==Phrack Inc.==

Volume Two, Issue Nineteen, Phile #1 of 8

Index

Welcome to Phrack Issue Nineteen! You will notice it is not as long as the last Phrack but this is the month of SummerCon and plans have been made for that. If you are interested just check PWN for details. Also, we do need writers, so if you have a phile or know someone who does, please get in contact with me. The next issue of Phrack will be full size again, but since it is summer we all slowed down a bit. Don't worry though, Phrack will still come out every month. Well, see you at SummerCon!

Crimson Death
Sysop of The Forgotten Realm

Contents:

==========

==Phrack Inc.==

Volume Two, Issue 19, Phile #2 of 8

DCL Utilities for the VMS Hacker

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

By
The Mentor

Special thanks to Silver Spy for
turning me onto DCL in the first place!

Anyone who spends time hacking on VAXes (by hacking, I don't just mean trying to get in... I mean *doing* something once you're in!) notices that the DCL command language is extremely powerful. I have put together a selection of utilities that not only should prove helpful to the hacker, but serve as a good example of programming in DCL.

Every attempt has been made to preserve unchanged the user-environment from the initialization of the file to the exit. Any session-permanent changes are documented.

Brief Overview of DCL

>>>>>>>>>>>>>>>>>>>>

There are numerous files out there on DCL (the VMS help files are the best place to find information), so I'm not going to teach you how to program in it. To use the following code, isolate the section of code you want in your favorite text editor, upload it into a file, and name the file <progname>.COM. Anytime you see a file ending with .COM, you know it's a DCL file. DCL files are executed by issuing the command

```
$@FILENAME
```

or, in the case of a file you want to run as a separate process,

```
$SPAWN/NOWAIT @FILENAME
```

Table of Contents

>>>>>>>>>>>>>>>>

I have about 300 more pages of COM files. If you need anything, drop me a line. I'll try and help out. I can be found on Forgotten Realm, or you can call a non-hacker (local to me) IBM game board if it's an urgent message (The Bastille-- 512/353-0590 300/1200 24 hrs. It's not the best hacker board in the world, but my mail arrives daily...)

Also, if programming of this type interests you, let me know! I'm considering putting up a board for the discussion of programming (compilers, AI/Expert Systems, Op Systems, etc...). If I get enough positive response, I'll go with it. Leave mail on the aforementioned systems.

CD.COM Version 5.0 VMS Change Directory Command

Sub-directories are a nice feature on many computers, but they're not always easy to take advantage of. The VMS commands to access sub-directories are a little obscure, even to PC programmers who are used to using directories.

The solution? CD.COM, a change directory command that works almost the same as the PC-DOS CD and PROMPT commands:

CD	- Display your home directory, current directory, and node name. (Similar to, but better than the VMS SHOW DEFAULT command.)
CD dir_name CD [dir_name]	- Move you to the [dir_name] directory. (Same as the SET DEFAULT [dir_name] command.)
CD .sub_name CD [.sub_name]	- Move you to the [.sub_name] subdirectory. (Same as the SET DEFAULT [.sub_name] command.)
CD \ CD HOME CD SYS$LOGIN	- Move you to your home (root) directory, which is the directory you are in when you login. (Same as the SET DEFAULT SYS$LOGIN command.)
CD .. CD [-]	- Move you to the directory above your current directory. (Same as the VMS SET DEFAULT [-] command.)
CD ..sub_name CD [-.sub_name]	- Move you "sideways" from one subdirectory to another subdirectory. (Same as the SET DEFAULT [-.sub_name] command.)
CD *	- Select a subdirectory to move to, from a list of subdirectories.
CD .	- Reset the current directory.
CD ?	- Display instructions for using CD.

The VMS SET DEFAULT command has a flaw: you can change directories to a directory that doesn't exist. CD handles this more elegantly; you're left in the same directory you were in before, and this message appears:

```
[dir_name] Directory does not exist!
```

PC-DOS lets you display the current directory as part of the prompt. (If you haven't seen this feature, try the PC-DOS command PROMPT PG.) CD.COM will change the prompt for you each time you change directories if you include this line in your LOGIN.COM file:

```
DEFINE SYS$PROMPT "ON"
```

Without this line, your prompt is not changed from what you have it set as. Instead, your home (root) directory name, current directory name, and node name are displayed whenever you issue the CD command.

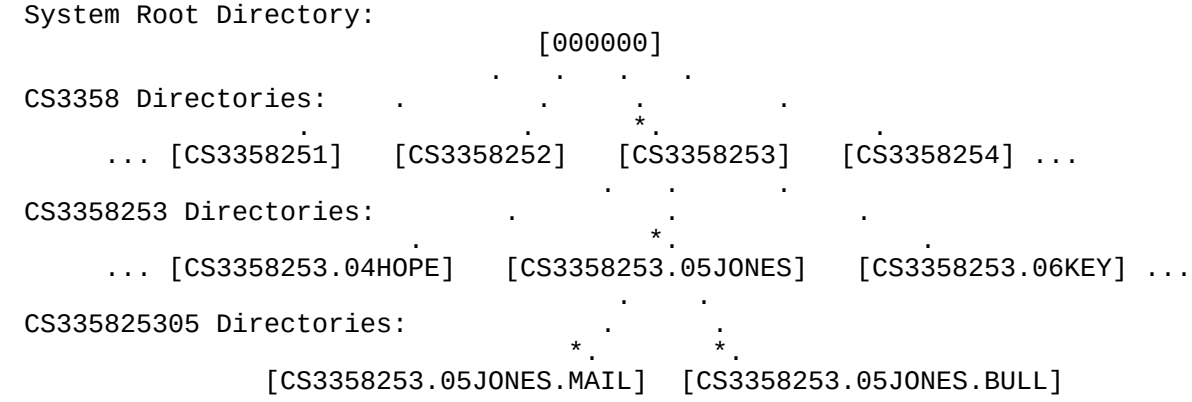
Since VMS allows prompts to contain no more than 32 characters, if you change to a subdirectory that would make your prompt too long, CD automatically leaves off some of the higher level sub-directories to keep your prompt short, and displays a "*" as one of the prompt characters.

CD lets you use directory names defined with with the DEFINE command. For example, if you're in one of Dr. Smiths' CS3358 classes, you might want to define his CS3358 assignments directory like this:

```
DEFINE SMITH "DISK$CS:[CS.SMITH.3358]"
```

Then, CD SMITH would move you to this directory. Try it! Also, some directories are already defined by the system. The SHOW LOGICAL command will give you clues to some of these system directories, if you want to go exploring. CD also supports the use of symbols for directory names.

Like with PC-DOS, VMS directories and sub-directories are tree structured. The system root directory for your disk has the name [000000], and in it are the names of all the sub-directories for your disk. The directories for an imaginary user, CS335825305, would be located like this:



If you're not using sub-directories, but want to, you can create them with the CREATE command:

```
CREATE/DIR  [.sub_name]
```

VMS allows directories to be seven or eight levels deep, but one or two levels is enough for most users.

VMS also allows the symbols < and > to be used instead of [and], to specify directory names. CD fully supports this.

Code for CD.COM
>>>>>>>>>>>>>>>

```
$! CD.COM v6.09
```

```
$! The Ultimate Change Directory Command.
$!
$  hdir     = f$trnlnm("SYS$LOGIN")                 ! Home Directory
$  ndir     = f$edit(p1,"UPCASE")                   ! New  Directory
$  odir     = f$environment("DEFAULT")              ! Old  Directory
$  prompton = (f$edit(f$trnlnm("SYS$PROMPT"),"UPCASE") .eqs. "ON")
$!
$  if (ndir .eqs. "")           then goto DISPLAY   ! No Dir
$  if (ndir .eqs. "*")          then goto DIRSEARCH ! Search for Dirs
$  if (ndir .eqs. "?")          then goto HELP      ! Instructions
$!
$  PARSE:
$  length   = f$length(ndir)                        ! Fix up ndir
$  if (f$location("@",ndir) .eq. 0) .or. -
      (f$location("$",ndir) .eq. 0) then ndir = f$extract(1, length - 1, ndir)
$  right    = f$location("]",ndir) + 1
$  if (right .gt. length) then right = f$location(">", ndir)
$  if (right .le. length) then ndir  = f$extract(0, right, ndir)
$!
$  if (f$trnlnm(ndir) .eqs. "") then goto CASESYM   ! Not Logical Name
$     ndir   = f$trnlnm(ndir)                       ! Logical Name
$     goto PARSE
$!
$  CASESYM:
$  if ("''&ndir'" .eqs. "")     then goto CASE0     ! Not Symbol
$     ndir = 'ndir'                                 ! Symbol
$     goto PARSE
$!
$  CASE0:
$  len_ndir = f$length(ndir)                        ! Regular Dir
$  if (f$location("[", ndir) .lt. len_ndir) .or. -
      (f$location("<", ndir) .lt. len_ndir) then goto SETDIR
$!
$  CASE1:                                           ! Home Dir
$  if ((ndir .nes. "HOME") .and. (ndir .nes. "\")) then goto CASE2
$     ndir = hdir
$     goto SETDIR
$!
$  CASE2:                                           ! . .. .dir
$  if (f$location(".", ndir) .nes. 0) then goto CASE3
$     if (ndir .eqs. "..") then ndir = "-"
$     if (f$extract(0, 2, ndir) .eqs. "..") -
         then ndir = "-" + f$extract(1, len_ndir - 1, ndir)
$     ndir = "[" + ndir + "]"
$     if (ndir .eqs. "[.]") then ndir = odir
$     goto SETDIR
$!
$  CASE3:                                           ! :
$  if (f$location(":", ndir) .ge. len_ndir) then goto CASE4
$     left    = f$location(":", ndir) + 1
$     symbol  = f$extract(left, 1, ndir)
$     if (symbol .eqs. ":")  then goto CASE3B       ! :: Node
$     if ((symbol .eqs. "[") .or. (symbol .eqs. "<")) then goto SETDIR
$        ndir = f$extract(0, left, ndir) + "[" -
             + f$extract(left, len_ndir - left+1, ndir) + "]"
$     goto SETDIR
$!
$  CASE3B:                                          ! NODE::nothing
$  if (f$length(ndir)-1 .gt. left) then goto CASE3C
$     ndir = ndir + "[000000]"
$     goto SETDIR
```

```
$!
$  CASE3C:                                              ! NODE::directory
$  if ((f$location("[", ndir) - f$location("<", ndir)) .ne. 0) -
      then goto SETDIR
$
$     ndir = f$parse(ndir,,,"NODE") + "[" + f$parse(ndir,,,"NAME") + "]"
$     goto SETDIR
$!
$  CASE4:                                               ! dir
$  ndir = "[" + ndir + "]"
$!
$  SETDIR:
$  set default 'ndir'
$  if (f$parse("") .eqs. "") then goto DIRERROR
$!
$  DISPLAY:
$  if ((ndir .nes. "") .and. prompton) then goto NODISPLAY
$     hnode = f$getsyi("NODENAME")
$     cnode = f$parse(f$trnlnm("SYS$DISK"),,,"NODE") - "::"
$     if (cnode .eqs. "") then cnode = hnode
$     cdir  = f$environment("DEFAULT")
$     write sys$output " "
$     write sys$output "          Home Node: ", hnode
$     write sys$output "     Home Directory: ", hdir
$     if (cdir .eqs. hdir) .and. (cnode .eqs. hnode) then goto DISPSKIP
$     write sys$output "       Current Node: ", cnode
$     write sys$output "  Current Directory: ", cdir
$  DISPSKIP:
$     write sys$output " "
$!
$  NODISPLAY:
$  ndir = f$environment("DEFAULT")
$  if .not. prompton then goto END
$!
$  if (f$length(ndir) .ge. 32) then goto TOOLONG
$!
$  SETPROMPT:
$  set prompt = 'ndir'" "
$!
$  END:
$  exit
$!
$  DIRERROR:
$  write sys$output " "
$  write sys$output "         ", ndir, " Directory does not exist!"
$  write sys$output " "
$  set default 'odir'
$  ndir = odir
$  goto NODISPLAY
$!
$! Prompt Problems-----------------------------------------------------------
$!
$  TOOLONG:
$! Prompt is too long. Get rid of everything to the left of [ or <. If that
$! doesn't work, get rid of a subdirectory at a time.  As a last resort,
$! set the prompt back to $.
$!
$  left     = f$location("[", ndir)
$  len_ndir = f$length(ndir)
$  if (left .ge. len_ndir) then left = f$location("<",ndir)
$  if (left .gt. 0) .and. (left .lt. len_ndir) -
```

```
      then ndir = f$extract(left, len_ndir - left, ndir)
$!
$  STILLTOOLONG:
$    if (f$length(ndir) .lt. 32) then goto SETPROMPT
$    left     = f$location(".", ndir) + 1
$    len_ndir = f$length(ndir)
$    if left .ge. len_ndir then ndir = "$ "
$    if left .ne. len_ndir -
        then ndir = "[*" + f$extract(left, len_ndir - left, ndir)
$    goto STILLTOOLONG
$!
$! Wildcard Directory-----------------------------------------------------------
$!
$  DIRSEARCH:
$  error_message = f$environment("MESSAGE")
$  on control_y then goto DIREND
$  on control_c then goto DIREND
$  set message/nosev/nofac/noid/notext
$  write sys$output " "
$  dispct = 1
$  dirct  = 0
$  pauseflag = 1
$!
$  DIRLOOP:
$    userfile = f$search("*.dir")
$    if (userfile .eqs. "") .and. (dirct .ne. 0) then goto DIRMENU
$    if (userfile .eqs. "") then goto DIRNONE
$    dispct = dispct + 1
$    dirct  = dirct  + 1
$    on severe then $ userprot = "No Priv"
$    userprot = f$file_attributes(userfile,"PRO")
$    if userprot .nes. "No Priv" then userprot = " "
$    userfile'dirct' = "[." + f$parse(userfile,,,"NAME") + "]"
$    userprot'dirct' = userprot
$    lengthflag = (f$length(userfile'dirct') .gt. 18)
$    if lengthflag then write sys$output -
        f$fao("  !3SL   !34AS  ", dirct, userfile'dirct'), userprot'dirct'
$    if (.not. lengthflag) then write sys$output -
        f$fao("  !3SL   !20AS  ", dirct, userfile'dirct'), userprot'dirct'
$    if (dispct .lt. 8) then goto DIRLOOP
$    dirct  = dirct  + 1
$    userfile'dirct' = ""
$    dirct  = dirct  + 1
$    userfile'dirct' = ""
$    if pauseflag then goto DIRMENU
$    dispct = 0
$    goto DIRLOOP
$!
$  DIRMENU:
$  write sys$output " "
$  if (userfile .eqs. "") then goto DIRMENU2
$     write sys$output "    M   More subdirectories"
$  if pauseflag then -
$     write sys$output "    N   More subdirectories/No pause"
$!
$  DIRMENU2:
$     write sys$output "    R   Re-Display subdirectories"
$     write sys$output "    Q   Quit (default)"
$
$  DIRINQUIRE:
$  write sys$output " "
```

```
$  inquire dirchoice "  Select One"
$  write sys$output " "
$!
$  if (dirchoice .gt. 0)    .and. -
      (dirchoice .le. dirct) then goto DIRCASEDIGIT
$  dirchoice = f$edit(dirchoice,"UPCASE")
$  if (dirchoice .eqs. "")  .or. -
      (dirchoice .eqs. "Q")  then goto DIRCASEBLANK
$  if (dirchoice .eqs. "M") .or. -
      (dirchoice .eqs. "N")  then goto DIRCASEMORE
$  if (dirchoice .eqs. "R")  then goto DIRCASERED
$!
$  DIRCASERROR:
$  if (dirct .eq. 1)   then write sys$output -
      "  Select 1 to change to the ", userfile1, " subdirectory. "
$  revdirct = dirct
$  if (dispct .eq. 8) then revdirct = revdirct - 2
$  if (dirct .gt. 1)   then write sys$output -
      "  Valid subdirectory selections are 1 through ", revdirct, " (Octal)."
$  goto DIRINQUIRE
$!
$  DIRCASEDIGIT:
$  if (userfile'dirchoice' .eqs. "") then goto DIRCASERROR
$  ndir = userfile'dirchoice'
$  goto DIREND
$!
$  DIRCASEBLANK:
$  write sys$output "  Subdirectory not changed."
$  write sys$output " "
$  goto DIREND
$!
$  DIRCASEMORE:
$  dispct = 0
$  if (dirchoice .eqs. "N") then pauseflag = 0
$  if (userfile .nes. "")   then goto DIRLOOP
$  write sys$output "  No more subdirectories to display."
$  goto DIRINQUIRE
$!
$  DIRCASERED:
$  dispct = 1
$  DISPLOOP:
$     if (userfile'dispct' .eqs "") then goto DISPDONT
$     lengthflag = (f$length(userfile'dispct') .gt. 18)
$     if lengthflag then write sys$output -
         f$fao("  !3SL   !34AS  ", dispct, userfile'dispct'), userprot'dispct'
$     if (.not. lengthflag) then write sys$output -
         f$fao("  !3SL   !20AS  ", dispct, userfile'dispct'), userprot'dispct'
$     DISPDONT:
$     dispct = dispct + 1
$     if (dispct .le. dirct) then goto DISPLOOP
$  goto DIRMENU
$!
$  DIRNONE:
$  write sys$output "No subdirectories to choose, or no directory privileges."
$  write sys$output " "
$  goto DIREND
$!
$  DIREND:
$  set message 'error_message'
$  on control_y then exit
$  on control_c then exit
```

```
$  if (ndir .eqs. "*") then goto DISPLAY
$  goto PARSE
$!
$!-Help-----------------------------------------------------------------------
$!
$  HELP:
$  type sys$input

               CD.COM  Version 6  VMS Change Directory Command

                         Usage:  CD command/directory

CD         Display home directory,       CD ..       Change directory to the
           current directory, node.      CD [-]      dir above current dir.

CD \       Change directory to your      CD ..sub    Change directory to a
CD HOME    SYS$LOGIN directory.          CD [-.sub]  "sideways" subdirectory.

CD dir     Change directory to the       CD *        Display/select the
CD [dir]   [dir] directory.                          available subdirectories.

CD .sub    Change directory to the       CD .        Reset current directory.
CD [.sub]  [.sub] subdirectory.          CD ?        Display CD instructions.

     CD :== @SYS$LOGIN:CD.COM                DEFINE SYS$PROMPT "ON"
     To make CD available from               To have the VMS $ prompt
     any directory you change to.            display the current directory.

                                By The Mentor
$  goto END


                                Code for HUNT.COM
                                >>>>>>>>>>>>>>>>>


$ ! HUNT.COM
$ ! By The Mentor
$ ! Updated by: The Mad Mexican
$ ! Usage: SPAWN/NOWAIT @HUNT
$ !
$ !Searches SHOW USER output for a specified user,  strobes at given
$ !intervals considering the severity of the hunt at which time output
$ !is generated and process terminates. If user loggs in then output
$ !is generated and process terminates. May check both nodes if a set
$ !host is called.  Also supports a file with the names to be hunted for.
$ !
$ !  *** NOTE ***   This is set up for a two-node system with NYSSA
$ !                 being the default node and TEGAN being the alternate
$ !                 node (Circuit Breaker and some others will recognize
$ !                 the nodes as my 'home' ones.)  You will need to
$ !                 slightly modify the code to reflect the nodename(s)
$ !                 of whatever system you are using...
$ !
$ !
$ !
$ say="write sys$output"
$ on control then goto door
$ monitored_node = "''NODE'"
$ say "Monitoring node ''monitored_node'.  <HIT RETURN>"
$ severity_of_hunt:
```

```
$ inquire selection "Severity of HUNT, 1 being the most urgent: 1-2-3"
$ if selection.ge.2 then goto selection_2
$ delay="wait 00:00:20"
$ loop_count=40
$ goto begin_process
$ selection_2:
$ if selection.eq.3 then goto selection_3
$ delay="wait 00:01:00"
$ loop_count=8
$ goto begin_process
$ if selection.gt.3 then goto severity_of_hunt
$ delay="wait 00:02:30"
$ loop_count=20
$ begin_process:
$ if monitored_node.eqs."TEGAN" then goto search_file_tegan
$ if f$search("nyssa.dat9") .nes. "" then goto file_exist
$ goto continue
$ search_file_tegan:
$ if f$search("tegan.dat9") .nes. "" then goto file_exist
$ continue:
$ say "hit <RETURN>"
$ inquire/nopunctuate choice9 "Who are we hunting for? "
$ if choice9 .eqs. "" then exit
$ count = 0
$ bell_sound[0,8]=%X07
$ top:
$ sho user/output='monitored_node'.dat9
$ purge 'monitored_node'.dat9
$ set message/nofac/noid/notext/nosev
$ search 'monitored_node'.dat9 'choice9'
$ a=$severity
$ if a .eqs. "1" then goto found_user
$ set message 'temp_msg9'
$ count = count + 1
$ if count .ge. 'loop_count' then goto give_up
$ delay
$ goto top
$ file_exist:
$ say "ERROR - Could not create temporary data file."
$ say "Please delete or rename ''NODE'.DAT9"
$ exit
$ found_user:
$ say bell_sound
$ say "''choice9' is now online on node ''monitored_node'."
$ say bell_sound
$ goto door
$ give_up:
$ say " "
$ say "''choice9' has not yet logged in on ''monitored_node'."
$ door:
$ say bell_sound
$ say "HUNT routine has terminated on node ''monitored_node'."
$ delete/noconfirm/nolog 'monitored_node'.dat9;*
$ set message 'temp_msg9'
$ exit
```

Code for ALARM.COM
>>>>>>>>>>>>>>>>>>

```
$ ! ALARM.COM
$ ! By The Mentor
```

```
$ ! Usage: SPAWN/NOWAIT @ALARM
$ ! Strobes f$time() every 5 seconds until specified time
$ ! is met at which time output is generated and process terminates.
$ CLR = " "
$ count = 0
$ PID           = F$PID(CONTEXT)
$ TERMINAL      = F$GETJPI(''PID',"TERMINAL")
$ DEVICE        = F$GETDVI(TERMINAL,"DEVTYPE")
$ IF DEVICE .EQS. 110 THEN CLR = "[H[2J"  ! VT220
$ IF DEVICE .EQS.  98 THEN CLR = "[H[2J"  ! VT102
$ IF DEVICE .EQS.  96 THEN CLR = "[H[2J"  ! VT100
$ IF DEVICE .EQS.  64 THEN CLR = "HJ"     ! VT52
$ CLS = "WRITE SYS$OUTPUT CLR"
$ DATE       = F$CVTIME(F$TIME())
$ NODE       = F$GETSYI("NODENAME")
$ bell[0,8]=%X07
$ ON CONTROL THEN GOTO DOOR
$ say = "write sys$output"
$ say f$cvtime(,,"TIME")
$ say " "
$ say "Hit (RETURN)"
$ say " "
$ inquire/nopunctuate alarm "What time shall I ring you - "
$ a_hour = f$element(0,":",alarm)
$ a_minute = f$element(1,":",alarm)
$ a_second = f$element(2,":",alarm)
$ time_check:
$ hour = f$element(0,":",f$cvtime(,,"TIME"))
$ minute = f$element(1,":",f$cvtime(,,"TIME"))
$ second = f$element(2,":",f$element(0,".",f$cvtime(,,"TIME")))
$ if hour .ge. a_hour .and. minute .ge. a_minute .and. second .ge.
  a_second then goto top
$ if hour .ge. a_hour .and. minute .ge. a_minute then goto top
$ wait 00:00:05
$ goto time_check
$ top:
$ count = count + 1
$ cls
$ say " "
$ say " "
$ say " "
$ say " "
$ say " "
$ say " "
$ say " "
$ say " "
$ say " "
$ say " "
$ say " "
$ say "                                A L A R M   O N"
$ say bell
$ say "                                   ",f$element(0,".",f$cvtime(,,"TIME"))
$ say " "
$ say " "
$ say " "
$ say " "
$ say " "
$ say " "
$ say " "
$ say " "
$ say " "
```

```
$ wait 00:00:01.50
$ if count .le. "6" then goto top
$ door:
$ say "ALARM OFF"
$ say f$element(0,".",f$cvtime(,,"TIME"))
$ say bell
$ exit
```

Code for CGO.COM
>>>>>>>>>>>>>>>>

```
$! CGO.COM
$! By The Mentor
$! One-Line compile/link/execute of C programs
$! Usage: CGO :== @CGO.COM
$!        CGO filename
$!
$if p1 .nes. "" then c_filename :== 'p1
$ write sys$output "Compiling:"
$ cc 'c_filename/list='c_filename.lst
$ write sys$output "Linking:"
$ link 'c_filename ,options_file/opt
$ write sys$output "Running:"
$ assign/user sys$command sys$input
$ run 'c_filename
$ exit
```

Well, that's it. I hope to be back in the next issue with some other programs. And remember, any programmers out there, get in touch with me!

The Mentor
Thanksgiving 1987

==========

==Phrack Inc.==

Volume Two, Issue 19, Phile #3 of 8

Understanding the Digital Multiplexing System (Part2)

by

Control C

&

The Tribunal Of Knowledge

Well some of you may recall my file on Digital Multiplexing in Phrack 10. Well this is part 2 that was promised about a year and a half ago. I was finished with this file in May of 87 and I just decided to release it now. Here it is!

DMS switches were first introduced in 1979, since then it has been modified to interface numerous types of switches. DMS has the ability to interface with SP-1, #5 XBar, 1ESS, 2ESS, 3ESS, 4ESS, NX1D, NX1E, TSD, SXS, ETS4, NO. 1 EAC, NO. 2 EAX, NO. 3 EAX, TSPS, CAMA/3CL boards, Stromberg Carlson Turret of ONI and Visual Indicators, Modified North Electric TSD for ONI, Stomberg Carlson (CAMA operator Position - ONI/ANI), AE #31 Switchboard, Co-located NT/AE switchboard I/C, O/G, UDC data poller of OM, DACS (Directory Assistance Charging System), NT #144 LTD, WECO #14 LTD, WECO #16 LTD, CALRS (Centralized Automated Loop Reporting System), Badger 612A, AE #1 and #21 LTD, AE #30, SC #14 LTD, Lordel MITS70 line Test System, Porta System Line Test Unit, Pulsar II IMTS, Teradyne loop test unit, and the WECO MLT 1 (Mechanized Loop Testing System).

Common Channel Interoffice Signaling

Common Channel Interoffice Signaling (CCIS) is a way of signaling and a way of implementing network level services. CCIS provides reliable, crystal clear data signaling links between the network and the switching offices. The CCIS signaling method uses transmission equipment that is separate from voice trunks.

Common Channel Interoffice Signaling No. 6

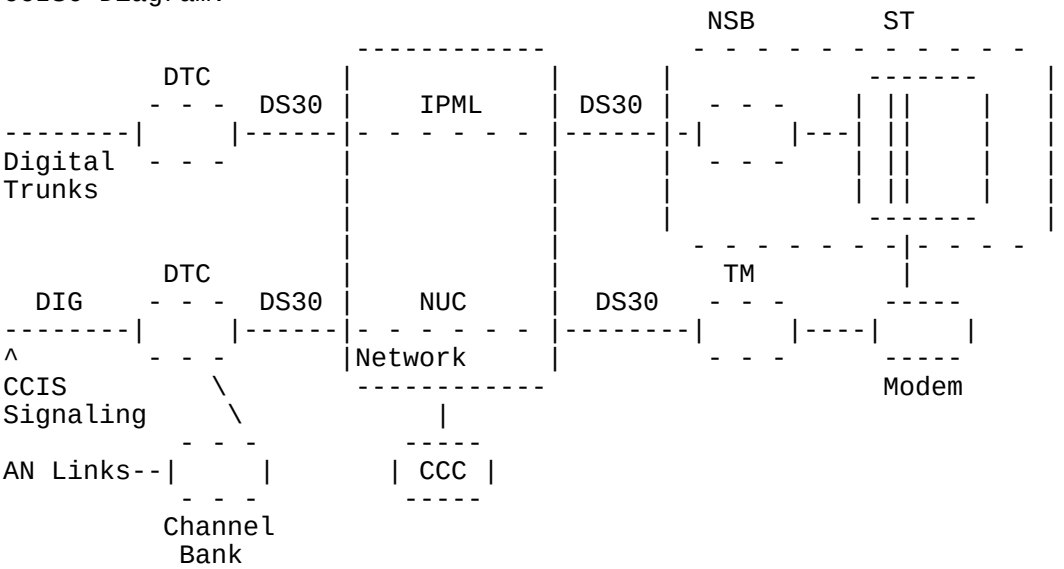
The basis for the CCIS system is the International Consultative Committee on Telephone and Telegraph (CCITT) No. 6 international standard, which is brought to it's fullest capacity for use in the Stored Program Control (SPC) network of AT&T.

The CCIS6 network contains a bunch of signaling regions, each having a pair of interconnected Signal Transfer Points (STP). The switching systems put into CCIS6 then connecting to STPs are called Serving Offices (SO).

Band Signaling (CCIS-BS) is used on trunk signaling for intertoll-type trunks using the CCIS network.

Direct Signaling (CCIS-DS) is used for signaling between SPC switching machines and a Network Control Point (NCP). At the present time CCIS6 can handle Enhanced INWATS Originating Screening Office (OSO), Calling Card Validation (CCV), Mechanized Calling Card Service (MCCS), and Billed Number Screening (BNS). CCIS6 is available with DMS-100/200, DMS-200, and DMS-100/200 or DMS-200 with TOPS.

CCIS6 Diagram:



Acronyms:

DIG - Digital
AN - Analog
DTC - Digital Trunk Controller
MSB - Message Switch Buffer
ST - Signaling Terminal
TM - Trunk Module
NUC - Nailed-Up Connection
IPML - Inter-Peripheral Message Link

Common Channel Interoffice Signaling No. 7

Common Channel Signaling (CCS) No. 7 or CCIS7 is a CCS system bases on CCITT No. 7. CCIS7/CCS7 on the DMS switch consists of two parts the Message Transfer Part (MTP) and the Interim Telephone user Part. They are compatible with DMS-100, DMS-200, DMS-100/200, and DMS-100/DMS-100/200 with TOPS.

CCIS7 can't tell the difference between banded and direct signaling. CCIS7 uses Destination/Origination Point Codes (DPC/OPC) to rout back to the switch.

CCIS7 can handle Automatic Calling Card Service (ACCS), Enhanced INWATS, Local Area Signaling Services, and Direct Service Dialing Capabilities.

Equal Access

The DMS-200 Access Tandem (AT) gives a traffic concentration and distribution function for interLATA traffic originating and a distribution function for interLATA traffic origination or terminating inside a Local Access and Transport Area (LATA). This gives the interLATA Carrier (IC) access

to more that one end office inside the LATA. It can handle InterLata Carrier access codes (10xxx), 10xxx and 950-yxxx dialing, Automatic Number Identification (ANI) on all calls, answer supervision, equal access Automatic Message Accounting (AMA) for both originating and terminating calls, and operator service signaling.

The DMS-100 EA gives direct and tandem switched access service inside the LATA for originating and terminating to interLATA Carriers. It is available in the following three ways:

Equal Access End Office (EAEO)

DMS-100 Equal Access End Office (EAEO) gives a direct interconnection to interLATA Carriers (IC) and international Carriers (INCs) Point of Presence (POP) inside the LATA.

Access Tandem with Equal Access End Office

The DMS-200 Access Tandem (AT) when used with equal access end office (EAEO) lets trunk tandem interconnect to ICs/INCs POP inside the LATA.

The connection of the Equal Access End Office (EAEO) to an IC/INC through the DMS-200 Access Tandem (AT) uses what is called two-stage overlap output pulsing which makes the time it takes to set up a call quicker. The AT uses the digits OZZ + XXX out pulsed in the first stage to identify the IC/INC dialed and to pick and outgoing trunk. Then a connection is established from the IC/INC to the EAEO through the AT. The second stage digits, consist of ANI and the called numbers are passed through the DMS- 200 AT at the IC/INC.

A AMA terminating record in AT&T format is produced by the DMS-200 for all the EAEOs. A per call terminating AMA record is made for calls that get to the stage where the trunk from the IC/INC has been seized and a "wink" has been returned by the DMS-200 AT.

Access Tandem with a Non-Equal Access End Office

DMS-200 AT using a non-equal access end office gives trunk tandem connection to an IC/INC POP within the LATA. To set up a call, connection of Feature Group B (FGB) or Feature Group C (FGC) End Office to an IC/INC through the DMS-200 AT, uses the standard Bell Central Automatic Message Accounting (CAMA) signaling. The Access Tandem uses the XXX digits of the access code 950-YXXX out pulsed from the FGB end office to identify the IC/INC and to connect to a outgoing trunk.

Mechanized Calling Card Service (MCCS)

The fraudulent use of calling cards, third number and collect calls and the increasing movement to automate current operator services has directly led to the implantation of the Mechanized Calling Card Service (MCCS) to DMS-200/TOPS and to the remote and host Operator Centralization (OC).

MCCS uses CCIS to relay queries and responses to and from the DMS-200/TOPS. Operator handled calling card calls and the direct entry by subscribers of Calling Cards by DTMF (Touch-Tones) telephones are given special provisions by the MCCS. Both, the operator handling and the direct entry of calling card calls, are decreasing the size operators.

Billed Number Screening (BNS) gives an enhancement to the operator-handled collect and third-number billing by using CCIS to screen a number at the billing validation data base for billing restrictions (i.e. the third number is a fortress). This feature naturally will reduce fraudulent use of the

collect call feature.

Common Channel Interoffice Signalling-Direct Signalling (CCIS-DS), which is the feature that the MCCS is designed around, is used to transmit messages to and from many possible Billing Validation Centers (BVCs). Messages transmitted to the BVC about MCCS include the billing number and the Personal Identification Number (PIN). In BNS the messages have the special billing number (collect or third number). The return messages from the BVC include validity (of the number), billing restrictions (if any), and the Revenue Accounting Office (RAO) code.

Auxiliary Operator Services System

The DMS-200 Auxiliary Operator Services System (AOSS) is used primarily for Directory Assistance and the intercept needs that are not included in the TOPS package. The AOSS is similar to TOPS and co-exist with TOPS on the DMS-200 Toll system.

Major benefits of the AOSS include Directory Assistance is provided with a modern environment, AOSS position administrative activities are performed by the DMS-200 toll maintenance system, trunking savings are achieved by combining trunking for 1+ and 0+, and Directory Assistance traffic, DA services are managed by using TOPS methods, Creation of a built-in training system, which does not require additional training equipment and reduces training costs.

Integrated Business Network

The Integrated Business Network (IBN) is a revenue-producing concept designed for small and big businesses to offer modernized PBX and Centrex features. The Operating Company can use the IBN to maintain and enhance its competitive position on a operational DMS-100 and DMS 100/200 switches. While using the DMS-100 switch, the Operating Company can support varying business features along with existing local/toll traffic.

IBN services can be introduced to a Centrex-Central Office (CO) or a Centrex-Customer Unit (CCU) by additional software modules and minor hardware enhancements.

Current IBN features include: A growing system that can handle 30,000 lines, networking capabilities, city wide service for DMS- 100 switch and remotes for any one customer station Message Detail Recording (SMDR), which gives IBN customers call records. The records can be used for system analysis and control and station charge-back. SMDR can use LAMA records, if the IBN host has LAMA equipment, Centralized attendant maintenance and administration functions and Direct Inward Dialing (DID).

Electronic Switched Network (ESN)

The Electronic Switched Network is designed to meet the telecommunication needs of large multi-location corporations. The ESN is made up of a SL-1 or SL-100 Digital Business Communications System with networking features or a DMS-100 IBN host. The SL-1 can handle from 30-5000 lines. The SL-100 and the DMS-100 IBN hosts can hold from a few thousands to 30,000 lines.

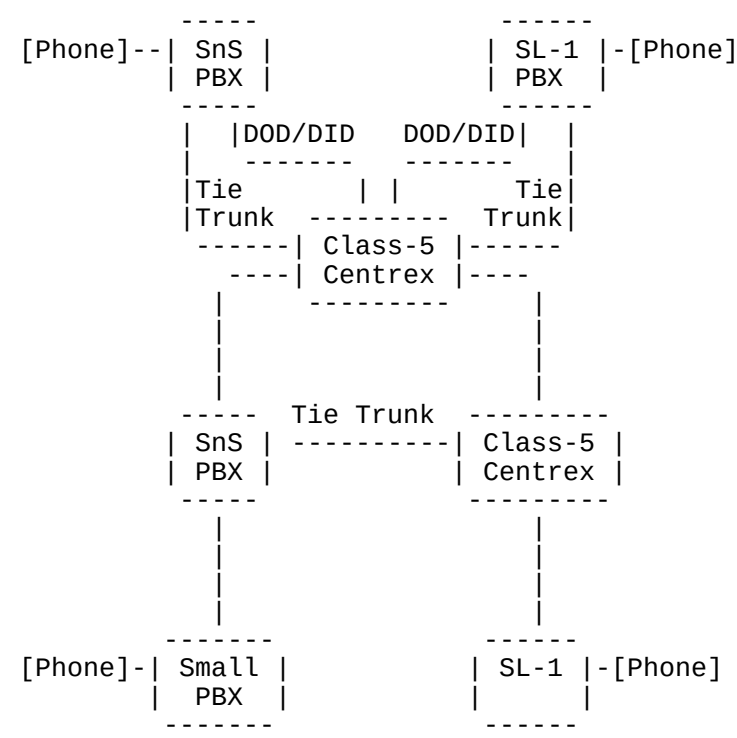
A DMS-100 IBN or SL-100 can remotely serve many locations from the host site. This is done by a connection through digital transmission facilities which are set up at remote modules at the subscriber's premises.

Specialized Common Carrier Service (SCCS)

The DMS-250 Specialized Common Carrier Service (SCCS) provides the capability of Analog to Digital (A/D) and Digital to Analog (A/D) conversions which are necessary with analog circuits. The DMS-250 can also switch voice and data circuits.

The DMS-250 takes either analog or digitally encoded info and by using time slot interchange, switches it from any input port to a temporary addressed and connected exit port. The info may or may not be converted back to analog.

Normal Private Telecommunications Network Diagram:



Cellular Mobile Radio Service

A cellular system consists of two main parts a cellular switch and cell site equipment.

Cellular Switching Systems

A cellular switch performs three main functions audio switching, cell site control, and system administration.

The DMS switches provide three basic implementations for cellular switching Stand-alone, Combined, and Remote.

Stand-alone switching is done by a Mobile Telephone Exchange (MTX) which is interfaced with one or more class 5 end offices. The connection is made by DID/DOD trunks. Depending on the needs of the area, the MTX can be divided as follows: MTX which serves urban areas, MTXC which handles suburban areas, and MTXM which is used for rural areas.

Combined switching is incorporated into a DMS-100 by some hardware additions and cellular software. Combined switching is designed to give a easy, cost-effective way to install cellular services to an existing host.

Remote Switching is done by combining Remote Switching Center (RSC) with a Cell Site Controller (CSC). This combination is hosted by either a stand-alone or a combined switch. Remote Switching is designed for serving suburban centers, remote areas, or a small community and it gives extra flexibility for a growing system.

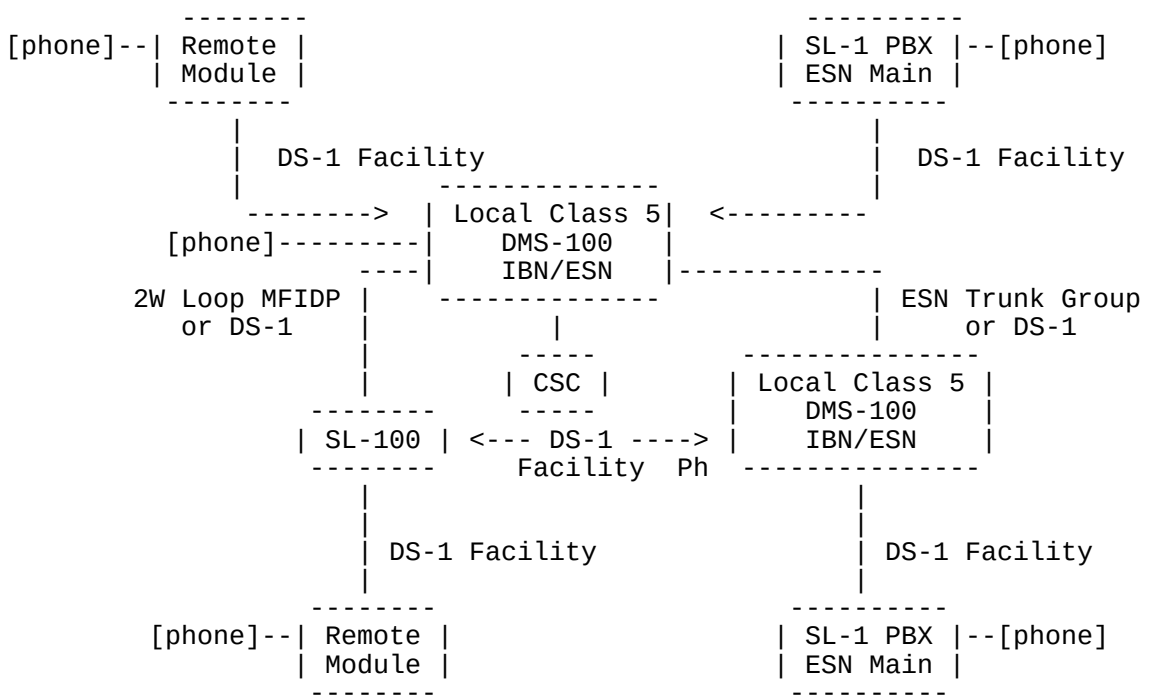
All of these cellular switches have the ability to balance the workload among various cell sites. For example, if one site's workload reaches the programmable level of congestion, calls would be routed to nearby sites that can handle the extra calls.

Cell Site Equipment

Cell site equipment consists of a CSC and radio equipment. The CSC is controlled by the cellular switch and it controls radio equipment and maintenance tasks. The CSC will work on any MTX cellular switch because of the Remote Cluster Controller (RCC).

The radio equipment consists of self-contained Radio Channel Units (RCU), antennas, transmitter multi-couplers and receiver combiners.

By different program software a RCU can perform voice, control locating, and test functions. The self contained nature allows the RCU be remotely located to the CSC. A RCU has built-in circuitry for extended testing of the radio part of the system.



<End of File>
<5-23-87>

If you have any questions contact me or any other member of the T0K!

Control C

!T0K! (1987)

==Phrack Inc.==

Volume Two, Issue 19, Phile #4 of 8

Social Security Number Formatting

Shooting Shark 21 June 88

Certain types of computer-related fraud, such as creating dummy entries in payroll databases, require the creation of a false Social Security Number (SS#). Many employers attempt to detect "ghost" SS#s by running a verification program on them. In this article I will show how to defeat verification by creating a legitimate-looking SS#.

First, some general rules to follow:

- The middle two digits of a SS# can be odd or even if issued after 1965. All numbers issued before 1965 that have middle digits of 10 or above should be even.
- So far, no SS#s have been issued with a first digit of 8 or 9. Very few numbers above 595 have been issued, so use caution. 700-729 were issued by the Railroad Retirement Agency a long time ago, and thus would belong to older people. No numbers in the 596-626 have been assigned yet (as far as I know), but 596-599 has been reserved for Puerto Rico, 600-601 for Arizona, and 602-626 has been reserved for California.

The next step is required only if it is necessary that the place of issuance (and thus, probably, state of birth or residence) match the SS#. In this case, refer to the following table:

First Three Digits	Area
000	Foreign-Exchange, visitor, etc. (many college students will have these)
001-003	New Hampshire
004-007	Maine
008-009	Vermont
010-034	Massachusetts
035-039	Rhode Island
040-049	Connecticut
050-134	New York
135-158	New Jersey
159-211	Pennsylvania
212-220	Maryland
221-222	Delaware
223-231	Virginia
232-236 (EXCEPT SS#s starting with "232 30"...)	West Virginia
232 30	North Carolina
237-246	North Carolina
247-251	South Carolina
252-260	Georgia
261-267	Florida
589-595	Florida
268-302	Ohio
303-317	Indiana
318-361	Illinois
362-386	Michigan
387-399	Wisconsin
400-407	Kentucky
408-415	Tennessee
416-424	Alabama
425-428	Mississippi
587-588	Mississippi
429-432	Arkansas
433-439	Louisiana
440-448	Oklahoma

449-467	Texas	468-477	Minnesota
478-485	Iowa	486-500	Missouri
501-502	North Dakota	503-504	South Dakota
505-508	Nebraska	509-515	Kansas
516-517	Montana	518-519	Idaho
520	Wyoming	521-524	Colorado
525	New Mexico	585	New Mexico
526-527	Arizona	528-529	Utah
530	Nevada	531-539	Washington
540-544	Oregon	545-573	California
574	Alaska	575-576	Hawaii
577-579	Washington, D.C.	580	Virgin Islands
580-584	Puerto Rico		

586	Guam, American Samoa, and Philippine Islands

700-729	Railroad Retirement

An example: If you were Stan Cisneros living in Burlingame, California, and you were born in 1970, your SS# might be 546-28-4197.

==========

==Phrack Inc.==

Volume Two, Issue 19, Phile #5 of 8

Facility Assignment and Control System

Written by Phantom Phreaker

INTRODUCTION

The Facility Assignment and Control System (FACS) is an integrated network component system that most phreaks and hackers know of from an old file named 'FACS FACTS' written by Sharp Razor. While this file provides an accurate description of the FACS system, it is lacking in detail and length. This file will provide accurate information about the FACS system and is intended for the true telecom enthusiast (i.e. this article is not for people who use codes and call it 'phreaking' or for people who think that phreaking is just 'making free phone calls'). Hopefully the phreaks and hackers of the world who want to know how things work in the telephone network will benefit from this information. Any malicious use of this information is strictly prohibited. The contents of this file are for informational and educational purposes only.

GENERAL DESCRIPTION

FACS can be described as a full-featured outside plant and central office facilities assignment system. For the people who are unfamiliar with these terms, the outside plant is the portion of the telephone network that runs from a telco office (such as a class five end office (EO)) to the subscriber, including manholes and distribution/access points such as Serving Area Interfaces (SAI) which are large, double-door outdoor equipment cabinets which allow the repair craft to repair, test, and access a multitude of service lines in that area.

FACS is made up of five component systems, or sub-systems, and some of these are also used as stand-alone systems (i.e. in an area that does not use FACS, COSMOS can be thought of as a stand-alone system).

The component systems are:

PREMIS - PREmise Information System
SOAC - Service Order Analysis & Control
LFACS - Loop Facility Assignment and Control System
COSMOS - COmputer System for Main Frame OperationS
WM - Work Manager

FACS is used by many departments and work centers in the BOC network. A general example of telco interaction will be included later in the article.

PREMIS

PREMIS supports the customer negotiation (i.e. while a customer talks with a BOC service rep, PREMIS is the computer system the rep has access to) and service order (SO) preparation process (a SO is basically a request for

service). PREMIS is a computer-based information storage and retrieval system designed to support the Residence/Residential Service Center (RSC), and in some cases, the Business Service Center (BSC). The RSC is the center that residence customers deal with, and the BSC is the center that business customers deal with.

PREMIS provides fast easy access to customer address verification for numbered and unnumbered addresses (information is stored by telephone number not address), telephone service status at an address (whether the phone is in service, disconnected, pending connect, pending disconnect, disconnected due to non-payment, etc.), telephone number assignment for customers (PREMIS can generate a list of available telephone numbers in a given exchange and the available TNs come from COSMOS) and facility assignment data for outward orders.

The following PREMIS features are available to the service reps and have special significance to the LAC:

Customer Negotiation:

Provides customer service address check against a mechanized Street Address Guide (SAG).

Provides customer status check to a mechanized facility address file which identifies potential Interfering Station (IS) conditions.

Provides new telephone number assignments through an available TN (Telephone Number) file.

Service Order Preparation:

Provides SAG data.

Provides correct address spelling.

PREMIS, as far as I know, does not have any direct dialups so don't get your hopes up high. There may be other ways to access information in PREMIS however.

SOAC

The SOAC system is what interfaces FACS with the BOC SOP (Service Order Processor). The SOP is what the service reps enter SO information into and the SOP sends the data entered to the SOAC system. The SOAC system interprets and validates this input data.

SOAC generates Assignment Requests (ARs) which are sent to LFACS and COSMOS (see respective sections of this file) to request outside plant (OSP) and CO facility assignments, respectively.

SOAC receives AR Responses (ARRs) from LFACS and WM/COSMOS and merges this data and formats the output into a Universal Service Order (USO) assignment section. This USO is returned to the SOP after SOAC has processed it.

SOAC returns status information and error notification to the SOP. Status information is what tells the service rep who entered the data into the SOP

whether or not FACS can process that Service Order. Error notifiers are sent back to the SOP when part of the SO is in error.

SOAC keeps record of status and control information on all SO requests, as well as the input image and specific data that came from processing. This information, along with the input image and processing results are referred to as the pending assignment data.

SOs do not automatically flow through SOAC in all cases. SOAC can analyze an order to determine if manual assistance is required, and if it is, a Request for Manual Assistance (RMA) notice is sent to the LAC. LAC personnel will use SOAC, and possibly other systems in FACS, such as COSMOS/WM and LFACS, to complete the assignment on that SO.

SOAC also may receive COSMOS system output from certain commands. One such command may be the IJR command, which sets up a circuit for jeopardy status. Jeopardy status means that the assignment looks as if it will be (or already is) behind schedule for completion. An example of this is as follows (showing COSMOS messages).

```
WC% IJR
H ORD nxxxxxxxx/TN nxx-xxxx/JR nx
RMK NEED TIE PR FOR nxx
-.
**ORDER nxxxxxxxx              HAS BEEN GIVEN JEOPARDY STATUS
  CKTID: TN nxx-xxxx
**JEOPARDY REASON: nx          mm-dd-yy hh:mm
OUTPUT SENT TO SOAC
**IJR COMPLETED   mm-dd-yyy    hh:mm
```

The H-line input is the SO number, where n can be alphabetic and x can be numeric. TN is the affected telephone number, JR is the Jeopardy Reason, which is a one alpha/one numeric code, RMK is a ReMarK, in this case, a tie pair is needed. The section that starts and ends with two asterisks is the COSMOS output, and the rest of the information should be self-explanatory.

LFACS

The LFACS system keeps an inventory of outside loop plant facilities, such as cables (CA), cable pairs (CP), serving terminals, interconnecting points, cross-connecting terminals, and things of that nature which should be known to the serious phreak. By the way, if you want to get some very good information about the outside loop plant, look for Phucked Agent 04's article in the LOD/H Technical Journal issue number 1. These are excellent files and I recommend that every phreak read them if they haven't already. Anyway, LFACS also assigns the outside loop plant facilities to ARs received from SOAC as a result of customer SO activity. The assignment process is automatic on 95% of the service requests.

LFACS provides a computerized version of DPAC and ECCR (Dedicated Plant Assignment Cards and Exchange Cable Conductor Records respectively) which were previously physical records that were stored at the LAC. The information stored in DPAC is information such as data about a Living Unit Serving Terminal, and Living Unit Dedicated Loop Facilities, and ECCR contains information such as Pair Selection, Add/Break count, Line and Station Transfer, as well as Work Order (WO) information. Some of this information may be used by the LAC Field Assistance Bureau to assist the outside plant craft in obtaining necessary information.

When conditions necessary for LFACSS to automatically respond to a SOAC AR are not met, a RMA noticed is generated in the LAC. Appropriate people in the LAC will interact with LFACS, and maybe SOAC and WM/COSMOS to complete the process of assignment.

COSMOS

COSMOS has been written about many times, so I will not go into deep detail about this system as many people are already familiar with it.

COSMOS keeps a database inventory of CO facilities (such as TN, CP, OE, CS, BL - telephone number, cable pair, office equipment, class of service, bridge lifter respectively) and assigns these facilities to ARs received from SOAC as a result of customer SO activity.

COSMOS assists the Network Administration Center (NAC) and Frame Control Center (FCC) in managing, controlling, and utilizing the MDF and COE, as well as CO facilities and circuits. COSMOS does assignment of TNs, line equipment, jumper use/reuse, TP management, frame work management, and other things of that nature.

When the conditions are not met for COSMOS to respond to a SOAC AR, a RMA is generated in the LAC (as with the other systems mentioned in this article). The LAC can then use WM/COSMOS, SOAC, and LFACS to complete assignment.

WM
--

The WM is what links one set of SOAC/LFACS systems with one or more COSMOS systems. All input to COSMOS from the LAC is directed through the WM. The WM provides message switching, load control, and other functions to the LAC.

-EOF-

RC:LINE;CHNG!/ORD 1/TN LOD-LOD-LODH/ESM YES/ESX YES/ESL YES/RC:TRK!/TNN LOD.

I hope the information presented in this article has been of interest to all who read it. I have not included as much information as I could have, some sensitive information has not been included because it could cause problems. My personal thanks goes out to the fine people who designed the FACS system, as well as to all the telephone companies in existence, for without you, phone phreaks would not exist. Thank you for allowing us access to your networks, although this access is taken rather than given. Try hiring a phreak sometime, it might be beneficial.

A note to telecom/computer enthusiasts who read this article: DO NOT SCREW ANYTHING UP! IF YOU ARE NOT RESPONSIBLE ENOUGH TO USE THIS DATA IN A WISE AND NON-ABUSIVE WAY THEN DISCARD THIS ARTICLE NOW AND PRETEND LIKE YOU NEVER READ IT.

==========

==Phrack Inc.==

Volume Two, Issue 19, Phile #6 of 8

Phrack Editorial on Microbashing

I was toying with the idea of writing a history of the Microcomputer Revolution, viewed through the eyes of one who lived through it, perhaps with some recollections of a Telecommunications Hobbyist thrown in for spice.

Upon reflection however, I thought that I might use this forum to address a problem that has bothered me for some time. I refer to the phenomena of microbashing.

This is, in my opinion, a serious problem in the MicroUnderground.

For the record, I'm 36 years old, I have been screwing around with computers, Mainframe, Mini and Micro since 1976/77. I built an Altair 8800 way back when, and wrote what may have been the first software pirating program. (Something that mass produced papertape copies of Bill Gates' Altair BASIC). I also built a TV Typewriter based on Don Lancaster's designs, and a 100 baud modem to go along with it. For the record, I use a Commodore 64 computer. I have a 1200 baud modem, two disk drives, a spiffy printer and a color monitor. For the record, I sold an Apple //e to buy the C64. I have never regretted that decision.

Now, there are those who will read that last sentence and say to themselves, "Fuckin' Commie user! He SOLD an Apple to buy a Commie? What an asshole!" Now, I could say to the Apple //e user who thinks that, "You poor boob! You spent all that money for a //e! Plus all that extra cash for plug in cards so it can do what my C64 has built in? Geeze! Some folks need keepers!"

That, Gentle Readers, is microbashing. So, in the space of a few minutes, this hypothetical exchange has engendered ill feelings, if not outright hostility. What a waste of time and effort! We both have powerful computers that I could not even begin to imagine could exist 12 (12!) years ago. My Altair had 16k of RAM in it, and I thought that was hot stuff! Most folks only had 4 to 8k in their homebrew micros. I even had a disk drive! A huge monster that weighed 20 pounds, used 8 inch single sided disks that had all of 120k of storage. This whole system, complete with TeleType (my terminal/printer) cost about $5000 in 1977 dollars. In 1988 dollars, maybe $15000. (My little C64 system, total cost less than $1000 just blows that Altair/Teletype out of the water).

What are the roots of microbashing? I'm not sure, but here are some thoughts.

Status, I'm sure, plays a major role in microbashing. A C64/128 will always cost less than an equivalent Apple //e system. "My computer cost more than your computer! Therefore, my computer is better! Nyah!" By that logic, my old $5000/$15000 Altair is a better computer than most Apple machines. Patently ridiculous, isn't it? (I've noticed that there is now a Let's Bash the //e subculture developing among the Mac Plus, SE and II crowd, along with //gs users. I do take a perverse pleasure, I'm sorry to say, with all this. The shoe is now on the other foot, eh?)

Conformity, particularly among the teenage/young adult users, might also be a factor. "Everyone important uses Apples. Only gameplayers use Kmart toy computers. If you don't use an Apple, you ain't shit!" The peer pressure of Conformity is a powerful thing.

A mate of mine in the Computer Services department at Harvard has a Mac II on his desk at work and a Mac Plus at home. Another friend has a Zenith AT clone at his office at the Mitre Corporation in Maryland and an Apple][+ at home. A good friend of mine who's an editor at a major disk-based publication had a //gs given to him by Apple. All these guys are high powered computer users. The guy at Harvard is their UNIX wizard. The fellow in MD is a GS-13 employed by the Air Force as a general purpose MS-DOS/ADA wizard, and just spent $1000000 to fund distributed processing research at Los Alamos. The last person is the Apple edition editor at this publication. Not a single one of them wastes a second denigrating my C64. Now, if these guys consider me a peer, an equal, (and they do!) and they don't care what computer I use, why do some //e users waste their time and energy putting down the C64?

A third factor may be the sneaking suspicion that, "Geeze! If a C64 can do all that, why did I spend all that money on an Apple?" Guilt and self doubt can be a powerful factor in microbashing. "If I put Commies down enough, maybe other people will buy Apples and then I won't be the only one who has one." Psychologists call that "Transference." Transferring the negative feelings/doubt about oneself to something else and then denigrating that something else. The Old Testament calls it a "Scapegoat."

I suppose what I'm finally trying to say is let's all grow up and stop this foolish bickering and sniping. No one profits, and we all lose. We lose time, information, disk space on BBSs, companionship and fun! I don't like to see some Apple user bashing Commodore. Neither do I enjoy seeing a C64 user bashing a TI user, as I dislike watching that TI user make fun of someone with an Adam. Don't you think we have more important things to do than make mountains out of molehills when it comes to our respective computers?

I do. If we can't act any better than a kindergarten kid whining over a toy, then maybe we don't deserve these powerful tools we have sitting on our desktops.

Written by THE NIGHTSTALKER, June, 1988.
==========

==Phrack Inc.==

Volume Two, Issue 19, Phile #7 of 8

```
PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN
PWN                                         PWN
PWN      >>>>>=-* Phrack World News *-=<<<<<<    PWN
PWN                  Issue XVIV/1               PWN
PWN                                         PWN
PWN           Created by Knight Lightning       PWN
PWN   Written and compiled by Knight Lightning  PWN
PWN                                         PWN
PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN
```

>From The Creators Of Phrack Incorporated...

The Phoenix Project
>>>>>>>>>>>>>>>>>>>

-=-

Just what is "The Phoenix Project?"

Definition: Phoenix (fe/niks), n. A unique mythical bird of great beauty fabled to live 500 or 600 years, to burn itself to death, and to rise from its ashes in the freshness of youth, and live through another life cycle.

Project (proj/ekt), n. Something that is contemplated, devised, or planned. A large or major undertaking. A long term assignment.

- -

Why is "The Phoenix Project?"

On June 1, 1987 Metal Shop Private went down seemingly forever with no possible return in sight, but the ideals and the community that formed the famous center of learning lived on. On June 19-21, 1987 the phreak/hack world experienced SummerCon'87, an event that brought much of the community together whether physically appearing at the convention or in spirit. On July 22, 1987 the phreak/hack community was devastated by a nationwide attack from all forms of security and law enforcement agencies...thus setting in motion the end of the community as we knew it. Despite the events of July 22, 1987, PartyCon'87 was held on schedule on July 26-28, 1987 as the apparent final gathering of the continent's last remaining free hackers, unknown to them the world they sought to protect was already obliterated. As of August 1, 1987 all of the original members and staff of the Metal Shop Triad and Phrack Inc. had decided to bail out in the hopes that they could return one day when all would be as before...

THAT DAY HAS COME...

A new millennium is beginning and it all starts on July 22, 1988. How fitting that the One year anniversary of the destruction of the phreak/hack community should coincidentally serve as the day of its rebirth.

Announcing SummerCon '88 in (where else would you expect) St. Louis, Missouri!

Knowledge is the key to the future and it is FREE. The telecommunications and security industries can no longer withhold the right to learn, the right to explore, or the right to have knowledge. The new age is here and with the use of every *LEGAL* means available, the youth of today will be able to teach the youth of tomorrow.

- -

SummerCon'88 is a celebration of a new beginning. Preparations are currently underway to make this year's convention twice as fun as last year's and the greater the turnout the greater the convention shall be. No one is directly excluded from the festivities and the practice of passing illegal information is not a part of this convention (contrary to the opinions of the San Francisco Examiner, and they weren't even at the last one). Anyone interested in appearing at this year's convention should leave mail to Crimson Death immediately so we can better plan the convention for the correct amount of participants.

The hotel rooms purchased for SummerCon'88 are for the specified use of invited guests and no one else. Any security consultants or members of law enforcement agencies that wish to attend should contact the organizing committee as soon as possible to obtain an invitation to the actual convention itself.

Sorry for the short notice this year...

:Knight Lightning "The Future Is Forever"

-=-=-=-=--=-=-=-=-=-=-=-=-=-=-=-=-=-=-=--=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-

===

==Phrack Inc.==

Volume Two, Issue 19, Phile #8 of 8

```
PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN
PWN                                         PWN
PWN      >>>>>=-* Phrack World News *-=<<<<<    PWN
PWN                 Issue XVIV/2                PWN
PWN                                         PWN
PWN          Created by Knight Lightning        PWN
PWN        Written and compiled by Epsilon      PWN
PWN                                         PWN
PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN
```

Doc Holiday In Legal Trouble

One night, Doc Holiday 713 decided to visit his CO. This CO was surrounded by a fence with barbed wire on top. He climbed over the fence with ease and looked around the perimeter of the building for any cameras. When he was sure that there were no cameras, he decided to try entering through the back door. To his surprise, the back door was unlocked [Hey, at least he didn't get charged with breaking and entering, right? -Epsilon], and he entered the building. He looked around a bit, past some boxes full of test sets and cable. This got boring, so he headed down to a room with some terminals and a large control panel. The instructions for using the terminal were taped to the side of the desk, so he tried them out. He had fun monitoring phone lines and testing other subscribers' touch tone polarity, and he decided to get out of the building.

On his way out of the building, he came across a box that was labeled with something to the effect of 'Switching Unit'. He didn't bother to look inside the box, because he was in a hurry to get out of the building, so he opted to take the box home, then look inside. When he opened the door to leave, he saw a flashlight waving around in the dark. He got scared and set the box down [Incidentally, this door that he got in through was at the top of a stairway at the back of the building, outside. -Epsilon]. The box was unstable, and rolled down the stairs, probably causing damage to whatever was inside. He tried to run down the stairs and climb over the fence. He found that the police were outside the fence, and he proceeded to run. A policeman shouted at him to stop, and threatened to shoot him, so he dropped. He was apprehended and taken to the police station.

He had learned that the police knew about his whereabouts, because he had tripped a silent alarm, probably upon entering the building.

At the station, they questioned him. He was getting fed up, and said he was going to leave the station. He started to leave, when a policeman grabbed him and kneeing him, broke his rib. They then, after some persuasion, took Doc to the hospital.

He is at home now, and awaiting a hearing. This little tiff is not expected to affect his hacking activities.

Information Provided By Doc Holiday

The Disk Jockey...Busted!

The Disk Jockey, whom we all knew, was arrested for 22 counts of aiding a

fraud and other miscellaneous charges last Friday. He is now in jail and is being held for $150,000.00 bail [Yes, that's right. One-hundred-fifty-thousand dollars. -Epsilon].

This incident was believed to have been caused by a 'phreak' by the name of White Lightning (616), who informed Sprint Security that The Disk Jockey was using their service illegally.

He is now awaiting a court date, and is unavailable for questions.

Information Provided By Compaq (219)

SummerCon '88
========= ===

We at Phrack Inc. are proud to present SummerCon '88. The convention will take place at the Westport Ramada Inn in St. Louis, Missouri the week-end of July 22nd. The Con is expected to be held from Friday afternoon to Sunday afternoon. Please contact us, via the Phrack accounts, or the Phrack In. VMS at (800)331-8477, * #, Ext. 6660, if you plan to attend. Illegal information at this CON is not encouraged. Thank you.

Information Provided By Knight Lightning

PWN Quicknotes
=== ==========

The first step in what is called The Phoenix Project, which is a re-birth of the hack/phreak community is underway. This first step is a public education bulletin board system dedicated to teaching the public about telecommunications and computer systems. The board is called The Phoenix Project, and the number is (512)754-8182. No illegal information is to be posted on this system. Our SysOp is The Mentor. Thank you, and call if you're interested.

Information Provided By The Mentor

Rumor has it that a new group is forming in the hack/phreak community. This group is looking for about eight skilled members who have diverse interests. If you think you are qualified, and are interested, please contact Doc Holiday on any BBS he is on. Thank you.

Control C of 313 was NOT busted, contrary to popular belief. This is all a big confusion, and we will let you know how this started in a future issue. Until then, please, don't spread the rumor around anymore.

==========