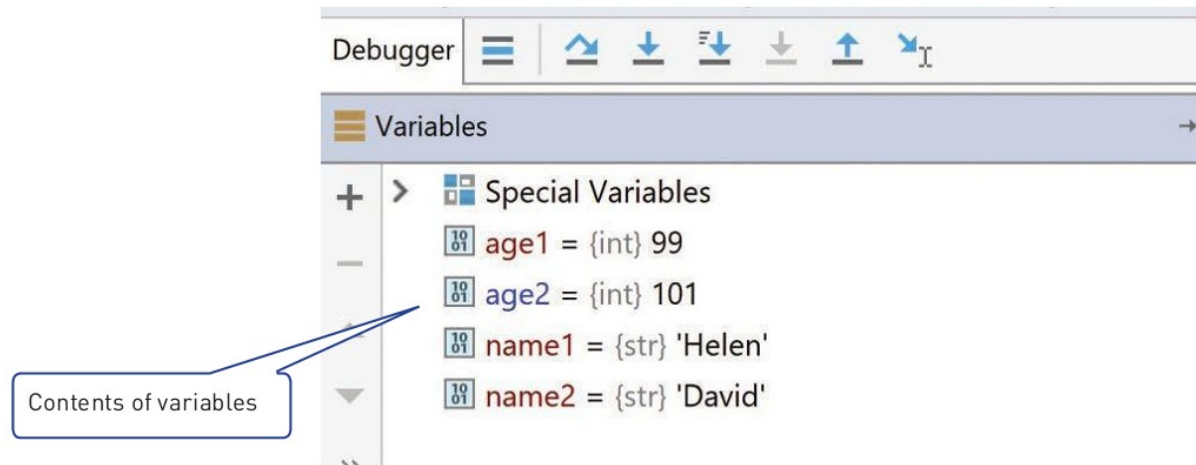
▲ Figure 4.17 Visual Studio code editor and program running



## 4 SOFTWARE

### A runtime environment with a debugger

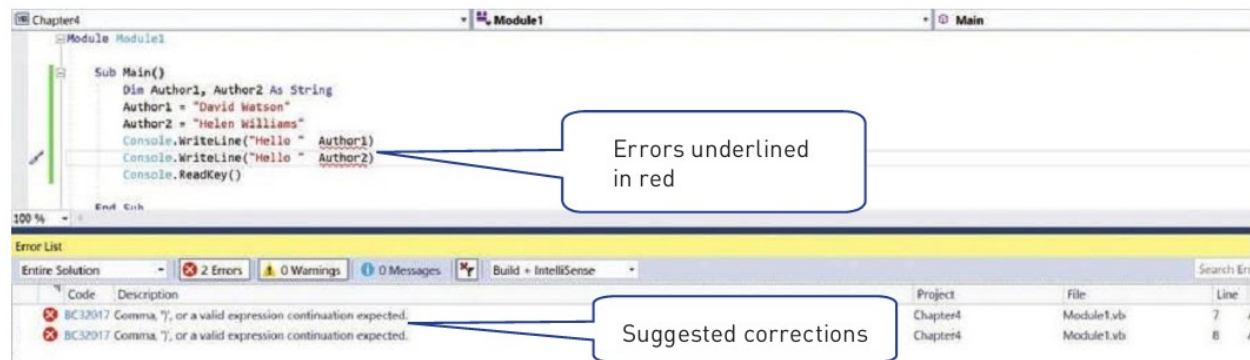
A debugger is a program that runs the program under development and allows the programmer to step through the program a line at a time (single stepping) or to set a breakpoint to stop the execution of the program at a certain point in the source code. A report window then shows the contents of the variables and expressions evaluated at that point in the program. This allows the programmer to see if there are any logic errors in the program and check that the program works as intended.



▲ Figure 4.18 PyCharm debugger

### Error diagnostics and auto-correction

Dynamic error checking finds possible errors as the program code is being typed, alerts the programmer at the time and provides a suggested correction. Many errors can therefore be found and corrected during program writing and editing before the program is run.

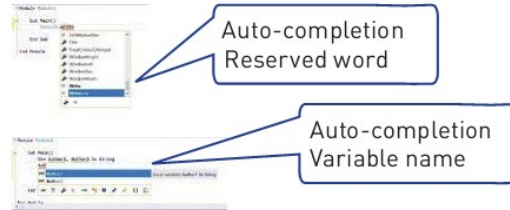


▲ Figure 4.19 Visual Studio error list with suggested corrections

4.2 Types of programming language, translators and integrated development environments (IDEs)

### Auto-completion

Code editors can offer context-sensitive prompts with text completion for variable names and reserved words.



▲ **Figure 4.20** Visual Studio showing auto-completion

Auto-documenter explaining the purpose of Console.WriteLine



▲ **Figure 4.21** Visual Studio auto-documenter

### Auto-documenter and prettyprinting

IDEs can provide an auto-documenter to explain the function and purpose of programming code.

Most code editors colour code the words in the program and lay out the program in a meaningful way – this is called **prettyprinting**.



▲ **Figure 4.22** Visual Studio code editor showing prettyprinting

**Find out more**

Find out which of these features are available in the IDE you are using in school.

### Activity 4.2

1 Tick (✓) the appropriate column in the following table to indicate whether the statement about the translator is True or False.

	True	False
An assembler translates a high-level language program.		
It is more difficult to write a program in a low-level language.		
Java is an assembly language.		
It is quicker to develop a program using a high-level language.		
You always need a compiler to run a compiled program.		
A program that is interpreted takes a longer time to run than a compiled program.		
Low-level languages are machine dependent.		

- 2 a Suki is writing a program in a high-level language. Describe three features of an IDE that she would find helpful.
- b Describe the difference between a compiler and an interpreter.
- c Explain why a programmer would choose to write a program in assembly language.

## 4 SOFTWARE

### Extension

For those students considering the study of this subject at A Level, the following shows how interrupts are used in the Fetch–(Decode)–Execute cycle.

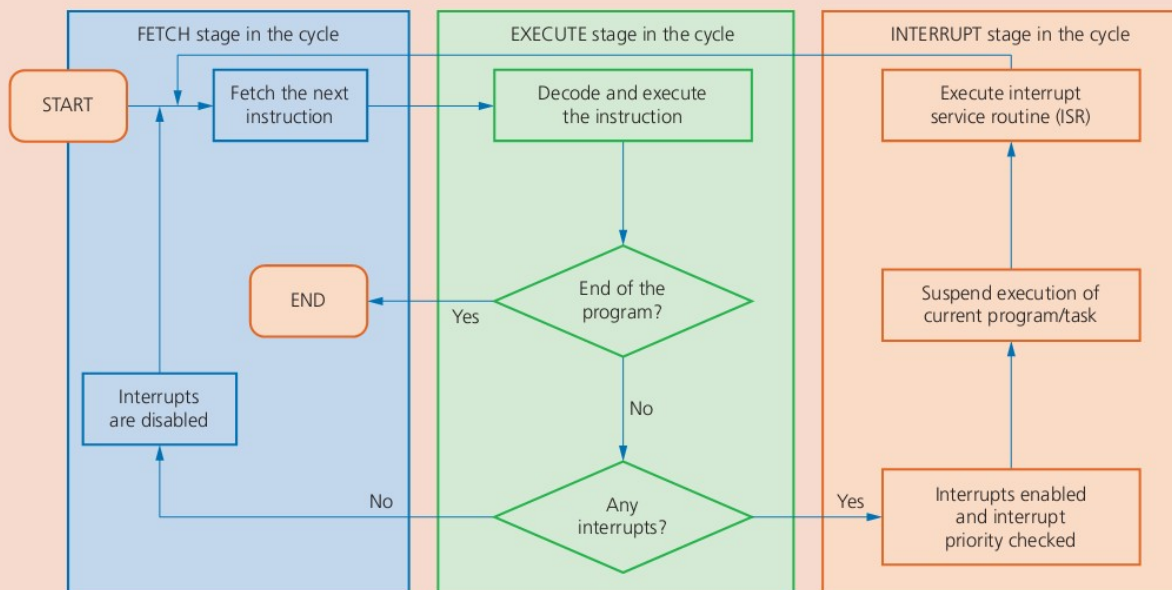
#### Use of interrupts in the Fetch–Execute cycle

The following figure shows a general overview of how a computer uses interrupts to allow it to operate efficiently and to allow it, for example, to carry out multi-tasking functions. Just before we discuss interrupts in this general fashion, the following notes explain how interrupts are specifically used in the Fetch–Execute cycle.

A special register called the interrupt register is used in the Fetch–Execute cycle. While the CPU is in the middle of carrying out this cycle, an interrupt could occur that will cause one of the bits in the interrupt register to change its status. For example, the initial status might be 0000 0000 and a fault might occur while writing data to the hard drive; this would cause the register to change to 0000 1000. The following sequence now takes place:

- » at the next Fetch–Execute cycle, the interrupt register is checked bit by bit
- » the contents 0000 1000 would indicate an interrupt occurred during a previous cycle and it still needs servicing; the CPU would now service this interrupt or 'ignore' it for now depending on its priority
- » once the interrupt is serviced by the CPU, it stops its current task and stores the contents of its registers
- » control is now transferred to the interrupt handler (or interrupt service routine, ISR)
- » once the interrupt is fully serviced, the register is reset and the contents of registers are restored.

The following flow diagram summarises the interrupt process during the Fetch–Execute cycle:



▲ Figure 4.23 The interrupt process in the Fetch–Execute cycle



## 4.2 Types of programming language, translators and integrated development environments (IDEs)

In this chapter, you have learnt about:

- ✓ system software
- ✓ application software
- ✓ utility programs
- ✓ the role and the function of an operating system
- ✓ how hardware, firmware and operating systems are used to run application software
- ✓ the role and operation of interrupts
- ✓ types of programming language – high-level and low-level
- ✓ translation software – compilers, translators and assemblers
- ✓ an Integrated Development Environment (IDE).

### Key terms used throughout this chapter

**utility programs (utilities)** – part of an operating system which carries out certain functions such as virus checking, defragmentation and screensaver

**malware** – programs (such as viruses, worms and Trojan horses) that are installed on a user's computer with the aim of deleting, corrupting or manipulating data illegally

**anti-virus software** – software that quarantines and deletes files or programs infected by a computer virus; the software can run in the background or be initiated by the user

**heuristic checking** – checking software for behaviour that could indicate a possible virus

**quarantine** – to isolate (in order to later delete) a file or program identified by anti-virus software as being infected by a virus

**defragmentation** – a process that reorganises sectors on an HDD by rearranging blocks of data so that they are contiguous

**contiguous** – next to each other

**back-up** – make copies of files onto another storage media in case the original file becomes corrupted or is deleted

**screensaver** – software that supplies a still or moving image on a monitor if a computer has been inactive for a period of time

**device driver** – software that communicates with the operating system and translates data into a format understood by the device

**descriptor** – a collection of information about a device plugged into a USB port; this can be vendor ID (VID), product ID (PID) or serial number

**operating system** – software that provides an environment in which applications can run and also provides an interface between computer and human operator

**boot up/bootstrap loader** – a small program that is used to load other programs to correctly 'start-up' a computer system

**EEPROM** – stands for electronically erasable programmable ROM

**human computer interface (HCI)** – an interface supplied by the operating system to 'hide' the complexities of the software and hardware from the human user

**command line interface (CLI)** – an interface which allows communication with the computer by typing in commands using a keyboard

**graphical user interface (GUI)** – an interface that uses icons to represent apps and tasks which the user can select/launch by clicking on a mouse or using a touch screen

**windows icons menu and pointing device (WIMP)** – an interface that uses a pointing device such as a mouse to select options from screen icons or a menu

**post-WIMP** – a modern touch screen interface system that allows actions such as pinching and rotating

**memory management** – the part of an operating system that controls main memory

**security management** – the part of an operating system that ensures the integrity, confidentiality and availability of data

**hardware management** – the part of an operating system that controls all input and output devices; it is made up of sub-systems such as printer management

**buffer** – a memory area used to store data temporarily

**file management** – part of an operating system that manages files in a computer (for example, the ability to create, delete, copy, open, close and rename files)

**interrupt** – a signal sent from a device or software to a microprocessor requesting its attention; the microprocessor suspends all operations until the interrupt has been serviced

## 4 SOFTWARE

**multitasking** – a function that allows a computer to process more than one task/process at a time

**administrator** – a person responsible for the upkeep and maintenance of a computer system that involves multi-user function

**user account** – an agreement that allows an individual to use a computer; the user needs a user name and password to enter the user's area

**error handling routine** – a routine in a program or operating system that recognises and recovers a system from abnormal inputs or hardware faults (for example, recovery from an attempt to divide by zero)

**firmware** – a program that provides low level control for devices

**interrupt priority** – the priority assigned to an interrupt are given a priority so that the microprocessor knows which one needs to be serviced first and which interrupts are to be dealt with quickly

**service (an interrupt)** – when an interrupt is received, some action needs to be taken by the processor depending on what caused the interrupt; until this is resolved (that is, it is serviced), the interrupt cannot be removed to allow the processor to continue

**interrupt service routine (ISR)** – software that handles interrupt requests (for example, when the printer out of paper) and sends a request to the CPU for processing

**machine code** – a binary programming language, a program written in machine code can be loaded and executed without translation

**high-level language (HLL)** – a programming language that is independent of computer hardware, a program written in a HLL needs to be translated into machine code before it is executed.

**low-level language (LLL)** – a programming language that is dependent on computer hardware, both machine code and assembly language are LLLs

**assembly language** – a programming language that is dependent on computer hardware, a program written in an assembly language program needs to be translated into machine code before it is executed

**assembler** – a computer program that translates programming code written in assembly language into machine code

**compiler** – a computer program that translates a source program written in a high-level language to machine code

**translator** – converts a program written in a high-level language program into machine code

**interpreter** – a computer program that analyses and executes a program written in a high-level language line by line

**Integrated Development Environment (IDE)** – a suite of programs used to write and test a computer program written in a high-level language

**debugging** – finding errors in a computer program by running or tracing the program

**prettyprinting** – displaying source code using different colours and formatting, which make the code easier to read and understand

**report window** – a separate window in the runtime environment of an IDE that shows the contents of variables during the execution of a program



## Exam-style questions

- 1 There are five types of software on the left and four items of hardware on the right.

Draw lines to connect each software item to the hardware item where the software will reside; each hardware item may be used once, more than once or not at all.

part of a program or OS currently in use	CMOS
the actual BIOS settings	flash memory
the actual BIOS software	hard disk drive (HDD)
operating system software	random access memory (RAM)
virus scanner software	

[5]

- 2 Which utility programs are being described below?
- Software that runs in the background and checks for malware; suspect programs are quarantined and deleted if necessary
  - Software that rearranges data on a hard disk drive (HDD) to reduce the scattering of the data stored on the HDD
  - Software that manages access control and user accounts and also protects network interfaces
  - Program that supplies static or moving images on a monitor when the computer has been idle for a period of time
  - Software that communicates with the operating system and translates data into a format understood by an input/output device

[5]

- 3
- Describe the purpose of an **operating system**.
  - What is meant by **virtual memory**?
  - What is meant by **disk thrashing** and why does it occur?
  - What is meant by **multitasking**?
  - Describe what an **interrupt** is.
- 4
- Explain the differences between a **Graphical User Interface (GUI)** and a **Command Line Interface (CLI)**.
  - Give **one** advantage and **one** disadvantage of using a GUI interface.
  - Give **one** advantage and **one** disadvantage of using a CLI interface.

[10]

[3]

[2]

[2]



## 4 SOFTWARE

5 In this question you will be given a statement followed by four possible answers. Select which of the four answers you think is correct.

a What is meant by the term *buffer*?

A	part of the RAM which is used to store the operating system in use
B	unused areas on a hard disk drive
C	an area in memory that temporarily holds data
D	an example of firmware

b What is the function of a printer driver?

A	area of memory that holds data waiting to be printed out
B	software that communicates with the operating system and translates data into a format which can be printed out
C	an area in memory that temporarily holds data
D	an example of firmware used to interface with the printer

c Which of these options describes a task carried out by the operating system?

A	preventing unauthorised access to a computer system
B	handling the HTTPS requests from a website
C	allocating memory to competing applications running on a computer system
D	booting up the computer motherboard when the computer is powered up

d Which of these is the correct name for the operating system function that allows many programs to run simultaneously?

A	utility program
B	application package
C	embedded system
D	multitasking

e Which of these is the name of a type of interface that allows the user to type in commands on a keyboard which gives the computer instructions?

A	command line interface (CLI)
B	graphical user interface (GUI)
C	touchscreen interface
D	drop down menu interface

f Which of these descriptions is the main purpose of a GUI?

A	allows a user to directly carry out very complex tasks on a computer
B	the apps will run faster in a GUI environment
C	a GUI interface makes it much easier to use the computer
D	it is a type of app that allows a user to create drawings and images on the screen

- g** Which of the following will produce a signal which is sent to the CPU to suspend its current operation?

A	bootstrap
B	buffer
C	interrupt
D	quarantine

- h** On which one of the following memories are the BIOS *settings* stored?

A	ROM
B	RAM
C	EEPROM
D	CMOS

- i** Which one of the following statements is TRUE?

A	any device plugged into the USB port of a computer must have a unique serial number
B	the BIOS settings are stored in ROM so that they cannot be altered at any time by the user
C	interrupts from software always have a higher priority than an app currently being run on a computer
D	user accounts allow users to share resources, such as printers

- j** Which one of the following statements is FALSE?

A	security management is part of a typical operating system
B	post-WIMP interfaces make use of the more modern optical mouse to select icons
C	logical errors include errors such as 'division by zero'
D	printers use interrupts when they need more data to continue a printing job

[10]

- 6 a** Describe the differences between a *compiler* and an *interpreter*. [4]  
**b** Give **one** advantage and **one** disadvantage of using a compiler. [2]  
**c** Give **one** advantage and **one** disadvantage of using an interpreter. [2]
- 7** Describe the differences between a *compiler* and an *assembler*. [4]
- 8** An Integrated Development Environment contains these features:  
 – **Auto-completion**  
 – **Auto-correction**  
 – **Prettyprinting**
- a** Explain what is meant by the term *IDE*. [2]  
**b** Describe each of the features given. Include a suitable example with each description. [6]  
**c** Identify **two** other features that should be included in an IDE. Give a reason why each feature is necessary. [4]
- 9** Pedro has written a program in a high-level language to do some calculations for a friend. His friend needs to use the program immediately on his laptop. Pedro does not know what software is available on the laptop. Also, his friend's internet connection is very slow. Explain which type of translator Pedro should use for his program. Give reasons why this is the best choice in this case. [6]

## 5

## The internet and its uses

**In this chapter you will learn about:**

- ★ the internet and the World Wide Web
  - the differences between the internet and the World Wide Web
  - what is meant by a uniform resource locator (URL)
  - the purpose and operation of hypertext transfer protocols (HTTP and HTTPS)
  - the purpose and function of a web browser
  - how web pages are located, retrieved and displayed
  - cookies (including session and persistent cookies)
- ★ digital currency
  - digital currencies and how they are used
  - the process of blockchaining and how it is used to track digital currency transactions
- ★ cyber security
  - cyber security threats
  - solutions to keep data safe from security threats.

The internet is probably one of the greatest inventions of the twentieth century; it has changed the way the world works and communicates for ever. It is a great source of good, but has also spawned new types of crime which can be just as devastating as physical crime. This chapter will investigate many of the features of the internet but, in particular, will concentrate on cyber security threats, how we can recognise such threats and take the necessary action to stay safe.

## 5.1 The internet and the World Wide Web (WWW)

### 5.1.1 The differences between the internet and the World Wide Web (WWW)

#### Link

See Section 3.4 for more on network hardware and devices.

The word **internet** comes from **INTER**connected **NET**work, since it is basically a worldwide collection of interconnected networks. The internet is actually a concept rather than something tangible (that is, something we can touch). It relies on a physical infrastructure that allows networks and individual devices to connect to other networks and devices.

In contrast, the **World Wide Web (WWW)** is only a part of the internet that users can access using web browser software. The World Wide Web consists of a massive collection of web pages, and is based on the hypertext transfer protocol – see Section 5.1.3. Therefore, the World Wide Web is a way of accessing information using the internet; so the internet and the World Wide Web are actually quite different. In summary:



## 5.1 The internet and the World Wide Web (WWW)

▼ **Table 5.1** Summary of differences between the internet and the World Wide Web

Internet	World Wide Web (WWW)
<ul style="list-style-type: none"> <li>• users can send and receive emails</li> </ul>	<ul style="list-style-type: none"> <li>• it is a collection of multimedia web pages and other information on websites</li> </ul>
<ul style="list-style-type: none"> <li>• allows online chatting (via text, audio and video)</li> </ul>	<ul style="list-style-type: none"> <li>• http(s) protocols are written using hypertext mark-up language (HTML)</li> </ul>
<ul style="list-style-type: none"> <li>• makes use of transmission protocols (TCP) and internet protocols (IP)</li> </ul>	<ul style="list-style-type: none"> <li>• uniform resource locators (URLs) are used to specify the location of web pages</li> </ul>
<ul style="list-style-type: none"> <li>• it is a worldwide collection of interconnected networks and devices</li> </ul>	<ul style="list-style-type: none"> <li>• web resources are accessed by web browsers</li> <li>• uses the internet to access information from web servers</li> </ul>

## 5.1.2 Uniform resource locators (URLs)

Web browsers are usually just referred to as browsers

**Web browsers** are software that allow users to access and display web pages on their device screens. Browsers interpret **hypertext mark-up language (HTML)** sent from websites and produce the results on the user's device. **Uniform resource locators (URLs)** are text addresses used to access websites. A URL is typed into a browser address bar using the following format:

`protocol://website address/path/file name`

The **protocol** is usually either **http** or **https**.

The **website address** is:


- » **domain host** (www),
- » **domain name** (website name),
- » **domain type** (.com, .org, .net, .gov, for example),
- » and sometimes **country code** (.uk, .de, .cy, for example).

The **path** is the web page, but is often omitted and it then becomes the root directory of the website (see example below).

The **file name** is the item on the web page. For example:

`https://www.hoddereducation.co.uk/ict`

## 5.1.3 HTTP and HTTPS

**Hypertext transfer protocol (http)** is a set of rules that must be obeyed when transferring files across the internet. When some form of security (for example, SSL or TLS) is used, then this changes to **https** (you will often see the green padlock  in the status bar as well). The 's' stands for secure, and indicates a more secure way of sending and receiving data across a network (for example, the internet).

## 5 THE INTERNET AND ITS USES

### 5.1.4 Web browsers

As mentioned earlier, browsers are software that allow a user to access and display web pages on their device screens. Browsers interpret (translate) the HTML from websites and show the result of the translation; for example, videos, images/text and audio. Most browsers have the following features:

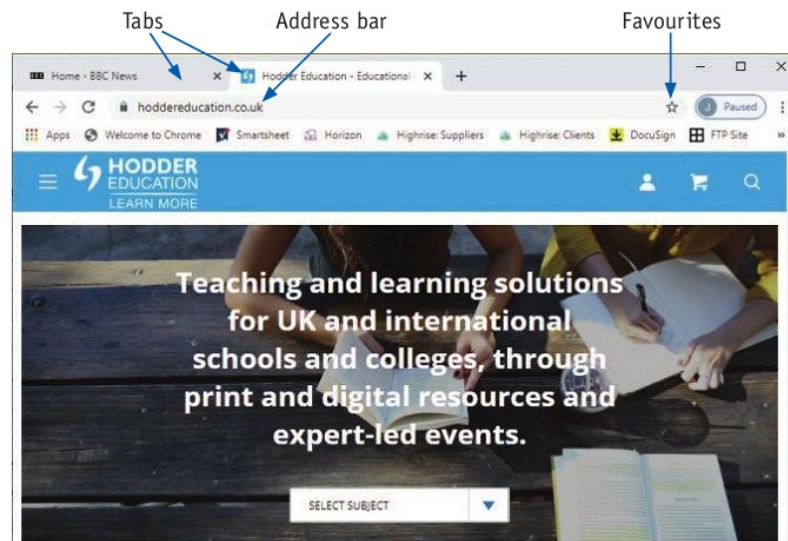
- » they have a home page
- » they can store a user's favourite websites/web pages (referred to as bookmarks)
- » they keep a history of websites visited by the user (user history)
- » they have the ability to allow the user to navigate forwards and backwards through websites/web pages already opened
- » many web pages can be open at the same time by using multiple tabs
- » they make use of cookies (see Section 5.1.6)
- » they make use of hyperlinks that allow navigation between websites and web pages; links can be opened in one of two ways:
  - either* open in a new tab by using <ctrl> + <click>
  - or* open in the same tab by simply clicking on the link

[www.hoddereducation.com](http://www.hoddereducation.com)



▲ **Figure 5.1**

- » data is stored as a cache (see Section 5.1.5)
- » make use of JavaScript
- » they use an address bar; for example:



▲ **Figure 5.2** Browser address bar

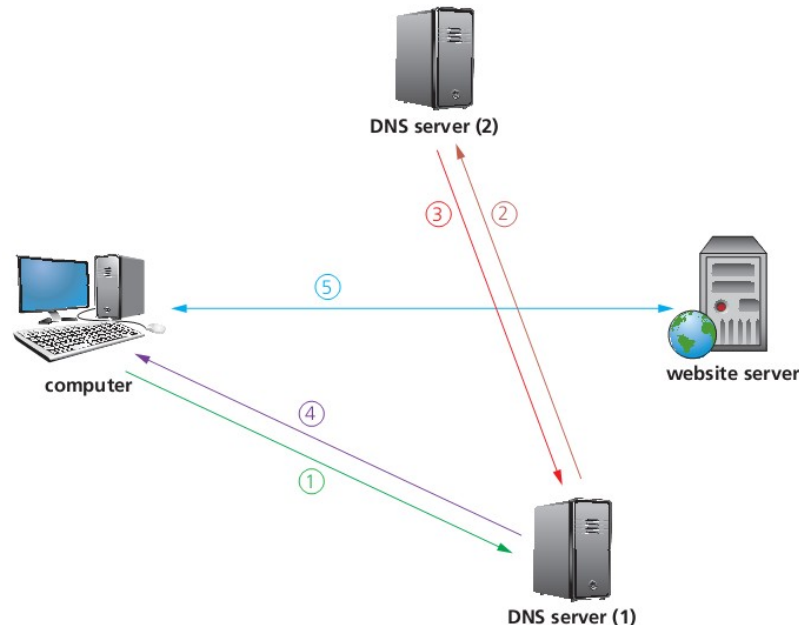
### 5.1.5 Retrieval and location of web pages

HTML (HyperText Markup Language) is a language used to display content on browsers. All websites are written in HTML and hosted on a web server that has its own IP address. To retrieve pages from a website your browser needs to know this IP address. The **Domain Name Server (DNS)** (also known as domain name system) is a system for finding IP addresses for a domain name given in a URL. URLs and domain name servers eliminate the need for a user to memorise IP addresses. The DNS process involves converting a URL (such as `www.hoddereducation.co.uk`) into an IP address the computer can understand (such as `107.162.140.19`). The DNS process involves more than one server.

#### Link

For more on IP addresses see Section 3.4.

DNS servers contain a database of URLs with the matching IP addresses. Figure 5.3 shows how a web page can be located and then sent back to the user's computer. The DNS plays a vital role in this process:



▲ **Figure 5.3** How DNS is used to locate and retrieve a web page

- (1) The user opens their browser and types in the URL (`www.hoddereducation.co.uk`) and the browser asks the DNS server (1) for the IP address of the website.
- (2) In this case, let's assume the DNS server can't find `www.hoddereducation.co.uk` in its database or its cache, so it sends out a request to a DNS server (2).
- (3) The DNS server (2) finds the URL and can map it to `107.162.140.19`; this IP address is sent back to the DNS server (1) which now puts this IP address and associated URL into its cache/database.
- (4) This IP address is then sent back to the user's computer.
- (5) The computer now sets up a communication with the website server and the required pages are downloaded. HTML files are sent from the website server to the computer. The browser interprets the HTML, which is used to structure content, and then displays the information on the user's computer.

(Note: in this case, the IP address was found on the second DNS server.)



## 5 THE INTERNET AND ITS USES

### 5.1.6 Cookies

this tracks data about users, such as IP addresses and browsing activity

**Cookies** are small files or code stored on a user's computer. They are sent by a web server to a browser on a user's computer. Each cookie is effectively a small look-up table containing pairs of (**key, data**) values, for example, (**surname, Jones**) (**music, rock**). Every time a user visits a website, it checks if it has set cookies on their browser before. If so, the browser reads the cookie which holds key information on the user's preferences such as language, currency and previous browsing activity. Cookies allow user tracking and maintain user preferences. Collected data can also be used to customise the web page for each individual user. For example, if a user buys a book online, the cookies remember the type of book chosen by the user and the web page will then show a message such as *"Customers who bought Hodder IGCSE ICT also bought Hodder IGCSE Computer Science"*.

There are two types of cookie:

- » session cookie
- » persistent (or permanent) cookie.

If a cookie doesn't have an expiry date associated with it, it is always considered to be a session cookie. So what are the basic differences?

#### Session cookies

**Session cookies** are used, for example, when making online purchases. They keep a user's items in a **virtual shopping basket**. This type of cookie is stored in temporary memory on the computer, doesn't actually collect any information from the user's computer and doesn't personally identify a user. Hence, session cookies cease to exist on a user's computer once the browser is closed or the website session is terminated.

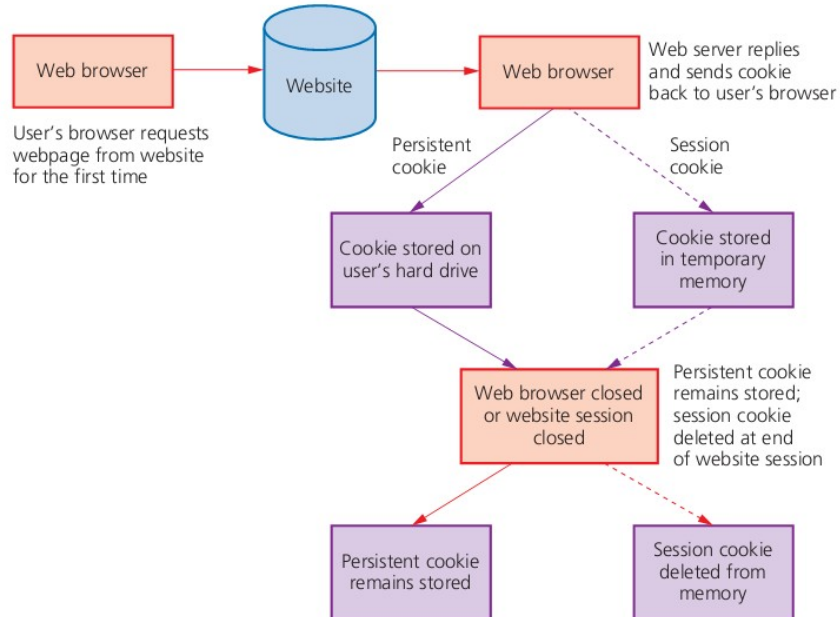
#### Persistent (permanent) cookies

**Persistent cookies** remember a user's log in details (so that they can authenticate the user's browser). They are stored on the hard drive of a user's computer until the expiry date is reached or the user deletes it. These cookies remain in operation on the user's computer even after the browser is closed or the website session is terminated. Their advantage is that they remove the need to type in login details every time a certain website is visited. Some websites use cookies to store more personal information or user preferences. However, this can only be done if the user has provided the website with certain personal information and agrees to it being stored. Legitimate websites will always encrypt any personal information stored in the cookie to prevent unauthorised use by a third party that has access to your cookie folder. Many countries have introduced laws to protect users and these cookies are supposed to become deactivated after six months (even if the expiry date has not yet been reached).

Persistent cookies are a very efficient way of carrying data from one website session to another, or even between sessions on related websites; they remove the need to store massive amounts of data on the web server itself. Storing the data on the web server without using cookies would also make it very difficult to retrieve a user's data without requiring the user to log in every time they visit the website.

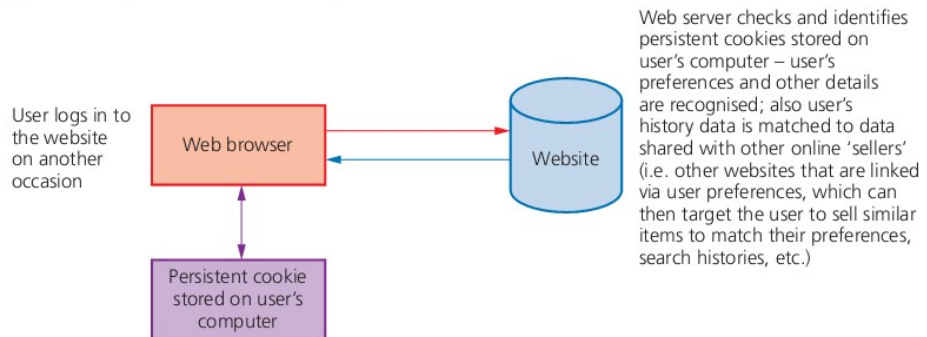
Figures 5.4 and 5.5 summarise what happens when a website is first visited and then what happens in subsequent visits:

**1** First time the user logs in to website:



▲ **Figure 5.4** Cookies (first login)

**2** User logs in to website again



▲ **Figure 5.5** Cookies (subsequent logins)

Summary of the uses of (persistent) cookies:

- » allow the website to remember users' passwords, email addresses and invoice details, so they won't have to insert all of this information every time they visit or every time they purchase something from that website
- » serve as a memory, enabling the website to recognise users every time they visit it
- » save users' items in a virtual shopping basket/cart
- » track internet habits and users' website histories or favourites/bookmarks
- » target users with advertising that matches their previous buying or surfing habits
- » store users' preferences (for example, recognise customised web pages)

## 5 THE INTERNET AND ITS USES

- » are used in online financial transactions
- » allow progress in online games and quizzes to be stored
- » allow social networking sites to recognise certain preferences and browsing histories
- » allow different languages to be used on the web pages automatically as soon as users log on.

### Activity 5.1

- 1 A URL being entered is: **http://www.urlexample.co.ie/sample\_page**

Identify:

- a the domain name
  - b the domain type
  - c the file name
  - d which protocol is being used.
- 2 a Give two differences between session cookies and persistent cookies.  
b Describe three uses of cookies.
- 3 The following table shows five features of the internet and the World Wide Web. Tick (✓) the appropriate box to indicate which feature refers to the internet and which feature refers to the World Wide Web:

Feature	Internet	World Wide Web
it is possible to send and receive emails		
makes use of http protocols		
uses URLs to specify the locations of websites and web pages		
resources can be accessed by using web browsers		
makes use of TCP and IP		

- 4 Why do you think persistent cookies are sometimes referred to as **tracking cookies**? Give at least two pieces of evidence to support your answer.

## 5.2 Digital currency

### 5.2.1 What is digital currency?

**Digital currency** exists purely in a digital format. It has no physical form unlike conventional **fiat currency** (for example, \$, £, €, and ¥).

(Note: Fiat is a Latin word meaning 'let it be done'; since conventional currency is backed by governments and banks rather than being linked to gold or silver reserves, it is referred to as fiat currency.)

Digital currency is an accepted form of payment to pay for goods or services. As with cash or credit/debit cards, digital currency can be transferred between various accounts when carrying out transactions. It has made it possible to bank online (for example, using *PayPal*) or via a smartphone app (for example,



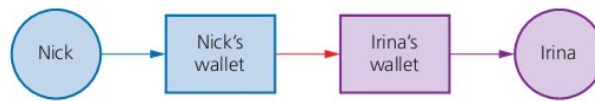
*Apple Pay*). This is all possible because money only exists as data on a computer system, but it can be transferred into physical cash if we need it.

Digital currency relies on a **central banking system**. For example, suppose Nick wishes to send Irina some money; Nick uses bank 'X' and Irina uses bank 'Y':



▲ **Figure 5.6** Digital currency

The problem with centralisation is maintaining confidentiality and security; these have always been issues with digital currency systems. However, one example of digital currency, known as cryptocurrency, has essentially overcome these issues by introducing decentralisation:



▲ **Figure 5.7** Cryptocurrency and decentralisation

- » Cryptocurrency uses **cryptography** to track transactions; it was created to address the problems associated with the centralisation of digital currency.
- » Traditional digital currencies are regulated by central banks and governments (in much the same way as fiat currencies). This means all transactions and exchange rates are determined by these two bodies. Cryptocurrency has no state control and all the rules are set by the cryptocurrency community itself.
- » Unlike existing digital currencies, cryptocurrency transactions are publicly available and therefore all transactions can be tracked and the amount of money in the system is monitored.
- » The cryptocurrency system works by being within a **blockchain** network which means it is much more secure.

## 5.2.2 Blockchaining

Blockchain is a decentralised database. All the transactions of networked members are stored on this database. Essentially, the blockchain consists of a number of interconnected computers but they are **not connected** to a central server. All transaction data is stored on **all** computers in the blockchain network.

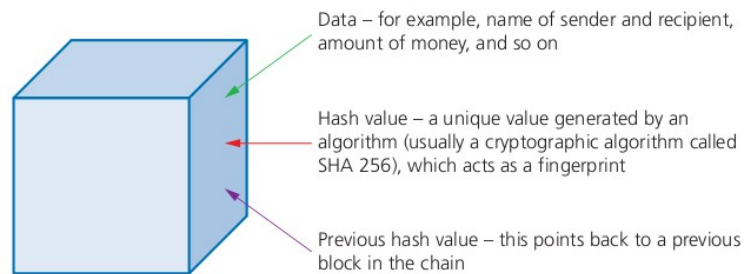
Whenever a new transaction takes place, all the networked computers get a copy of the transaction; therefore **it cannot be changed without the consent of all** the network members. This effectively removes the risk of security issues such as hacking. Blockchain is used in many areas, such as:

- » cryptocurrency (digital currency) exchanges
- » smart contracts
- » research (particularly within pharmaceutical companies)
- » politics
- » education.

## 5 THE INTERNET AND ITS USES

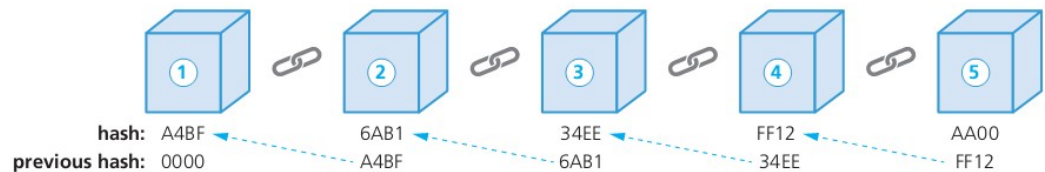
### How blockchain works

Whenever a new transaction takes place, a new **block** is created:



▲ **Figure 5.8** Block description

A new hash value is created each time a new block is created. This hash value is unique to each block and includes a **timestamp**, which identifies when an event actually takes place. We will now consider what happens when a chain of blocks is created. Figure 5.9 shows part of a typical blockchain:



▲ **Figure 5.9** Part of blockchain (showing 5 blocks)

It is clear from Figure 5.9 how these blocks are connected. Block '1' is known as the **genesis block** since it doesn't point to any previous block. Now suppose block '2' is changed in some way. Any changes to the data within block '2' will cause the value of the hash to change (it will no longer have the value 6AB1). This means that block '3' and beyond will now be invalid since the chain was broken between block '2' and '3' (previous hash 6AB1 in block '3' is no longer valid).

This will prevent tampering (for example, by a hacker). However, it may be crossing your mind that computers are now so fast that it should be possible to quickly create a whole new string of blocks, and therefore recreate a new chain before the problem has been discovered. This is prevented by **proof-of-work**, which makes sure it takes ten minutes to determine the necessary proof-of-work for *each* block **before** it can be added to the chain. This is 'policed' by **miners**, which are special network users that get a commission for each new block created. Thus, the whole process of creating new blocks is slowed down which foils hackers and also means that the currency is regulated by all the network computers.

Consequently, this makes it almost impossible to hack into the blockchain since it would be necessary to attack every single block in the chain at the same time. It only takes one block to break the link for any transaction to be terminated. When a new block is created, it is sent to each computer in the blockchain and is checked for correctness before being added to the blockchain. If a new network user is created, they get a copy of everything in the whole blockchain system.

### Activity 5.2

- 1
  - a A blockchain has seven blocks. Draw a diagram to show how they are all connected to form a blockchain network.
  - b Describe what would happen if block 4 was hacked to change the sum of money in the transaction.
- 2 What are the main differences between digital currency and cryptocurrency?

## 5.3 Cyber security

### 5.3.1 Cyber security threats

Keeping data safe is extremely important for many reasons. It may be personal data that you want to keep within your family or close friends, or it may be commercial data, such as passwords and bank details.

Data can be corrupted or deleted either through accidental damage or malicious acts. There are also many ways data can be intercepted leading to cyber security threats. The following list shows the cyber threats which will be considered in this section:

- » brute force attacks
- » data interception
- » distributed denial of service (DDoS) attacks
- » hacking
- » malware (viruses, worms, Trojan horse, spyware, adware and ransomware)
- » phishing
- » pharming
- » social engineering.

#### Brute force attacks

If a hacker wants to 'crack' your password, they can systematically try all the different combinations of letters, numbers and other symbols until eventually they find your password. This is known as a **brute force attack** and there isn't a lot of sophistication in the technique.

One way to reduce the number of attempts needed to crack a password is to first go through a series of logical steps:

- 1 Check if the password is one of the **most** common ones used (the five most common are: 123456, password, qwerty, 111111 and abc123); since these simple passwords are seen so many times it's a good place for the hacker to start.
- 2 If it isn't in the common password list, the next thing to do is to start with a strong **word list** (this is a text file containing a collection of words that can be used in a brute force attack); some programs will generate a word list containing a million words. Nonetheless this is still a faster way of cracking a password than just total trial and error.

Clearly method (2) would still take several hours until the password was found. The longer a password is and the greater the variation of characters used, the harder it will be to crack (also refer to the notes on the use of passwords in authentication).



## 5 THE INTERNET AND ITS USES

---

### Link

For more on data packets refer to Chapter 2.

### Data interception

**Data interception** is a form of stealing data by **tapping** into a wired or wireless communication link. The intent is to compromise privacy or to obtain confidential information.

Interception can be carried out using a **packet sniffer**, which examines data packets being sent over a network. The intercepted data is sent back to the hacker. This is a common method when wired networks are used.

Wi-Fi (wireless) data interception can be carried out using **wardriving** (or sometimes called **Access Point Mapping**). Using this method, data can be intercepted using a laptop or smartphone, antenna and a GPS device (together with some software) outside a building or somebody's house. The intercepted Wi-Fi signal can then reveal personal data to the hacker, often without the user being aware this is happening.

Obviously, encryption of data makes life more difficult for the hacker. While it doesn't stop the data being intercepted or altered in some way, encryption will make the data incomprehensible to the hacker if they don't have access to a decryption key. Therefore, to safeguard against wardriving, the use of a **wired equivalency privacy (WEP)** encryption protocol, together with a firewall, is recommended. It is also a good idea to protect the use of the wireless router by having complex passwords. It is important not to use Wi-Fi (wireless) connectivity in public places (such as an airport) since no data encryption will exist and your data is then open to interception by anyone within the airport.

### Distributed Denial of Service (DDoS) attacks

A **denial of service (DoS)** attack is an attempt at preventing users from accessing part of a network, notably an internet server. This is usually temporary but may be a very damaging act or a large breach of security. It doesn't just affect networks; an individual can also be a target for such an attack. The attacker may be able to prevent a user from:

- » accessing their emails
- » accessing websites/web pages
- » accessing online services (such as banking).

One method of attack is to flood the network with useless **spam** traffic. How does this cause a problem?

When a user enters a website's URL in their browser, a request is sent to the web server that contains the website or web page. Obviously, the server can only handle a finite number of requests. So if it becomes overloaded by an attacker sending out thousands of requests, it won't be able to service a user's legitimate request. This is effectively a denial of service. In a **distributed denial of service (DDoS)** the spam traffic originates from many different computers, which makes it hard to block the attack.

This can happen to a user's email account, for example, by an attacker sending out many spam messages to their email account. Internet service providers (ISPs) only allow a specific data quota for each user. Consequently, if the attacker sends out thousands of emails to the user's account, it will quickly become clogged up

and the user won't be able to receive legitimate emails. An individual user or a website can guard against these attacks to some degree by:

- » using an up-to-date malware checker
- » setting up a firewall to restrict traffic to and from the web server or user's computer
- » applying email filters to filter out unwanted traffic (for example, spam).

There are certain signs a user can look out for to see if they have become a victim of a DDoS attack:

- » slow network performance (opening files or accessing certain websites)
- » inability to access certain websites
- » large amounts of spam email reaching the user's email account.

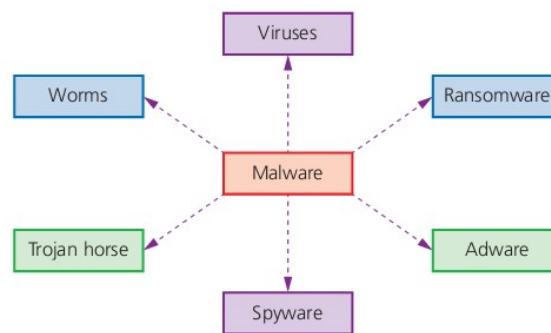
### Hacking

**Hacking** is generally the act of gaining illegal access to a computer system without the user's permission. This can lead to identity theft or the gaining of personal information; data can be deleted, passed on, changed or corrupted. As mentioned earlier, encryption does not stop hacking; it makes the data meaningless to the hacker but it doesn't stop them from deleting, corrupting or passing on the data. Hacking can be prevented through the use of firewalls, user names and frequently changed strong passwords. Anti-hacking software and intrusion-detection software also exists in the fight against hacking.

Malicious hacking, as described above, takes place without the user's permission, and is always an illegal act. However, universities and companies now run courses in **ethical hacking**. This occurs when companies authorise paid hackers to check out their security measures and test how robust their computer systems are to hacking attacks.

### Malware

**Malware** is one of the biggest risks to the integrity and security of data on a computer system. There are many forms of malware; this chapter will only consider the following in any detail:



▲ **Figure 5.10** Malware types

#### Viruses

**Viruses** are programs or program code that replicate (copies themselves) with the intention of deleting or corrupting files, or causing a computer to malfunction (for example, by deleting .exe files, filling up the hard drive with 'useless' data, and so on).

## 5 THE INTERNET AND ITS USES

---

Viruses need an **active host** program on the target computer or an operating system that has already been infected, before they can actually run and cause harm (that is, they need to be executed by some trigger before starting to cause any damage).

Viruses are often sent as email attachments, reside on infected websites or on infected software downloaded to the user's computer. Apart from all the usual safety actions (for example, don't open emails from unknown sources, don't install non-original software), always run an up-to-date virus scanner (refer to Chapter 4 for more details).



### Find out more

Reading this chapter and other chapters throughout this book, find out the various ways viruses can be sent. Produce a wall chart showing all of these ways and the various ways to avoid receiving viruses.

### Worms

**Worms** are a type of stand-alone malware that can self-replicate. Their intention is to spread to other computers and corrupt whole networks; unlike viruses, they don't need an active host program to be opened in order to do any damage. They remain inside applications which allows them to move throughout networks. In fact, worms replicate without targeting and infecting specific files on a computer; they rely on security failures within networks to permit them to spread unhindered.

Worms frequently arrive as message attachments and only one user opening a worm-infested email could end up infecting the whole network. As with viruses, the same safeguards should be employed, together with the running of an up-to-date anti-virus program. Worms tend to be problematic because of their ability to spread throughout a network without any action from an end-user; whereas viruses require each end-user to somehow initiate the virus.

Examples include the 'I love you' worm, which attacked nearly every email user in the world, overloaded phone systems and even brought down television networks. All of this makes them more dangerous than viruses.

### Trojan horse

A **Trojan horse** is a program which is often disguised as legitimate software but with malicious instructions embedded within it. A Trojan horse replaces all or part of the legitimate software with the intent of carrying out some harm to the user's computer system.

They need to be executed by the end-user and therefore usually arrive as an email attachment or are downloaded from an infected website. For example, they could be transmitted via a fake anti-virus program that pops up on the user's screen claiming their computer is infected and action needs to be taken. The user will be invited to run fake anti-virus as part of a free trial. Once the user does this, the damage is done.

Once installed on the user's computer, the Trojan horse will give cyber criminals access to personal information on your computers, such as IP addresses, passwords and other personal data. Spyware (including key logging software) and ransomware are often installed on a user's computer via Trojan horse malware.



Because they rely on tricking end-users, firewalls and other security systems are often useless since the user can overrule them and initiate the running of the malware.

### Spyware

**Spyware** is software that gathers information by monitoring a user's activities carried out on their computer. The gathered information is sent back to the cybercriminal who originally sent the spyware. They are primarily designed to monitor and capture web browsing and other activities and capture personal data (for example, bank account numbers, passwords and credit/debit card details). Spyware can be detected and removed by anti-spyware software. The big danger of spyware is the method it used to enter a user's system and exploit it; for example, did it come from social engineering? If spyware is found on a computer, it should set off alarm bells since a weakness in the security has been found which could be exploited by other, often more dangerous, malware.



#### Find out more

Key logging software is often part of spyware.

- 1 How does this type of malware gather data from the user's computer?
- 2 Some banks use drop-down menus to overcome key logging software. Explain how using drop-down boxes to enter characters from, for example, a password can help in security.

### Adware

**Adware** is a type of malware. At its least dangerous it will attempt to flood an end-user with unwanted advertising. For example, it could redirect a user's browser to a website that contains promotional advertising, it could appear in the form of pop-ups, or it could appear in the browser's toolbar and redirect search requests.

Although not necessarily harmful, adware can:

- » highlight weaknesses in a user's security defences
- » be hard to remove – it defeats most anti-malware software since it can be difficult to determine whether or not it is harmful
- » hijack a browser and create its own default search requests.

### Ransomware

Essentially, ransomware are programs that encrypt data on a user's computer and 'hold the data hostage'. The cybercriminal waits until the ransom money is paid and, sometimes, the decryption key is then sent to the user. It has caused considerable damage to some companies and individuals.

Imagine a situation where you log on to your computer, only to find the screen is locked and you can't unlock it until the demands of the cybercriminal have been met. This malware restricts access to the computer and encrypts all the data until a ransom is paid. It can be installed on a user's computer by way of a Trojan horse or through social engineering.

When ransomware is executed, it either encrypts files straightaway or it waits for a while to determine how much of a ransom the victim can afford. The malware can be prevented by the usual methods (for example, by avoiding phishing

## 5 THE INTERNET AND ITS USES

emails) but once it is executed, it is almost impossible to reverse the damage caused. The best way to avoid a catastrophe is to ensure regular back-ups of key files are kept and thus avoid having to pay a ransom.

### Summary of malware

Table 5.2 summarises the six types of malware described in Section 5.3.1.

▼ **Table 5.2** Summary of types of malware


Viruses – programs (or program code) that can replicate/copy themselves with the intention of deleting or corrupting files, or causing the computer to malfunction. They need an active host program on the target computer or an operating system that has already been infected before they can run
Worms – these are types of standalone viruses that can replicate themselves with the intention of spreading to other computers; they often networks to search out computers with weak security that are prone to such attacks
Trojan horses – these are malicious programs often disguised as legitimate software; they replace all or part of the legitimate software with the intent of carrying out some harm to the user's computer system
Spyware – software that gathers information by monitoring, for example, all the activity on a user's computer; the gathered information is then sent back to the person who sent the software (sometimes spyware monitors key presses and is then referred to as key logging software)
Adware – software that floods a user's computer with unwanted advertising; usually in the form of pop-ups but can frequently appear in the browser address window redirecting the browser to a fake website which contains the promotional adverts
Ransomware – programs that encrypt the data on a user's computer; a decryption key is sent back to the user once they pay a sum of money (a ransom); they are often sent via a Trojan horse or by social engineering

### Phishing

**Phishing** occurs when a cybercriminal sends out legitimate-looking emails to users. The emails may contain links or attachments that, when initiated, take the user to a fake website; or they may trick the user into responding with personal data (for example, bank account details or credit/debit card details).

The email usually appears to be genuine coming from a known bank or service provider (also refer to Section 5.3.2). The key point is that the recipient has to initiate some act before the phishing scam can cause any harm. If suspicious emails are deleted or not opened, then phishing attacks won't cause any problems.

There are numerous ways to help prevent phishing attacks:

- » users need to be aware of new phishing scams; those people in industry or commerce should undergo frequent security awareness training to become aware of how to identify phishing (and pharming) scams
- » it is important not to click on any emails links unless totally certain that it is safe to do so; fake emails can often be identified by 'Dear Customer .....
- » it is important to run anti-phishing toolbars on browsers (this includes tablets and mobile phones) since these will alert the user to malicious websites contained in an email
- » always look out for https or the green padlock symbol  in the address bar



- » regular checks of online accounts are also advisable as well as maintaining passwords on a regular basis
- » ensure an up-to-date browser is running on the computer device (which contains all of the latest security upgrades) and run a good firewall in the background at all times; a combination of a desktop firewall (usually software) and a network firewall (usually hardware) considerably reduces the risk of hacking, pharming and phishing on network computers
- » be very wary of pop-ups and use the browser to block them; if pop-ups get through your defences, don't click on 'cancel' since this can ultimately lead to phishing or pharming sites – the best option is to select the small **X** in the top right-hand corner of the pop-up window which closes it down.

Note: another term connected to phishing is **spear phishing**; this is where the cybercriminal targets **specific** individuals or companies to gain access to sensitive financial information or industrial espionage – regular phishing is not specific regarding who the victims are.

### Pharming

**Pharming** is malicious code installed on a user's computer or on an infected website. The code redirects the user's browser to a fake website **without** the user's knowledge. Unlike phishing, the user doesn't actually need to take any action for it to be initiated. The creator of the malicious code can gain personal data, such as bank details, from the user. Often the website appears to come from a trusted source and can lead to fraud and identity theft.


#### Why does pharming pose a threat to data security?

As mentioned above, pharming redirects internet users to a fake or malicious website set up by, for example, a hacker; redirection from a legitimate website to the fake website can be done using **DNS cache poisoning**.

Every time a user types in a URL, their browser contacts the DNS server; the IP address of the website will then be sent back to their browser. However, DNS cache poisoning changes the real IP address values to those of the fake website; consequently, the user's computer will connect to the fake website.

When a user enters a web address (URL) into a browser, the computer is sent the IP address of the website; if the IP address has been modified somehow the user's computer will be redirected to the fake website.

It is possible to mitigate against the risk of pharming:

- » Use of anti-virus software can detect unauthorised alterations to a website address and warn the user of the potential risks.
- » However, if the DNS server itself has been infected (rather than the user's computer) it is much more difficult to mitigate the risk.
- » Many modern browsers can alert users to pharming and phishing attacks.
- » It is very important to check the spelling of websites to ensure the web address used is correct.
- » As with phishing, use of **https** or the green padlock symbol  in the address bar is an additional form of defence.

### Link

See Section 5.1 for more on URLs, IP addresses and DNS (Domain Name Server).



## 5 THE INTERNET AND ITS USES

### Activity 5.3

- 1 A company has offices in four different countries. Communication and data sharing between the offices is done via computers connecting over the internet.
  - a Describe **three** data security issues the company might encounter during their day-to-day communications and data sharing.
  - b For each issue described, explain why it could be a threat to the security of the company.
  - c For each issue described, describe a way to mitigate the threat that has been posed.
- 2 Explain the following three terms:
  - worm
  - ransomware
  - Trojan horse.
- 3 John works for a car company. He maintains the database that contains all the personal data of the people working for the car company. John was born on 28th February 1990 and has two pet cats called Felix and Max.
  - a John needs to use a password and a user name to log onto the database. Why would the following passwords not be a very good choice:
    - i 280290
    - ii FiLix1234
    - iii John04
  - b Describe how John could improve his passwords and also how he should maintain his passwords to maximise database security.
  - c When John enters a password on his computer he is presented with the following question on his screen:

Would you like to save the password on this device?

Why is it important that John always says **No** to this question?



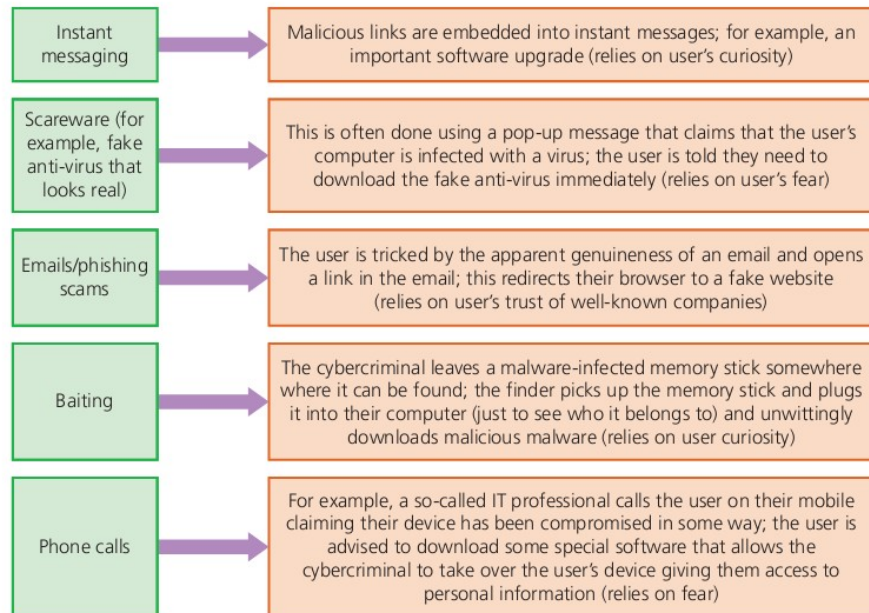
### Find out more

Apart from malware, data can be **accidentally** lost. Find out ways that data could be recovered and ways to minimise the risk for each of the following situations:

- 1 Accidental data loss, such as accidental deletion of a file
- 2 Hardware fault, such as a head crash on a hard disk drive
- 3 Software fault, due to installation of software incompatible with existing software
- 4 Incorrect operation of the computer, such as using incorrect procedure for the removal of a memory stick from a computer.

### Social engineering

**Social engineering** occurs when a cybercriminal creates a social situation that can lead to a potential victim dropping their guard. It involves the manipulation of people into breaking their normal security procedures and not following best practice. There are five types of threat that commonly exist:



▲ **Figure 5.11** Social engineering

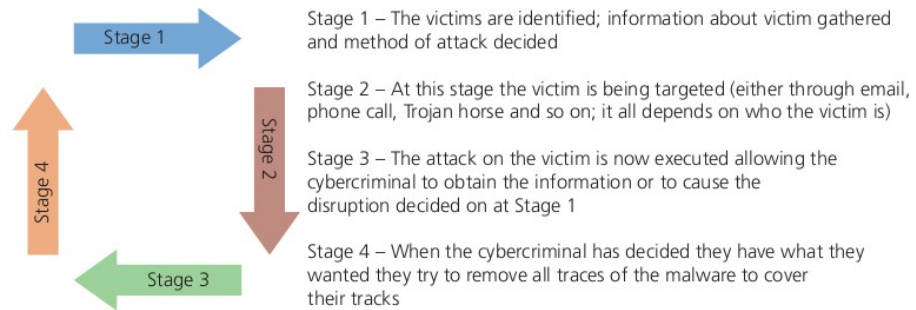
It is clear from the five examples that social engineering links into many other types of malware, and is an effective method of introducing malware. The whole idea is based on the exploitation of certain human emotions; the three most common ones to exploit are:

- » fear – the user is panicked into believing their computer is in immediate danger and isn't given time to logically decide if the danger is genuine or not; fear is a very powerful emotion that can easily be exploited by a cybercriminal
- » curiosity – the user can be tricked into believing they have won a car or they find an infected memory stick lying around; their curiosity gets the better of them and they give their details willingly to win the car (for example, credit card details to pay for delivery or road tax) or they are curious who the memory stick belongs to; without thinking clearly, their curiosity gets the better of them and the damage is done
- » empathy and trust – a real belief that all genuine-sounding companies can be trusted, therefore emails or phone calls coming from such companies must be safe; a dangerous assumption that the cybercriminal can exploit fully.

There is no hacking involved, since the user is willingly allowing the cybercriminal to have access to their computer, to download malicious software or visit fake websites; the user is rushed into making rash decisions.

## 5 THE INTERNET AND ITS USES

Figure 5.12 shows the course of action taken by a cybercriminal in targeting their victim:

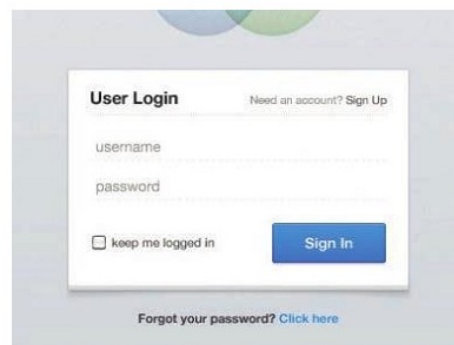


▲ **Figure 5.12** Stages in a typical social engineering scam

### 5.3.2 Keeping data safe from security threats

#### Access levels

In many computer systems, user accounts control a user's rights. This often involves having different **levels of access** for different people. For example, in a hospital it would not be appropriate for a cleaner to have access to medical data about a patient. However, a consultant would need access to this vital data. Therefore, most systems have a hierarchy of access levels depending on a person's level of security; this is usually achieved using a user name and password as shown in Figure 5.13.



▲ **Figure 5.13** Access level log in screen

#### Link

See Chapter 9 for more details about databases.

When using databases, levels of access are particularly important; it is essential to determine who has the right to read, write and delete data, for example. By having different views of data tables, it is possible for different users to only have access to certain data.



Another area where access levels are very important is in social networks (such as Facebook); with this type of application, there are usually four access levels:

- 1 public access (this refers to the data anyone from the general public can access)
- 2 friends (only people identified as 'friends' by the owner of the data can see certain data)
- 3 custom (this allows the user to further refine what data can be seen by 'friends' allowing them to exclude certain content from selected people)
- 4 data owner (this is data only the owner of the data can see).

In this type of application, users are allowed to use **privacy settings** rather than passwords to decide the level of access (for more on this see later in this section).

### Anti-malware

The two most common types of anti-malware are anti-virus and anti-spyware.

#### Anti-virus

Anti-virus has already been described in great detail in Chapter 4.

#### Anti-spyware

**Anti-spyware** software detects and removes spyware programs installed illegally on a user's computer system. The software is based on one of the following methods:

- » rules – in this case, the software looks for typical features which are usually associated with spyware thus identifying any potential security issues
- » file structures – in this case, there are certain file structures associated with potential spyware which allows them to be identified by the software.

Anti-spyware is now often part of a generic malware bundle that contains an anti-virus, anti-spyware and a personal firewall.

The general features of anti-spyware are:

- » detect and remove spyware already installed on a device
- » prevent a user from downloading spyware
- » encrypt files to make the data more secure in case it is 'spied' on
- » encryption of keyboard strokes to help remove the risk posed by the keylogging aspects of some spyware
- » blocks access to a user's webcam and microphone (the software stops the spyware taking over the control of a user's webcam and microphone which can be used to collect information without the user's knowledge)
- » scans for signs that the user's personal information has been stolen and warns the user if this has happened.

### Authentication

**Authentication** refers to the ability of a user to prove who they are. There are three common factors used in authentication:

- » something you know (for example, a password or PIN code)
- » something you have (for example, a mobile phone or tablet)
- » something which is unique to you (for example, biometrics).

#### Link

For more on anti-virus software see Section 4.1.

## 5 THE INTERNET AND ITS USES

---

There are a number of ways authentication can be done.

### Passwords and user names

Passwords are used to restrict access to data or systems. They should be hard to crack and changed frequently to retain any real level of security. Passwords can also take the form of biometrics (for example, on a mobile phone – see later). In addition to protecting access levels to computer systems, passwords are frequently used when accessing the internet. For example:

- » when accessing email accounts
- » when carrying out online banking or shopping
- » accessing social networking sites.

It is important that passwords are protected; some ways of doing this are described below:

- » run anti-spyware software to make sure that your passwords aren't being relayed back to whoever put the spyware on your computer
- » change passwords on a regular basis in case they have come into the possession of another user, illegally or accidentally
- » passwords should not be easy to crack (for example, your favourite colour, name of a pet or favourite music artist); passwords are grouped as either strong (hard to crack or guess) or weak (relatively easy to crack or guess)
- » strong passwords should contain:
  - at least one capital letter
  - at least one numerical value
  - at least one other keyboard character (such as @, \*, &, etc.)
  - an example of a strong password would be: Sy12@#TT90kj=0
  - an example of a weak password would be: GREEN

When the password is typed in, it often shows on the screen as \*\*\*\*\* so nobody else can see what the user has typed in. If the user's password doesn't match up with the user name then access will be denied. Many systems ask for a new password to be typed in twice as a verification check (to check for input errors). To help protect the system, users are only allowed to type in their password a finite number of times – usually three times is the maximum number of tries allowed before the system locks the user out. After that, the user will be unable to log on until they have reset their password.

When using an online company, if a user forgets their password or they need to reset it, they will be sent an email which contains a link to a web page where they can reset their password. This is done as an added precaution in case an unauthorised person has tried to change the user's password.

As mentioned above, it is usually necessary to use a user name as well as a password. This gives an additional security level since the user name and password must match up to allow a user to gain access to, for example, a bank website.

### Activity 5.4

- 1 Which of the following are weak passwords and which are strong passwords? Explain your decision in each case.
  - a 25-May-2000
  - b Pas5word
  - c ChapTer@06
  - d AbC\*N55!
  - e 12345X
- 2 An airport uses a computer system to control security, flight bookings, passenger lists, administration and customer services.
  - a Describe how it is possible to ensure the safety of the data on the system so that senior staff can see all the data, while customers can only access flight times (arrivals and departures) and duty-free offers.
  - b Describe how the airport can guard against malware attacks from outside and also from customers using the airport services.

### Biometrics

**Biometrics** can be used in much the same way as passwords as a way of identifying a user. Biometrics relies on certain unique characteristics of human beings; examples include:

- » fingerprint scans
- » retina scans
- » face recognition
- » voice recognition.

#### Link

For more on face recognition scans see Section 3.2.1.

Biometrics is used in a number of applications as a security device. For example, some of the latest mobile phones use fingerprint matching before they can be operated; some pharmaceutical companies use face recognition or retina scans to allow entry to secure areas.

We will now consider fingerprint scanning and retina scans in a little more detail.

#### *Fingerprint scans*

Images of fingerprints are compared against previously scanned fingerprint images stored in a database; if they match, then a user has been correctly recognised. The system compares patterns of 'ridges' and 'valleys' that are unique. The accuracy of the scan is about around 1 in 5000. Fingerprint scanning techniques have the following benefits as a form of security:

- » fingerprints are unique, therefore this technique can improve security since it would be difficult to replicate a person's fingerprints
- » other security devices (such as magnetic cards to gain entry to a building) can be lost or even stolen which makes them less effective
- » it would be impossible to 'sign in' for somebody else since the fingerprints would match with only one person on the database
- » fingerprints can't be misplaced; a person always has them!



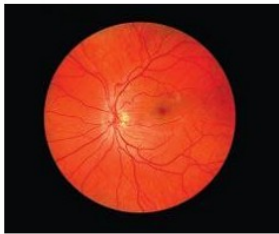
▲ **Figure 5.14** Fingerprint scan



## 5 THE INTERNET AND ITS USES

What are the drawbacks of fingerprint scanning?

- » it is relatively expensive to install and set up
- » if a person's fingers are damaged through an injury, this can have an effect on the scanning accuracy
- » some people may regard any biometric device as an infringement of civil liberties.



▲ **Figure 5.15** Retina scan

### Retina scans

Retina scans use infrared light to scan the unique pattern of blood vessels in the retina (at the back of the eye); it is a rather unpleasant technique requiring a person to sit totally still for 10 to 15 seconds while the scan takes place; it is very secure since nobody has yet found a way to duplicate the blood vessels patterns. The accuracy is about 1 in 10 million.

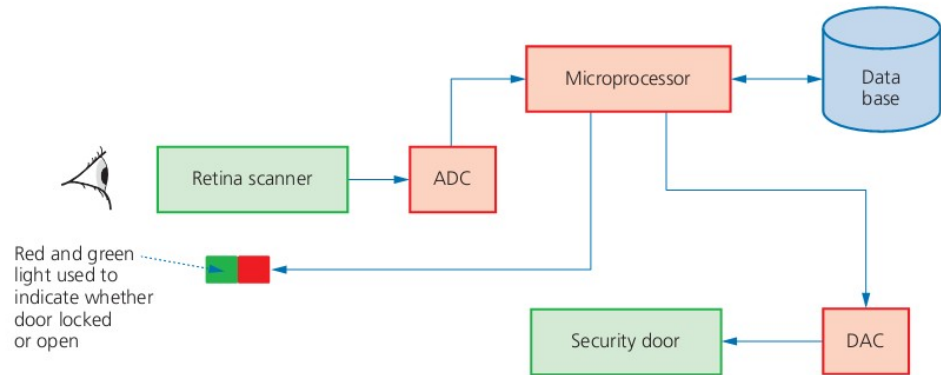
Table 5.3 shows a comparison of the benefits and drawbacks of the four common biometric techniques:

▼ **Table 5.3** Comparison of biometric devices

Biometric technique	Benefits	Drawbacks
fingerprint scans	<ul style="list-style-type: none"> <li>it is one of the most developed biometric techniques</li> <li>very easy to use</li> <li>relatively small storage requirements for the biometric data created</li> </ul>	<ul style="list-style-type: none"> <li>for some people it is very intrusive, since it is still related to criminal identification</li> <li>it can make mistakes if the skin is dirty or damaged (e.g. cuts)</li> </ul>
retina scans	<ul style="list-style-type: none"> <li>very high accuracy</li> <li>there is no known way to replicate a person's retina</li> </ul>	<ul style="list-style-type: none"> <li>it is very intrusive</li> <li>it can be relatively slow to verify retina scan with stored scans</li> <li>very expensive to install and set up</li> </ul>
face recognition	<ul style="list-style-type: none"> <li>non-intrusive method</li> <li>relatively inexpensive technology</li> </ul>	<ul style="list-style-type: none"> <li>it can be affected by changes in lighting, the person's hair, change in age, and if the person is wearing glasses</li> </ul>
voice recognition	<ul style="list-style-type: none"> <li>non-intrusive method</li> <li>verification takes less than 5 seconds</li> <li>relatively inexpensive technology</li> </ul>	<ul style="list-style-type: none"> <li>a person's voice can be easily recorded and used for unauthorised access</li> <li>low accuracy</li> <li>an illness such as a cold can change a person's voice, making absolute identification difficult or impossible</li> </ul>

**Biometric applications****? A door security system protected by retina scanner**

In this example, a company uses retina scans to permit entry to their secure research laboratories.



▲ **Figure 5.16** Security system controlled by retina scanners

A person stands facing the retina scanner. The scanned data is sent via an ADC (analogue-digital converter) to a microprocessor. The microprocessor compares the data received with retina scan data already stored in a database. If the two sets of data match, a signal is sent to turn a light from red to green and also unlock the security door. The door is controlled by a DAC (digital-analogue converter) and an actuator. If the retina scan data and database data don't match, then entry is denied and the light remains red.

**Link**

For more on actuators see Chapter 3.

**➔ Find out more**

One of the most common security systems used on mobile phones is the **capacitance fingerprint reader**. Describe how this system works.

**Activity 5.5**

- 1 In the biometric application example, retina scans were used to control entry to a secure research building.
  - a Describe how the system might change if face recognition was used instead of retina scanners. The system is triggered automatically if a motion sensor detects the presence of a person.
  - b Name other biometric devices which could be used to control entry to this building.
- 2 Many cars now use voice control as a form of security before a car can be started and it is also used to give some key commands, such as start navigation system. Describe the benefits and drawbacks of such systems in cars.

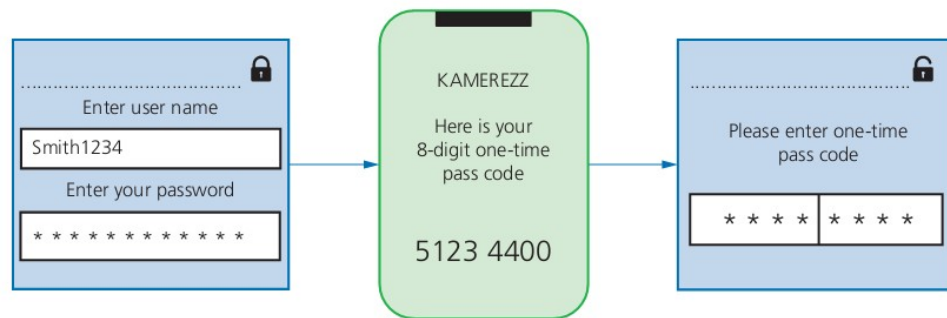
## 5 THE INTERNET AND ITS USES

### Two-step verification

**Two-step verification** requires two methods of authentication to verify who a user is. It is used predominantly when a user makes an online purchase using a credit/debit card as payment method.

For example, suppose Kate wishes to buy a new camera from a website. She logs into the website using her computer. This requires her to enter a user name and a password, which is step 1 of the authentication process.

To improve security, an eight-digit PIN (called a one-time pass code) is sent back to her either in an email or as a text message to her mobile phone (the mobile phone has already been registered by Kate on the website as the second stage of the authentication process). Kate now enters this eight-digit PIN into her computer and she is now authorised to buy the camera. In summary:



▲ **Figure 5.17** Two-step verification using a mobile phone

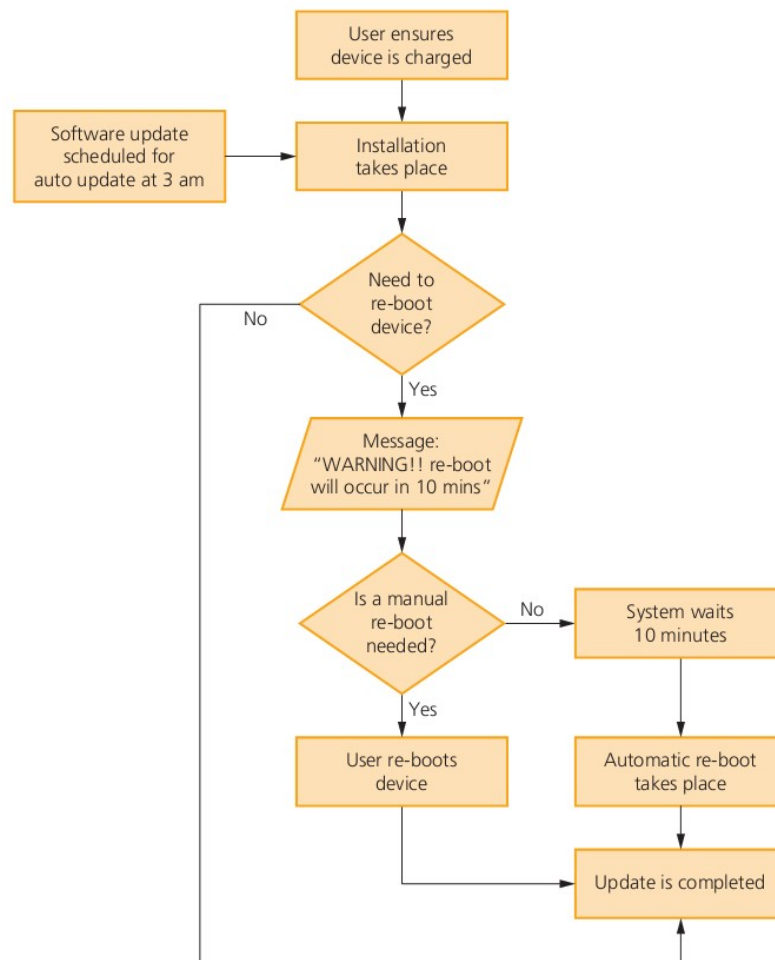
Using the definitions of authentication at the start of this section, the mobile phone is something she has and the password/PIN code is something she knows.

### Automatic software updates

Automatic software updates mean software on computers and mobile phones/tablets is kept up-to-date. Sometimes this is done overnight or when you log off the device.

These updates are vital since they may contain **patches** that update the software security (to protect against malware) or improve the software performance (for example, removal of bugs and addition of new features). The only downside to this is the potential for updates to disrupt your device following installation. If this happens, the user either has to wait for another patch to put this right, or use the techniques described in Chapter 4 that reverse the clock time to an earlier date before the updates were made.





▲ **Figure 5.18** Automatic software update flow chart

### Checking the spelling and tone of communication and URL links

When emails are sent to you, there are three actions you always need to take before opening them or activating any links in them.

- » Check out the spellings in the email and in the links; professional, genuine organisations will not send out emails which contain spelling or major grammatical errors (for example, Amazzon.com)
- » Carefully check the tone used in the email message; if it is rushing you into doing something or if the language used seems inappropriate or incorrect, then it could be a phishing email or worse.

## 5 THE INTERNET AND ITS USES

There are five things to look out for:

- 1 The email address itself; no legitimate company will use an email address such as: @gmail.com  
Carefully check the part of the address after the '@' symbol which should match the company's name; for example:  
account-update@amazon.com
- 2 The tone of the email and bad spelling of words is a clear indication of a potential scam. Look at this message that claimed it came from PayPal. See if you can find the ten errors in the email that should set off alarm bells.

**From:** PayPal <paypal@customer-notices55.com>  
**To:** PayPal user 551-121-998  
**Sent:** Feb 1<sup>st</sup> 2021 @ 10:55  
**Subject:** Compromised Account [CaseID Nr: KX-003-551-121-998]

Dear Customer

We need you help to resolve issue with account. We have temporarily stop account due to problem's.  
 Unusual account activity on PayPal account means action need be taken immediately. If your not sure this was you, an unauthorized user might be trying to access your accounts. Please to log in here to change your password:

[LOG IN HERE](#)

▲ **Figure 5.19** Sample scam email

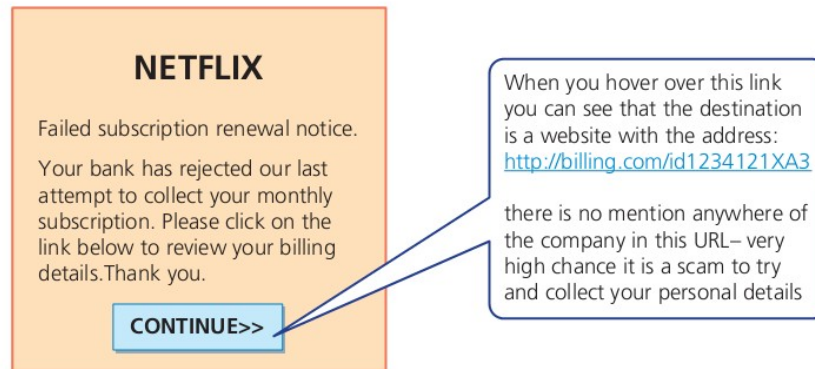
Did you find all the errors? An email like this looks official but there are many clues that it didn't come from a legitimate company; such as, many spelling mistakes, grammatical errors and the domain name in the email address. An email like this should be regarded as phishing; by clicking on the 'LOG IN HERE' box, you will divulge passwords and other key information since you will be sent to a fake 'PayPal' website.

- 3 Misspelling of domain names in a link are very common errors found in emails sent by scammers and fraudsters. The authors of this book have seen these incorrect spellings:

www.gougle.com  
 www.amozon.com

This is known as **typo squatting** where names close to the genuine names are used to fool you.

- 4 Suspicious links; destination addresses should match the rest of the email. Look at this message that claims to be from Netflix:



▲ **Figure 5.20** Second example of probable scam

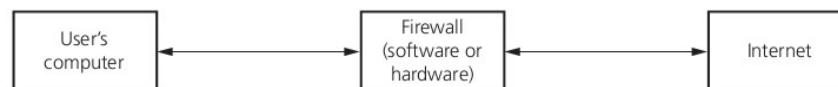
- 5 Other errors to look out for are just plain spelling mistakes. Look at this address from TKMaxx; find the three errors:

<http://www.tkmax.co.ie>

- » since the company involve online payments, it's very likely to use secure links therefore you would expect to see **https**
- » the spelling of the company is incorrect
- » it is more likely to see **.com** since they are a large company.

## Firewalls

A **firewall** can be either software or hardware. It sits between the user's computer and an external network (for example, the internet) and filters information in and out of the computer. This allows the user to decide whether or not to allow communication with an external source and it also warns a user that an external source is trying to access their computer. Firewalls are the primary defence to any computer system to help protect it from hacking, malware (viruses and spyware), phishing and pharming.



▲ **Figure 5.21** Typical firewall set up

The main tasks carried out by a firewall include:

- » to examine the 'traffic' between user's computer (or internal network) and a public network (for example, the internet)
- » checks whether incoming or outgoing data meets a given set of criteria
- » if the data fails the criteria, the firewall will block the 'traffic' and give the user (or network manager) a warning that there may be a security issue
- » the firewall can be used to log all incoming and outgoing 'traffic' to allow later interrogation by the user (or network manager)
- » criteria can be set so that the firewall prevents access to certain undesirable sites; the firewall can keep a list of all undesirable IP addresses
- » it is possible for firewalls to **help prevent** viruses or hackers entering the user's computer (or internal network)



## 5 THE INTERNET AND ITS USES

- » the user is warned if some software on their system is trying to access an external data source (for example, automatic software upgrade); the user is given the option of allowing it to go ahead or request that such access is denied.

The firewall can be a hardware interface which is located somewhere between the computer and the internet connection. Alternatively, the firewall can be software installed on a computer; in some cases, it is part of the operating system.

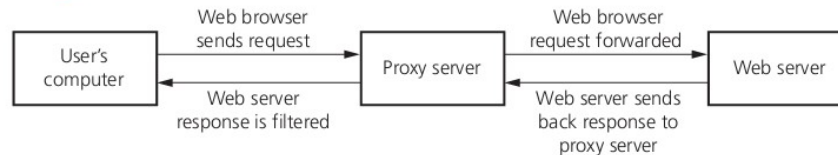
However, there are certain circumstances where the firewall can't prevent potential harmful 'traffic':

- » it cannot prevent individuals, on internal networks, using their own hardware devices (e.g. modems, smartphones) to bypass the firewall
- » employee misconduct or carelessness cannot be controlled by firewalls (for example, control of passwords or user accounts)
- » users on stand-alone computers can choose to disable the firewall, leaving their computer open to harmful 'traffic' from the internet.

All of these issues require management control or personal control (on a single computer) to ensure that the firewall is allowed to do its job effectively.

### Proxy servers

**Proxy servers** act as an intermediate between the user and a web server:



▲ **Figure 5.22** Proxy server

Features of proxy servers:

- » allows internet traffic to be filtered; it is possible to block access to a website if necessary
- » keeps users' IP addresses secret which improves security
- » if the internet traffic is valid, access to the web server is allowed
- » if the internet traffic is invalid, access to the web server is denied
- » it is possible to block requests from certain IP addresses
- » prevents direct access to a web server by sitting between the user and the web server
- » if an attack is launched, it hits the proxy server instead – this helps to prevent hacking, DoS, and so on
- » used to direct invalid traffic away from web servers which gives additional protection
- » by using the feature known as a cache, it is possible to speed up access to information/data from a website; when the website is first visited, the home page is stored on the proxy server; when the user next visits the website, it now comes from the proxy server cache instead, giving much faster access
- » proxy servers can also act as firewalls.


### Privacy settings

**Privacy settings** are the controls available on web browsers, social networks and other websites that are designed to limit who can access and see a user's personal profile. They were discussed earlier in the section on access rights. Privacy settings can refer to:

- » a 'do not track' setting; the intention here is to stop websites collecting and using browsing data which leads to improved security
- » a check to see if payment methods have been saved on websites; this is a useful safety feature which prevents the need to type in payment details again (every time you have type in financial details, there will be a risk of data interception)
- » safer browsing; an alert is given when the browser encounters a potentially dangerous website (the undesirable website will be in a 'blacklist' stored on the user's computer)
- » web browser privacy options (e.g. storing browsing history, storing cookies)
- » website advertising opt-outs; a website may be tracked by any number of third parties who gather information about your browsing behaviour for advertising purposes
- » apps; for instance, the sharing of location data in map apps can be switched off.

### Secure sockets layer (SSL)

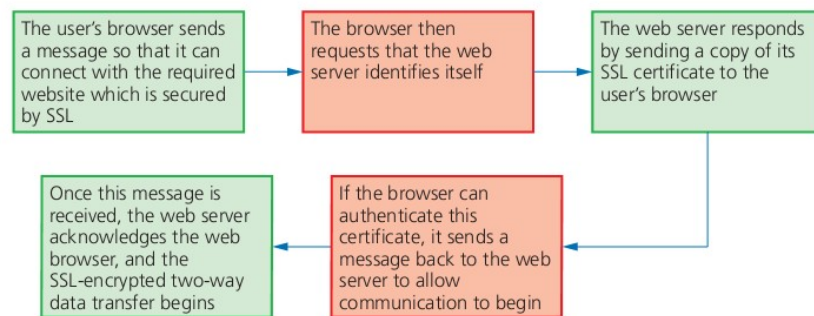
**Secure Sockets Layer (SSL)** is a type of protocol – a set of rules used by computers to communicate with each other across a network. This allows data to be sent and received securely over the internet.

When a user logs onto a website, SSL encrypts the data – only the user's computer and the web server are able to make sense of what is being transmitted. A user will know if SSL is being applied when they see https or the small padlock  in the status bar at the top of the screen.

The address window in the browser when https protocol is being applied, rather than just http protocol, is quite different:

using https:	 secure	https://www.xxxx.org/documents
using http:		http://www.yyyy.co.uk/documents

Figure 5.23 shows what happens when a user wants to access a secure website and receive and send data to it:



▲ **Figure 5.23** Secure sockets layer (SSL)

## 5 THE INTERNET AND ITS USES

The term **SSL certificate** was mentioned in Figure 5.23. An SSL certificate is a form of digital certificate which is used to authenticate a website. This means any communication or data exchange between browser and website is secure provided this certificate can be authenticated.

Examples of where SSL would be used:

- » online banking and all online financial transactions
- » online shopping/commerce
- » when sending software out to a restricted list of users
- » sending and receiving emails
- » using cloud storage facilities
- » intranets and extranets (as well as the internet)
- » Voice over Internet Protocols (VoIP) when carrying out video chatting and/or audio chatting over the internet
- » used in instant messaging
- » when making use of a social networking site.

### Activity 5.6

- 1 Which computer terms (used in this chapter) are being described below?
  - a a user is granted access only after successfully presenting two pieces of evidence to verify or identify who they are
  - b uses a cache to speed up access to web pages from a website
  - c controls that are used on social networks and other websites to allow users to limit who can access data from their stored profile
  - d protocol that is used to allow data to be sent securely over a network, such as the internet
  - e software or hardware that sits between a computer and an external network which monitors and filters out all incoming and outgoing traffic
  - f supplies domain names for internet hosts and is used to find IP addresses of domain names
  - g use of unique human characteristics to identify a user as a form of authentication
  - h made up of three types of identification:
    - something you know
    - something you have
    - something you are
  - i manipulation of people into breaking normal security procedures and best practices to gain illegal access to a user's computer system
  - j malicious code stored on a user's hard drive or web server used to redirect a browser to a fake website without their knowledge
- 2 Describe how SSL and TLS certificates are used to ensure that secure sharing of data between a browser and website takes place.
- 3
  - a Describe three things you should look out for when deciding whether or not an email is a potential phishing scam.
  - b Identify at least three potential problems with this email from a company called Watson, Williams and Co:

**From:** WW and Co <[accounts@customer.nr.012305555](mailto:accounts@customer.nr.012305555)>

**To:** customer 012305555

**Sent:** February 15<sup>th</sup> 2021 @ 13:45

**Subject:** Payment of January 2021 account

Dear WW & Co customer

We not able to take payments for account 012305555 on January 30<sup>th</sup>  
Please re-submit account details immediatly to the following address:

[Customer accounts link](#)



## Extension

For those students considering the study of this subject at A Level, the following extension to SSL may be of interest.

### Transport Layer Security (TLS)

Transport Layer Security (TLS) is a more modern and more secure version of SSL. It is a form of protocol that ensures the security and privacy of data between devices and users when communicating over a network (for example, the internet). It is essentially designed to provide encryption, authentication and data integrity in a more effective way than its predecessor, SSL. When a website and client communicate over the internet, TLS is designed to prevent third party eavesdropping which could cause a breach of security. TLS is comprised of two main layers:

- » record protocol – this part of the communication can be used with or without encryption (it contains the data being transmitted over the network/internet)
- » handshake protocol – this permits the web server and client to authenticate each other and to make use of encryption algorithms (a secure session between client and server is then established).

Only the most recent web browsers support both SSL and TLS which is why the older, less secure, SSL is still used in many cases (although very soon SSL won't be supported and users will have to adopt the newer TLS protocol if they wish to access the internet using a browser). The main differences between SSL and TLS can be summarised as follows:

- » it is possible to extend TLS by adding new authentication methods (unlike SSL)
- » TLS can make use of session caching which improves the overall performance of the communication when compared to SSL (see below)
- » TLS separates the handshaking process from the record protocol (layer) where all the data is held.

### Session caching

When opening a TLS session, it requires considerable computer time (due mainly to complex cryptographic processes taking place). The use of session caching can avoid the need to utilise as much computer time for each connection. TLS can either establish a new session or attempt to resume an existing session; using the latter can considerably boost the system performance.

### Summary

As already indicated, two of the main functions of SSL/TLS are:

- » encryption of data
- » identifying client and server to ensure each knows 'who they are communicating with'.

We will now consider how this is done:

#### Stage 1

Once the client types the URL into the browser and hits the <enter> key, several steps will occur before any actual encrypted data is sent; this is known as the handshaking stage.

#### Stage 2

The client's browser now requests secure pages (https) from the web server.

#### Stage 3

The web server sends back the TLS digital certificate (which also contains the public key) – the certificate is digitally signed by a third party called the Certificate Authority (CA).

#### Stage 4

Once the client's browser receives the digital certificate it checks:

- » the digital signature of the CA (is it one of those in the browser's trusted store – a list of trusted CAs is part of the browser which the client downloads to their computer)
- » that the start and end dates shown on the certificate are still valid
- » that the domain listed in the certificate is an exact match with the domain requested by the client in the first place

#### Stage 5

Once the browser trusts the digital certificate, the public key (which forms part of the digital certificate) is used by the browser to generate a temporary session key with the web server; this session key is then sent back to the web server.

#### Stage 6

The web server uses its private key to decrypt the session key and then sends back an acknowledgement that is encrypted using the same session key).

#### Stage 7

The browser and web server can now encrypt all the data/traffic sent over the connection using this session key; a secure communication can now take place.



## 5 THE INTERNET AND ITS USES

In this chapter, you have learnt about:

- ✓ the difference between the internet and the World Wide Web
- ✓ what is meant by a URL
- ✓ the purpose of hypertext transfer protocols (http and https)
- ✓ the purpose and function of a (web) browser
- ✓ how to locate and retrieve web pages from a website
- ✓ session and persistent cookies
- ✓ digital currency and blockchaining
- ✓ cyber security threats (brute force attacks, data interception, DDoS, hacking and social engineering)
- ✓ malware (viruses, worms, Trojan horses, spyware, adware and ransomware)
- ✓ phishing and pharming
- ✓ ways of alleviating cyber security threats (anti-malware, access levels, authentication, firewalls, proxy servers and SSL)
- ✓ improving security using automatic software updates, privacy settings and looking for security clues in emails and URL links.

### Key terms used throughout this chapter

**internet** – the world-wide interconnection of networks; the internet makes use of TCP and IP protocols

**World Wide Web** – a massive collection of web pages and is based on hypertext transfer protocols (http and https)

**(web) browser** – software that connects to a domain name server (DNS) to locate IP addresses; a browser interprets HTML web pages sent to a user's computer so that the user can read documents and watch multimedia

**hypertext mark-up language (HTML)** – the language used to design, display and format web pages, and to write http(s) protocols

**uniform resource locator (URL)** – a text-based address for a web page

**hypertext transfer protocol secure (https)** – http with extra security (such as SSL) applied

**hyperlink** – highlighted text or an image that is activated by clicking and links to further text, images, a web page or a website

**domain name server (DNS)** – a server that looks up domain names for websites (for example, www.hoddereducation.com) in order to find the IP addresses that a computer needs to locate the web servers (for example, 107.162.140.19)

**cookie** – a text file sent from a website to a user's browser; it is used to remember user preferences each time they visit the website

**user preferences** – settings or options stored in cookies that can remember customised web pages or indicate browsing history to target adverts

**session cookie** – a cookie that is stored temporarily on a computer; it is deleted when the browser is closed or the website session ends

**persistent cookies** – a cookie that is stored on the user's hard drive and only deleted when the expiry date is reached or the cookie is deleted by the user

**virtual shopping basket** – an area of memory in a website where items a user wishes to purchase are temporarily stored; items remain in the basket until payment is made or the session has ended

**digital currency** – currency (a system of money) that exists in electronic form only; it has no physical form and is essentially data on a database

**cryptocurrency** – a form of digital currency that uses a chain of decentralised computers to control and monitor transactions

**cryptography** – the protection of data/information by use of coding; it usually involves encryption and decryption

**blockchain** – a decentralised database where all transactions are stored; it consists of a number of interconnected computers but not a central server

**timestamp** – a digital record of the date and time that a data block is created in blockchain networks

**proof-of-work** – the algorithm used in blockchain networks to confirm a transaction and to produce new blocks to add to the chain; special users called miners complete and monitor transactions on the network for a reward

**brute force attack** – a 'trial and error' method used by cybercriminals to crack passwords by finding all possible combinations of letters, numbers and symbols until the password is found

**word list** – a text file containing a collection of words used in a brute force attack

**data interception** – an attempt to eavesdrop on a wired or wireless network transmission; cybercriminal often use



packet sniffing or access point mapping / wardriving to intercept data

**packet sniffing** – a method used by a cybercriminal to examine data packets being sent over a network and to find the contents of a data packet, which are sent back to the cybercriminal

**wardriving** – using a laptop, antenna, GPS device and software to intercept Wi-Fi signals and illegally obtain data; sometimes called Access Point Mapping

**wired equivalency privacy (WEP) encryption protocol security** – an algorithm for wireless networks to protect them against data interception

**denial of service (DoS) attack** – a cyberattack in which cybercriminals seek to disrupt the normal operation of a website by flooding it with requests; also used to clog up a user's mailbox by sending out thousands of spam emails

**distributed denial of service (DDoS) attack** – a denial of service (DoS) attack in which the fake requests come from many different computers, which makes it harder to stop

**spam** – unsolicited emails sent to a user's mailbox

**hacking** – the act of gaining illegal access to a computer system without the owner's permission

**malware** – programs (such as viruses, worms and Trojan horses) installed on a user's computer with the aim of deleting, corrupting or manipulating data illegally

**virus** – a program or program code that replicates itself with the intention of deleting or corrupting files or by causing the computer system to malfunction

**active host** – functioning software that a virus can affect by attaching itself to the code or by altering the code to allow the virus to carry out its attack

**worm** – a stand-alone type of malware that can self-replicate; unlike viruses, worms don't need an active host; they can spread throughout a network without the need for any action by an end-user

**Trojan horse** – a type of malware that is designed to look like legitimate software but contains malicious code that can cause damage to a computer system

**spyware** – a type of malware that gathers information by monitoring a user's activities on a computer and sends the gathered information back to the cybercriminal who sent out the spyware

**adware** – a type of malware that attempts to flood the end-user with unwanted advertising

**ransomware** – a type of malware that encrypts data on a user's computer and 'holds the data hostage' until a ransom is paid

**phishing** – sending out legitimate-looking emails designed to trick the recipients into giving their personal details to the sender of the email

**spear phishing** – similar to phishing but targeting specific people or organisations rather than carrying out a blanket attack

**pharming** – redirecting a user to a fake website in order to illegally obtain personal data about the user without their knowledge; unlike phishing, pharming is initiated without needing any action by the user

**DNS cache poisoning** – altering IP addresses on a domain name server (DNS) with the intention of redirecting a user's browser to a fake website; carried out by a pharmer (see pharming) or hacker (see hacking)

**social engineering** – manipulating people into breaking normal security procedures (such as giving away their password) in order to gain illegal access to computer systems or to place malware on their computer

**access levels** – different levels of access in a computer system allowing a hierarchy of access levels depending on user's level of security

**anti-spyware** – software that detects and removes spyware programs installed on a system; the software is based on typical spyware rules or known file structures

**authentication** – the process of proving a user's identity by using something they know, something they have or something unique to them

**biometrics** – type of authentication that uses a unique human characteristic, such as fingerprints, voice or retina blood vessel pattern

**two-step verification** – a type of authentication that requires two methods of verification to prove the identity of a user

**patch** – an update for software that is developed to improve the software and/or to remove any bugs

**typo squatting** – the use by cybercriminals of subtle spelling errors in website addresses used to trick users into visiting their fake websites

**firewall** – software or hardware that sits between a computer and an external network (for example, the internet); the firewall monitors and filters all incoming and outgoing traffic

**proxy server** – a server that acts as an intermediary server through which internet requests are processed; it often makes use of cache memory to speed up web page access

**privacy settings** – controls available on social networking and other websites which allow users to limit who can access their profile or what they are allowed to see

**secure sockets layer (SSL)** – a security protocol used when sending data over a network (such as the internet)

**SSL certificate** – a form of digital certificate which is used to authenticate a website; providing the SSL certificate can be authenticated, any communication or data exchange between browser and website is secure



## 5 THE INTERNET AND ITS USES

### Exam-style questions

- 1 a i What is meant by a **cookie**? [1]  
 ii Describe the difference between a **session cookie** and a **persistent cookie**. [2]  
 iii Give **three** uses of persistent cookies. [3]
- 2 A company has several offices. It uses the Internet to transfer data between offices. The company also makes payments to staff and suppliers using online banking.  
 The company are concerned about spyware and other security aspects of using the Internet.
- a Explain what is meant by spyware **and** how it is used to obtain data. [3]  
 b The company uses a web page to log on to the online bank. Identify **one** method that could be used by the online bank to reduce the impact of spyware when logging on. State **how** the method prevents the use of spyware. [2]  
 c The company has installed a firewall as part of its data security. Describe how a firewall can help protect against unauthorised access to data. [4]  
 d State **two** other methods the company could use to help prevent unauthorised access to data.  
 Method 1  
 Method 2 [2]
- 3 Six statements are shown on the left and six computer terms are shown on the right.  
 By drawing lines, connect each statement correct term. [6]

set of rules that must be obeyed when transferring files and data across the internet

software used to access, translate and display web pages on a user's screen

collection of multimedia web pages and other information on websites; these resources are accessed by a browser

worldwide collection of interconnected network computers that make use of TCP and IP protocols

small file or program downloaded when user visits a website; it remembers user preferences and other data

financial system which allows the transfer of funds and purchasing items electronically

cookie

World Wide Web (WWW)

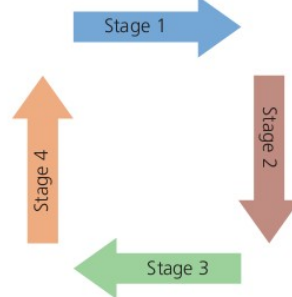
digital currency

hypertext transfer protocol (http)

internet

web browser

- 4 a** John uses two step verification when purchasing some items from a website. There are five stages in the process. These stages are listed below but are not in the correct order.  
Place the five stages into their correct order.
- A – user takes note of the one-time authentication code
  - B – user enters the one-time authentication code into the original device
  - C – user enters website user name and password into device
  - D – user is authenticated and allowed access to website to order items
  - E – one-time authentication code sent to user's email address
- b** One form of authentication is fingerprint recognition. A school is using fingerprints to uniquely identify each student. The system is used to act as a register instead of the existing manual system. Describe how fingerprint recognition can be used so that the school knows exactly which students are presently attending.
- 5 a** Describe four ways cybercriminals can use to trick a user into downloading malicious code onto their computers using social engineering.
- b** There are four stages in the course of action when a cybercriminal targets an individual using social engineering. Describe each of the four stages in the diagram below which depicts these stages.



- c** Some cybercriminals have decided to hack into a company's financial system.  
Customers buy goods using digital currency.
- i** How does digital currency vary from traditional fiat currency?
  - ii** Explain how blockchaining could protect the company and the customers from hackers.

## 5 THE INTERNET AND ITS USES

- 6 HTML can be used to create the structure and the presentation of web pages.
- a Describe what is meant by HTML structure. [2]
- b Gloria writes a paragraph as an answer to an examination question about accessing a website.  
Use the list given to complete Gloria's answer by inserting the correct **four** missing terms. Not all terms will be used.
- browser
  - cookies
  - Hypertext Markup Language (HTML)
  - hypertext transfer protocol (http)
  - hypertext transfer protocol secure (https)
  - Internet Protocol address (IP address)
  - Media Access Control address (MAC address)
  - web server

The user enters the URL of the website. The \_\_\_\_\_  
uses the DNS server to look up the \_\_\_\_\_ of the website.

The browser sends a request to the \_\_\_\_\_ to obtain  
the website files. The website files are sent as \_\_\_\_\_  
that is interpreted by the browser. [4]

*Cambridge IGCSE Computer Science 0478, Paper 11 Q9, Oct/Nov 2019*

- 7 An art gallery has a website that is used to display and sell art.
- a The gallery uses Secure Socket Layer (SSL) to provide a secure connection when selling art.  
Describe the process of SSL and how it provides a secure connection. [6]
- b The art gallery also uses a firewall.  
**Six** statements are given about firewalls.  
**Tick (✓)** to show if the statement is **True** or **False**.

Statement	True (✓)	False (✓)
Firewalls are only available as hardware devices		
Firewalls allow a user to set rules for network traffic		
Firewalls will automatically stop all malicious traffic		
Firewalls only examine traffic entering a network		
Firewalls encrypt all data that is transmitted around a network		
Firewalls can be used to block access to certain websites		

[6]

*Cambridge IGCSE Computer Science 0478, Paper 11 Q8, May/June 2019*