OPCUG The Ottawa PC Users' Group
Mailing Address - 3 Thatcher Street, Nepean, Ontario, K2G 1S6
JUNE 1990 (613) 723-1329 (voice) (613) 723-7064 (BBS) VOL 7, NO. 6

A LOOK AT COMPUTER VIRUSES

By John Whelan

The following combines two reports by Richard Perron, a computer specialist working in security in the federal government. It deals with the threat of viruses at the organisational rather than at the individual level, but most of the material will be of use to everyone.

Computer viruses have become a hot issue in computer security. The amount of recent articles, including those in the Globe and Mail, the feature article in Time, not to mention all the EDP literature, has made everyone aware of the threat, and in so doing, may have unnecessarily caused some of us to be paranoid.

A computer virus is a program written to disrupt the normal operation of a computer system. It normally hides inside legitimate computer programs, and under certain conditions set by the programmer, will trigger itself and perform actions ranging from writing obscene messages on the screen to erasing the whole hard disk.

Even though a virus can be written for any computer type, you should concern yourself with microcomputers viruses.

A classic infection scenario is as follows. From an electronic bulletin board or a friend of a friend, you receive a contaminated program which is usually found in games, home-made utilities or pirated software. While you try out the program, the virus looks for other programs and hides inside them. From that point on, the contamination spreads to any other computer systems where the contaminated utility or program has been loaded.

Some indications of contamination are: slower performance of systems, erratic write activities to disks and recurrent abnormal termination of programs. It is often impossible to ascertain if the symptoms are caused by a virus, a bug in the program, or a hardware malfunction. These questions are very hard to answer and could lead someone to blame everything on viruses. So what can be done about it?

Well, there's good and bad news. The good news is that many antidote and vaccine programs exist for PCs. The bad news is that there are many types of viruses which are difficult to identify and pinpoint, and the antidote and vaccine are not easy to interpret and use.

Once hit by a virus, it may take one to two weeks to identify the virus, restore the data and resume normal operations. Depending on the criticality of your systems, you may want to implement some or all of the following recommendations:

I. Ensure that employees are aware of the existence of viruses and of the measures that have been adopted to protect the operational environment.

II. Only use software supplied by the department or a reliable vendor.

III. Remind the staff that computers should be for authorized use only (i.e. no personal use, no games).

IV. Eliminate illegal copies of software.

V. Control access to microcomputers. During idle time, do not leave modems and microcomputers on.

VI. Maintain an effective backup/restore procedure.

VII. Conduct a reliability check before accessing virus-prone sources.

VIII. Set up a quarantine machine with a few antidote and vaccine programs to test foreign programs.

IX. Develop a contingency plan to support operations without the computer and data stored on them.

More technical information may be found in PC Magazine, Volume 8, Number 8, April 25, 1989, Pages 193 to 228.

(Continued on Page 2)

NEXT MEETING

The next meeting of the Ottawa PC Users' Group will be held on June 27, 1990. The Guest Speaker will be Guy Wardle from Microsoft Canada Inc.

The topic will be the recently released Windows 3.0. There will also be draws for prizes.

FROM THE EDITOR

Hi folks! The second part of the upcoming general meeting will be devoted to the proposed changes to the current OPCUG Constitution. Please bring the copy of the proposed constitution which you recently received in the mail. Please do forget your OPCUG Identification card as well. You will need it in order to be able to vote on the proposed changes.

A big thank you is extended to Jamie Simser of WordPerfect Corporation for a fine presentation of DrawPerfect, and congratulations to Carl-Henri Gomez who was the winner of some DrawPerfect clipart diskettes.

Happy reading!

* * *

CONTENTS

Meetings are held at 100 Sussex Drive, Ottawa
Last Wednesday of the month - 8:00 p.m. to 10:00 p.m.
No meetings in July and December

A LOOK AT COMPUTER VIRUSES

(Continued from Page 1)

A TYPICAL POLICY: EDP VIRUSES

Given the potential threat of EDP viruses, security measures must be taken to provide EDP support.

1. When the installation procedures for a software allow it, write-protect all diskettes.
1. Centralize the acquisition of software and hardware that will assist in the prevention, detection or correction of EDP viruses (e.g., IBMSCAN and McAfee and Associates' products).
2. Scan any systems or hard drives received new or coming back from a repair shop.
3. Develop contingency plans and assign responsibilities to deal with all instances of malicious software attacks. The plans should include the following:
   a) When the malicious software attack is detected, inform the user and quarantine all machines and media that may have been shared;
   b) Inform an EDP Security Coordinator who will inform management, the Information Centre, and the RCMP;
   c) Identify the virus and its source. This is normally done by the local manager, Systems and Operations, with the help of the EDP Security Coordinator;
   d) Determine the extent of the infection;
   e) Where possible, salvage any data files on the machines while excluding any files of the type COM, EXE, SYS, BIN and CNF;
   f) Clean up the machines, the backups and the media;
   g) Recover the software programs from reliable backups or originals;
   h) Monitor affected PCs to guard them from reinfection; and,
   i) Update the contingency plan to reflect the experience.

Hobbyists or home users should avoid suspect programs. If you download a .zip or even a virus infected program, and you get a warning BEFORE you have run the program, delete the program or the archived file. Your machine will not be contaminated unless you have actually run the infected program. Connecting to a BBS that has contaminated software on it will not cause your machine to become infected; you have to download files and subsequently run them. Do not depend on McAfee's, other scanning programs or on programs that detect direct writes to the hard disk. Some telecommunications programs write legitimately directly to the hard disk for speed reasons, some viruses write via DOS and cannot be detected by these programs.

When loading your machine from scratch, make a printout of all directories including the date stamp and size. Most viruses work by adding a small amount of code onto the end of a .com or .exe file. So, if you suspect a virus, check the file sizes and date stamps against those on the printout. A prime target is command.com.

Some bulletin boards are much better than others as a source of virus-free programs. As a rule of thumb, any board that allows anyone to join without any identity checks and solicits junk files by upload and download quotas should be avoided at all costs. Others such as the Ottawa PC Users' Group go to considerable lengths to check their files. A series of tests that will normally screen out most viruses can be performed on programs. I won't list them here for obvious reasons, but if you are a software librarian for a user group, you should be aware of them.

If you are hit, the best thing is to boot up the machine with a write-protected DOS bootable diskette and reformat the hard disk. Then reload the software from known sources. The virus that has a long incubation period is the most troublesome to clear, especially since your backup stretching back over a period of time may be contaminated. While your data files should be virus free, they may have had data deleted or added at random. If you can pin-point the type of virus you have been contaminated with, it may be possible to recover the data.

* * *

HANDY LESSONS FOR FIGHTING A COMPUTER VIRUS

By Michael Goddard and Marcel St. Pierre

We recently had the pleasure of fighting our first virus on a computer at work. After a lot of fumbling around, Marcel St. Pierre and I had the beast (of the Jerusalem B ilk) under control. After the dust settled, we realized that we had learned a lot from the experience and that the next time around, the fight would be different. We thought that members might be able to benefit from our experience and possibly put on a better first show than we did!

We used the McAfee utilities to detect and disinfect the victim's computer. Unfortunately, the latest CLEAN and SCAN files were kept in .zip format. Of course we could not trust the victim's copy of SCAN. However, we were not careful with the diskettes that we carried from our machines to his, and, in the heat of the battle, there was one point where we thought we may have infected one of our own machines.

This exchange of diskettes was partly due to the fact that our latest versions of McAfee's programs were still in .zip files.

Lesson one was that we should have quarantined any diskettes that may have been used on the affected machine.

Lesson two was that we should have unpacked the utilities, put them on a separate floppy and made the floppy bootable and write-protected. We now do this with every new version of a utility we receive.

Midway through the experience, SCAN started acting strangely. We would validate SCAN, and all the checks were satisfactory. Then we would run SCAN, and it would report that it detected a problem with itself! For an embarrassingly long time, we suspected that we had uncovered a new virus which modified SCAN in such a way that VALIDATE worked. However, the virus was not clever enough to escape SCAN's self detection. The virus led SCAN to think it was a common or garden Jerusalem B type. Thus, our copy of SCAN seemed to be the culprit! We then retrieved older copies of SCAN to check our suspicions.

Lesson three taught us to treat McAfee utilities like backup files. We now follow a grandfather style system and keep the latest three versions on write-protected self-booting diskettes.

Our suspicions, however, were wrong. The problem was traced to a copy of VALIDATE that we had infected ourselves by not quarantining floppies. The infected VALIDATE processed SCAN perfectly, reported the statistics we expected and THEN infected SCAN. Sharp!

Detection and cleanups were straightforward. We installed the virus shield on the user's machine just to be safe, and it was used after the event.

(Continued on Page 3)

HANDY LESSONS FOR FIGHTING A COMPUTER VIRUS

(Continued from Page 2)

Tracing the source of the virus turned out to be easy, and it provided to be a really good tip. The user shared programs with few others. He still had all the diskettes he had loaded onto the hard drive from the preceding couple of weeks. We Scanned these and eventually one yielded the Jerusalem B message.

The diskette was one of a series in a database system provided by a contractor in Toronto. We were delighted to see a write-protect tab on it. So, the contractor had write-protected the diskettes, but the victim had not added the protection. That eliminated the possibility that the diskette had been infected by the affected machine. This taught us lesson four.

NOTE: When dealing with virii, it is important to remember that all scanning and validation programs are either .COM or .EXE files. If these programs are located on a hard drive, they are just as susceptible to infection as any other program. This is what led to our wild goose chase.

* * *

ORGANIZATION OF FILES ON A HARD DISK

By Eric Clyde

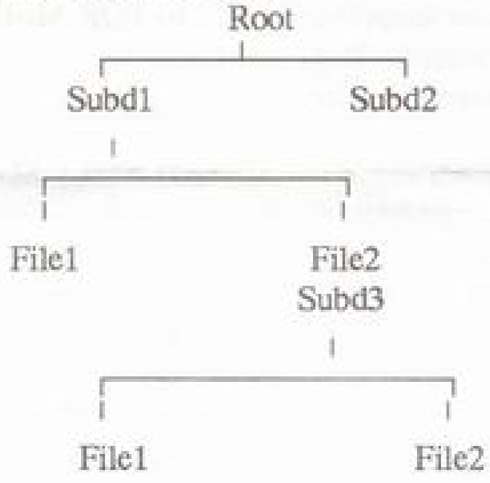
Perhaps the most important enhancement to a microcomputer system is a hard disk, which can hold the equivalent of 60 or more double-density floppy diskettes and speed up access to the files by a factor of ten or more. A hard disk is a fragile piece of hardware and should be handled carefully. When the computer is turned off, park the disk heads using a utility such as PARK, SHIPDISK. normally supplied with the machine. Back up all important files regularly and test the backups to verify that they work.

SUBDIRECTORIES

Hard disks can hold thousands of files, but most can only store 512 in the root directory (actually 509 plus two hidden files and the volume label file). On a new hard disk, this may seem like an unreachable number, but if you use a number of software packages or buy club disks, many files can be generated over a period of time.

After you have been using your computer for a few months, it is frequently difficult to remember the name of an older file you need or what a cryptic filename means, since DOS puts a limit of eight characters to a file name, plus three for the extension (e.g. CONVERT.EXE). Grouping files in subdirectories makes it easier and faster to find the necessary files.

Many software packages have files with the same name, (e.g. READ.ME). If these packages are installed in the same directory, the latest READ.ME file will overwrite the others.



1. Files with the same name in different subdirectories are not necessarily the same file: e.g., the two File1's and the two File2's files above are not necessarily identical.
2. The full name of File1 under Subd1 is C:\Subd1\File1. Under Subd3, it would be C:\Subd3\File1.
3. Keep things simple. It is theoretically possible to have 32 levels of subdirectories (if the subdirectory names are one letter each, (e.g.\a). Only on large hard disks is it necessary to have more than three levels, e.g. the root directory C:\, C:\Subd1 and Subd1\Subd2. This cuts down on the amount of typing required (see the substitute command below).
4. Unless you are running a DOS shell program, use short but informative file names.

PATH COMMAND

Only three files (plus two hidden files) are necessary in the root directory: command.com, autoexec.bat and config.sys. Executable programs and routines (those with extensions .com, .exe and .bat) can be run from anywhere on your hard disk if the subdirectories containing them are listed in the PATH statement in your autoexec.bat file. A path such as:

```
Path=C:\;C:\DOS;C:\UTILS
```

would allow access to any executable files in the root directory, the \dos or the \utils subdirectories from any other subdirectory. Thus, if you are in the subdirectory \WP and wish to format a diskette in drive A, your path statement would allow you to invoke the format command in \dos.

Many programs access information from other files, so it is important that the path reflects this or that you use the SET parameter (see your DOS manual for more information on this). For this purpose, it is also possible to use a utility such as DPATH30.

SUBST COMMAND

If you use many levels of subdirectories, the SUBST command can help you. Take for example the line:

```
SUBST E: C:\level1\level2\level3\level4
```

Instead of typing:

"C:\level1\level2\level3\level4\filename", you can simply type "E:filename". By putting the line "lastdrive=z" in your config.sys file, you can use the SUBST command for each letter of the alphabet, except A, B and C.

* * *

MEMBER-GET-A-MEMBER 1K RELAY!

More than $1000.00 in prizes for the winner of the 1K Relay!

Become a sponsor for a new member of the Ottawa PC Users' Group and your name will be entered for the SuperDraw as many times as you convince a new person to become a member.

From Carl-Henri Gomez
Your Membership Chairman

BASIC LOOPS

By Harry Gross

Loops are one of the fundamental control structures of any language. In interpreted BASIC, there are only two structures available to us: "FOR .. NEXT" and "WHILE .. WEND". In the compiled versions, we have another useful structure available - the "DO .. LOOP".

Let's consider the FOR .. NEXT loop. It has the full format:

```
10 FOR N=J TO K STEP L
..... do something more or less useful...
100 NEXT N
```

J,K, and L may be integer, single precision, positive or negative numbers. STEP L is optional and, if omitted, defaults to 1. As the loop goes through the procedure the first time, it does so with the counter N set to the value J. With each pass through the loop, the counter is incremented by the value L and compared with K. If it is equal to or greater than K, control passes to the statement after NEXT N. So a simple loop would look like this:

```
10  FOR N=1 TO 13 STEP 3
20  PRINT "Number = " N, "Squared = " N*N, "Cubed = " N*N*N
30  NEXT N
30  PRINT "N at end of loop - " N
```

And the screen would look like this:

```
Number =  1     Squared =  2     Cubed =  3
Number =  4     Squared =  16    Cubed =  64
Number =  7     Squared =  49    Cubed =  343
Number =  11    Squared =  121   Cubed =  1331
N at end of loop -  14
```

The last line illustrates a small but potential source of trouble, the value of the counter is live after the loop and should be treated with caution. Be aware that it is not equal to the loop limit. If we use a negative increment, the loop would be behave as follows:

```
10 FOR N=13 TO -2 STEP -3
20  PRINT "Number = " N, "Squared = " N*N, "Cubed = " N*N*N
30 NEXT N
30 PRINT "N at end of loop = " N
```

The screen would look like this:

```
Number =  13    Squared =  169   Cubed =  2197
Number =  10    Squared =  100   Cubed =  1000
Number =  7     Squared =  49    Cubed =  343
Number =  4     Squared =  16    Cubed =  64
Number =  1     Squared =  1     Cubed =  1
Number = -2     Squared =  4     Cubed = -8
N at end of loop =  5
```

Loops may be nested, one inside another, as follows:

```
110 PRINT "     Area Table"
120 PRINT "Length     Width";
130 FOR W = 4 TO 12 STEP 2
140   PRINT USING "  ####";W;
150 NEXT W
160 PRINT
170 FOR L= 5 TO 10
180   PRINT USING "  ###            "; L;
190   FOR W = 4 TO 12 STEP 2
200     PRINT USING " #####";L*W;
210   NEXT W
220   PRINT
230 NEXT L
240 END
```

For nested loops that all end in the same statement, such as:

```
90    x=J*K*L
100     NEXT L
110    NEXT K
120  NEXT J
```

the following may be used.

```
100 NEXT L,K,J
```

Remember, the order of the counters must be observed. For a simple one counter loop, NEXT instead of NEXT N is allowed, but should be avoided as it could make debugging more difficult.

In using nested loops, it is very important that they be nested, and not crossed. The following is correct.

```
100 FOR N=1 TO 100 ———————— range of N

110 FOR M=1 TO 20 ——— range of M

  ...........

200 NEXT M

210 NEXT N
```

The following is not:

```
100 FOR N=1 TO 100 ——— range of N

110   FOR M=1 TO 20
  ...........

200   NEXT N               range of M

210 NEXT M
```

Having shown how to get into a loop, the next item on the agenda is how to get out of it. There is no good way, other than to jump to a line number.

```
10 FOR J=1 TO 10
20   FOR K=1 TO 10
30     FOR L=1 TO 10
40       x=J*(K^L)
50       IF x> 1000 THEN GOTO 230
..............
160      NEXT L,K,J
230 ' number too large
240  .........
```

One important point to remember is not to change the value of the counter while in the loop. In a long loop, the programmer could lose track of variables in use and re-use the counter as another variable. This is most likely to happen with subroutines.

The above examples have used simple variables as for the loop, but we are not restricted to them. We may use arrays,

```
10  FOR N=1 TO 20
20  FOR K= A(N,1) TO A(N,2)
................
100  NEXT K,N
```

expressions,

```
10 A=3.5
20 FOR N=A^2 TO A^3 STEP .2
```

(Continued on Page 5)

```
30  PRINT N,SQR(N)
40 NEXT N
```

or functions.

```
10 PI=3.141596
20 FOR N=SIN(0) TO SIN(PI/2) STEP .1
...........
80 NEXT N
```

We have not yet introduced GOSUB routines, but the situation may be illustrated in the following:

```
10 FOR N=3 TO 6
20 GOSUB 1000
30 PRINT "N = " N, "Factorial N = " F
40 NEXT N
950 END
1000 ' a routine to calculate Factorial N
1010 F=1
1020 FOR K=1 TO N
1030   F=F*K
1040 NEXT K
1050 RETURN
```

The listing will appear as:

```
N = 3      Factorial N = 6
N = 4      Factorial N = 24
N = 5      Factorial N = 120
N = 6      Factorial N = 720
```

No problem, everything works as it should. But later on, we decide to use the same routine again inside another loop, but instead of N, we use K as our counter.

```
210 FOR K=1 TO 10
220 GOSUB 1000
230 PRINT "K = " K, "Factorial K ",F
240 NEXT K
```

The listing will now show:

```
K = 1      Factorial K = 1
```

and keep on repeating this in an endless loop.

What happens is that the same counter K is now used in both loops and the sub-routine keeps on resetting it back to 1. This problem alone is enough reason to consider using a compiler with subroutines that confine the effect of variables to the subroutine alone, unless made effective over the whole program.

The next article will discuss the WHILE ... WEND Loop.

* * *

A REVIEW OF GRAMMATIK IV

By André Lefrançois

Most word processors on the market today have spell-checking and thesaurus features. Developers and users are starting to employ electronic grammar verification and electronic translation. I will focus my attention on grammar checking.

Style and grammar verification, along with spell checking, provides a sound proofreading solution. Few word processors offer both grammar and style verification. However, these features are available in stand-alone products. The two most popular packages are "Grammatik IV" and "RightWriter". This article is a review of Grammatik IV, published by Reference Software.

Grammatik IV may be executed through one of three modes. In stand-alone modes, the program is launched from the DOS command line or from a batch process.

The third mode allows users to execute Grammatik IV from within a word processing package. This "pop-up" access to Grammatik IV is available for Wordperfect 5.0 or later;, Microsoft Word 4.0 or later;, PFS Professional Write 2.1 or later;, and Wordstar 5.5 or later.

The product supports several other popular packages in stand-alone mode (including Ventura Publisher). The documentation and installation instructions are clear and helpful. The program is not copy protected.

Grammatik IV features a very user-friendly "a la Microsoft" interface which lacks mouse support. The package proofreads your documents for mistakes in grammar, style, usage and spelling. A new Writing Style menu allows the user to tailor Grammatik IV to one of the following writing styles: General, Business, Technical, Fiction, Informal or Custom.

Because grammar and style is user-specific, the software doesn't correct mistakes, but rather allows users to either tag a situation, or edit directly "a la Wordperfect Spell Check". On finding a mistake, pressing F1 gives you a full background about the mistake, as well as a tutorial on how to avoid it via grammar rules. This makes Grammatik IV a good educational program as well as a powerful business tool.

The system offers good advice, which you can also choose to ignore. After completing its proofread, the system provides statistical information on the grade level of your writing. Over time, this utility should improve your writing skills.

Grammatik IV has addressed weaknesses of Grammatik III by adding execution within Wordperfect and through better user interface, although mouse support would be welcomed. For $72.00, this package is well worth the cost and is a must for anybody who writes regularly.

* * *

FUNCTION KEYS F11 AND F12

By Morris Turpin

I use Reveal Codes and Block more than any other function keys. My AT at work supports F11 and F12, and I became used to the convenience of these two extra keys. Unfortunately, my home keyboard does not send any codes for F11 or F12. (The keyboard is an old model bought with my original XT.)

I got around the problem by defining my own keyboard within WordPerfect. The "*" on the numberpad is defined as Reveal Codes and the "-" on the numberpad is defined as Block. Because these keys use different scan codes than their counterparts on the main keyboard, I do not lose the functions of these keys.

My keyboard definition is made with the NumLock on. With NumLock off, I can still use the "+" and "-" keys on the keypad for screen up and screen down. I've been using this keyboard version for several months without any problem. It's much more convenient than ALT-F3 or ALT-F4.

* * *

A BRIEF REVIEW OF TELIX

By John Whelan

Well, it has to be brief because I've never really found any limit to what Telix can do. Telix is a telecommunications program that evolved following the path of PC-Talk. We all have our own favourite telecommunications package and Telix is one of mine. Using PC-Talk on a PC many years ago, we managed to achieve 2,400 baud. Above that, on a direct connect, the program would lose characters. Using Cross-Talk, we could run at 9,600 bps. These direct-connection modems at the time really weren't usable at these high speeds.

There is a technical reason why the maximum baud rate is 2,400. If you go faster, you refer to the peed as bps or bits per second. Each serial character takes 10 bits to transmit, 8 bits for the character, 1 bit to say it's coming and one to say it's been. The ideal 2,400 baud or bps modem transmits and receives 240 characters per second.

The interfaces on these programs still make me shudder when I think of them. If you are using either of them and still think it's the bee's knees, give yourself the chance to try Telix: it's worth a shot. Telix will cope with 9,600 bps on an XT, but you need more CPU horse power to get the throughput. By the way, if you happen to have a liquid nitrogen-cooled 486 and plenty of RAM disk space, it will take you up to 115,200 bps.

The first time I used Telix, I was impressed by the message that said "Alt Z" for help. I typed in my usual ATDP560 whatever and it worked. Later I found the dialling directory. At first, I thought this was a great idea. I didn't have to have scraps of paper pinned all over the wall with phone numbers written on them. I had stopped writing them on the blackboard in the office when, one night, the new cleaner cleaned off my blackboard which contained my appointments and meetings for the following three weeks.

About this time, I decided I would run a BBS just for fun. I noticed something strange. I could tell which telecommunications package the caller was using by the length of time it took to download a file. This was because Telix made intelligent guesses at the name of the file to be downloaded, and partly because it was simply faster at processing Xmodem blocks or whatever and sending back the acknowledgements. I tried out many telecommunications packages and found that Telix managed to download files faster than all the others. Although these tests were done sometime ago, I still think that Telix is one of the faster programs.

One of the problems I came across early on was the bit of the list of files I was interested in had just gone sailing off the screen. <Alt B> gives you a window which lets you set the Pub more off and scroll your way through any interesting messages or list of files.

Telix is pretty much like many other telecommunications packages. It has a script file language. It also has a configuration file or environment file which tells Telix about a specific configuration's setup.

I started working with the script files and found that with just a few lines of code, I could save myself a considerable number of keystrokes. Besides, it was better at typing my password than I was. Using a technique of asking the user questions or processing an input file containing directives (JOHNPUB) through a Turbo Pascal program, non-technical users can use Telix in a very flexible way. Many secretaries and others use Telix front-ended by a series of Pascal programs to upload and download files from an HP 3000 using Kermit without even knowing Kermit. Kermit file transfers to and from an HP3000 took me a few days to work out.

The library wanted to access many different databases, each of which required different settings for parity, character mappings, etc. We wanted to use direct DATAPAC connections since they were being based on time connection at a high-dollar rate per minute; so we had a multitude of different equipment to look through. We found out later that modem-to-modem connections are simple (only two devices around), but when you use data switches into xmuxes into X.25 with different channels, and at the other end they have similar setups, if a single connection or X.3 parameter is wrong, it doesn't work most of the time. Telix was more forgiving and had enough flexibility that we could send with one set of parity and receive with another. The client in this case is presented with a menu giving the different databases, presses the button and gets logged on, all via scripts and Telix config. files.

Telix's script language Salt allows you to "track" or watch out for a specific string of characters, and to take action according to which string comes out. This incredibly useful tool allows you to take the correct action on any error message or to cope with the questions that TBBS asks you such as "*Enclosed file" or "-Press Any Key-". This can be done with other script languages, such as Reflection's but you have to capture each character in turn, store it in a buffer, then compare the buffer to a character string. Not my idea of fun. Salt, by the way, is very similar to C and is very powerful and flexible. A couple of sample scripts are included for your guidance.

The documentation is fairly good, but aimed more at the technical user than many other packages. Most beginners can fare better by ignoring the documentation and relying on the <Alt Z> key. The use of scripts can make your life very easy, but many less technically inclined people make use of the keyboard editing feature. Typically, they might remap one or more function keys to send a logon sequence with password as either a single keystroke or a two-keystroke combination. The embedded strings allow you to specify time delays within the strings. Most users can use all of this by following the help system and exploring the options.

For the more technically advanced user, Telix allows external protocols. If it doesn't have the one you want internally, and it comes with a set of twelve different protocols, you can use an external one. Other features are the ability to set up default paths for uploaded and downloaded files. To download, use the <Pg Dn> key. To upload, use the <Pg Up> key. You also have the normal options of using hardware flow control (CTS/RTS) (DSR/DTR) which are pretty well essential when using high speed modems (9600 bps and above). It has built-in translation tables which enable you to remap characters sent out or received. So, with code page 863 or 850, you can access an HP 3000 which uses the Roman 8 character set and see the French characters correctly displayed. It was actually the first product I found that could do this but now some AdvanceLink and Reflection programs have caught up. Quite useful when handling the EPSIDIC (Sure I've spelt epsidic wrong)(the IBM Mainframe) character set. It also offers a basic VT100 terminal emulation which is useful for VAXs, UNIX machines, running

(Continued on Page 7)

POWERHOUSE and getting pretty displays from BBSs. When dialling, you can give it a string of numbers to dial, and if it doesn't connect the first number, it goes onto the next. It then cycles through the list of numbers again, which is useful in accessing busy bulletin boards. Delays can be set so that it will wait a couple of minutes before trying again. It can also be configured to give warning sounds when it gets through.

I think I've covered enough to give a flavour of the product. Colin Sampaleanu wrote the product and lives in Toronto. The technical support from EXIS is some of the highest quality I've seen. The company is small but will tailor the product to clients' needs. Mind you, buying a few hundred copies helps if you would like the product modified. Most users find the echo mail support useful and sufficient to their needs.

Although I'm sure there are packages that do a little more (perhaps allowing you to download a file whilst composing a message), Telix is one of the telecommunications classics and an excellent place to start your search for your own favourite telecommunications package.

* * *

SPEAKERS

By Marty Sells

Every sound generated by a PC comes out of a speaker installed inside the case. These speakers are generic 1/4 Watt, 8 Ohm speakers on which anyone with basic electronics knowledge can hook up a switch in order to control the current.

Basically, unplug the Molex connector that attaches the speaker to the motherboard. If your speakers' connectors are soldered to the motherboard, either desolder them or carefully cut them with nail clippers or wire cutters.

A more hi-tech approach is to splice one of the wires and connect a small switch (available at Radio Shack) in between them that can easily fit through a carefully drilled hole in the case. If your're squeamish about drilling a hole in your metal case, there are several alternatives:

1. If you have a flip-top case, leave the switch inside but not touching anything. To turn the sound on or off, open the case and flick the switch.
2. Drill a hole in a bezel mounting plate (the ones used to cover empty drive bays) and run a wire out to the plate. This setup gives you front panel access to the switch, and if you ever need to clean the machine up, just buy a new bezel plate. (Try the Flea Market at Stitsville for these plates.)
3. Run the switch out of the back panel (where the I/O cards have their connectors). Again, you'll need a length of wire to do this. Some cases (mine) have unused holes in the back that are smaller than the I/O slots that are meant for DB9 connectors.
4. Disconnect the lock and wire up the speaker to that unused switch.

These alternatives will suit different people. My current setup is just a loose switch. I recommend the bezel plate method to those of you who have full-sized ATs or a drive bay.

* * *

ATI BULLETIN BOARD

By Robert Parkinson

ATI in Scarborough has a bulletin board on which they post new drivers for their video cards, as well as information on their modems, etc. It operates 24 hrs a day, 7 days a week. The BBS number is 416-756-4591. The settings are "N 8 1".

* * *

PUB TALK

QUERY: BONNIE CARTER
What exactly is a RAM disk? I know that RAM means Random Access Memory, and that you create a RAM disk, but where is it physically put and how do you create one?
REPLY: JOHN WHELAN
All storage in computers is a series of 0's and 1's, whether the medium is tape, WORM, RAM, Drums, hard or floppy disks. Different mediums have different characteristics. The major differences are access speed, cost, whether you can reuse the medium, and whether the medium loses its information when the power is turned off. Paper tape, for example, can be used to store information used in a program, i.e. variables. When all you have is 1K of RAM, then writing info to paper tape and re-reading it works. If the power fails, no big deal. All the information is saved. Punched cards can also be used to retain information when the power goes off. WORMs, floppy and hard disks, tapes are usually considered to be examples of non-volatile memory. RAM, whether it be valves or semi-conductors, usually is considered to be volatile. It loses all stored information when the power is turned off. Tapes have the largest access speed. You have to get to the correct bit of tape before reading in the information.

Floppy and hard disks are slower, but because you can jump about on the disk without reading in everything from the beginning, the random access speed decreases from minutes to seconds or even less. A mecanical read head has to move around, so this is the limiting factor here. RAM memory now has moving parts. The access method is ramdom, and the speed starts to pick up, typically where the fastest disk is 8 ms (assuming the head is on the correct track already). If it isn't, start to add 15 to 120 ms.

QUERY: BRICE WIGHTMAN
I'm OK on deleting words, lines, etc. The function I discovered is the ability to delete a continuous block of spaces between words. I'm certain I stumbled on the key combination once, but I can't find it documented. Now it would come in handy to doctor up a big database. Do you know how to delete a block (string) of all blanks?
REPLY: BRICE WIGHTMAN
I found the two-key combination that deletes a string of blanks. It's Ctl-DEL, the same as delete word, but there is a catch. The cursor must not be next to a character. Position the cursor anywhere along the string of blanks. This will delete all blanks up to the next non-blank character.

QUERY: BRICE WIGHTMAN
I am looking for a database package I like. I have downloaded File Express and PC-File. I would like to read the documentation. However, I am not about to print up to 150 pages just for a look. On the other hand, the DOC files are not ASCII and unreadable. The packages supply an .EXE to print the DOC file, and I have tried to pipe the output to another file, hoping to get readable ASCII. No go. I get "Out of paper" or whatever messages. Any ideas anyone?
REPLY: CHRIS TAYLOR
I use PRN2FILE. It redirects printer output to a file. It should work fine. You can find it in the DOS-Utility file section.

OTTAWA PC USERS' GROUP

EXECUTIVE		
President	David Terroux	238-4895
Past President	Stu Moxley	592-4933
Treasurer	Tony Frith	671-0401 Weekends
Secretary	Claude Jarry	521-3366
Facilities Officer	Douglas Poulter	745-8768
Membership Chairman	Carl-Henri Gomez	731-1462
Publicity	André Cyr	561-5207
Software Librarian	Chris Taylor	723-1329
Newsletter Editor	Bonnie Carter	236-1015
BBS System Operator	Mike Schupan	820-0293
BULK PURCHASING	Terry Mahoney	225-2630 226-2615 FAX
ASSISTANTS		
Software Assistant	John Ings	235-8132
Newsletter Assistants	Marc Riou	733-2092
Contributing Newsletter Editors	Chris Taylor	723-1329
	Ann Falkner	722-3226
	David Smith	837-9291
BEGINNERS' SESSIONS	Eric Clyde	749-2387

DISK OF THE MONTH

One-year subscription	(10 DOMs)	5 1/4"	$25.00
One-year subscription	(10 DOMs)	3 1/2"	$35.00
Individual 3 1/2" disk surcharge			$2.00

THE OTTAWA PC USERS' GROUP MEMBERSHIP APPLICATION - Please Print

Last Name: ______________________ First Name: ______________________

Mailing Address: ____

Postal Code: ______________ Telephone - Home: __________ Office: ______________

Profession: ______________ Date of Birth: __________ Sex: M ________ F ________

Disk of the Month: YES _____ NO _____ Size: 5 1/4" _____ 3 1/2" _____ Amount Enclosed $ ____________

Are you: A new member? _______ Renewing your membership? _______ Sponsor's Name: ______________

What in particular interests you in the Group? ______________________________

Can you help in Group activities? Check off the activities that apply: Programming Language Instruction __________

Newsletter Input __________ Memberships __________ Software Library ________ Promotion/Publicity __________

Hardware Techniques _________ Meeting Locations ________ Agendas & Speakers ________ Advertising __________

Bulletin Board__________ Other______________________________________

What hardware/software do you own and/or use? ______________________________