



```

CCCCCCCC  VV  VV  TTTTTTTTTT  NN  NN  AAAAAA  FFFFFFFFFF
CCCCCCCC  VV  VV  TTTTTTTTTT  NN  NN  AAAAAA  FFFFFFFFFF
CC         VV  VV  TT          NN  NN  AA      AA  FF
CC         VV  VV  TT          NN  NN  AA      AA  FF
CC         VV  VV  TT          NNNN   NN  AA      AA  FF
CC         VV  VV  TT          NNNN   NN  AA      AA  FF
CC         VV  VV  TT          NN  NN  AA      AA  FFFFFFFF
CC         VV  VV  TT          NN  NN  AA      AA  FFFFFFFF
CC         VV  VV  TT          NN  NN  AA      AA  FF
CC         VV  VV  TT          NN  NN  AA      AA  FF
CC         VV  VV  TT          NN  NN  AA      AA  FF
CC         VV  VV  TT          NN  NN  AA      AA  FF
CC         VV  VV  TT          NN  NN  AA      AA  FF
CC         VV  VV  TT          NN  NN  AA      AA  FF
CCCCCCCC  VV  VV  TT          NN  NN  AA      AA  FF
CCCCCCCC  VV  VV  TT          NN  NN  AA      AA  FF

```

```

LL         IIIIII  SSSSSSSS
LL         IIIIII  SSSSSSSS
LL         II      SS
LL         II      SS
LL         II      SS
LL         II      SS
LL         II      SSSSSS
LL         II      SSSSSS
LL         II      SS
LL         II      SS
LL         II      SS
LL         II      SS
LL         II      SS
LLLLLLLLLL IIIIII  SSSSSSSS
LLLLLLLLLL IIIIII  SSSSSSSS

```

....  
....  
....  
....

.....  
.....  
.....  
.....  
.....  
.....

```

1 0001 0 MODULE CVTNAF ( %TITLE 'Convert Network Proxy File'
2 0002 0     MAIN = MAIN,
3 0003 0     IDENT = 'V04-000'
4 0004 0 ) =
5 0005 1 BEGIN
6 0006 1
7 0007 1
8 0008 1 *****
9 0009 1 *
10 0010 1 *  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
11 0011 1 *  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
12 0012 1 *  ALL RIGHTS RESERVED.
13 0013 1 *
14 0014 1 *  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
15 0015 1 *  ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
16 0016 1 *  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
17 0017 1 *  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
18 0018 1 *  OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
19 0019 1 *  TRANSFERRED.
20 0020 1 *
21 0021 1 *  THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
22 0022 1 *  AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
23 0023 1 *  CORPORATION.
24 0024 1 *
25 0025 1 *  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
26 0026 1 *  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
27 0027 1 *
28 0028 1 *
29 0029 1 *****
30 0030 1
31 0031 1
32 0032 1 ++
33 0033 1 FACILITY:      Convert User Authorization File
34 0034 1
35 0035 1 ABSTRACT:      Converts a Release 3 compatible NETNAF.DAT to Release 4 format.
36 0036 1 The new NETNAF.DAT offsets are assumed. The old offsets have
37 0037 1 been renamed to ONF$etc...
38 0038 1
39 0039 1 ENVIRONMENT:   Release 3 to Release 4 conversion procedures
40 0040 1
41 0041 1 AUTHOR:       Chris Hume, CREATION DATE: 22-May-1979
42 0042 1
43 0043 1 MODIFIED BY:
44 0044 1
45 0045 1     V03-002 ACG0385      Andrew C. Goldstein,    11-Jan-1984  21:34
46 0046 1     Move ONFDEF to LIB$
47 0047 1
48 0048 1     V03--001 ACG0385      Andrew C. Goldstein,    5-Jan-1984  19:43
49 0049 1     Crib from CVTUAF utility.
50 0050 1
51 0051 1 --
52 0052 1
53 0053 1 LIBRARY 'SYSSLIBRARY:LIB':
54 0054 1 REQUIRE 'LIB$:ONFDEF':
55 0098 1
56 0099 1 MACRO
57 M 0100 1 RETURN_IF_ERROR[] =

```

```
58      BEGIN
59      LOCAL RMSERR;
60      (IF NOT (RMSERR = %REMAINING) THEN RETURN .RMSERR)
61      END
62      %:
63
64      LITERAL
65      TRUE = 1;
66
67      MACRO
68      COPY_PADDED (NAME) =
69      COPY_PADDED1 (%EXPLODE (NAME))
70      %,
71
72      COPY_PADDED1 (C1, C2, C3, C4, C5) =
73      COPY_PADDED2 (%STRING ('ONF$$', %REMAINING),
74      %STRING ('ONF$', C5, %REMAINING),
75      %STRING ('NAF$$', %REMAINING),
76      %STRING ('NAF$', C5, %REMAINING))
77      %,
78
79      COPY_PADDED2 (OLENGTH, ONAME, NLENGTH, NNAME) =
80      CH$COPY (%NAME (OLENGTH), ONFBUF[%NAME (ONAME)],
81      %NAME (NLENGTH), NAFBUF[%NAME (NNAME)])
82      %,
83
84      COPY_COUNTED (NAME) =
85      COPY_COUNTED1 (%EXPLODE (NAME))
86      %,
87
88      COPY_COUNTED1 (C1, C2, C3, C4, C5) =
89      COPY_COUNTED2 (%STRING ('ONF$$', %REMAINING),
90      %STRING ('ONF$', C5, %REMAINING),
91      %STRING ('NAF$$', %REMAINING),
92      %STRING ('NAF$', C5, %REMAINING))
93      %,
94
95      COPY_COUNTED2 (OLENGTH, ONAME, NLENGTH, NNAME) =
96      CH$COPY (%VECTOR [ONFBUF[%NAME (ONAME)], 0;, BYTE]+1, ONFBUF[%NAME (ONAME)],
97      %NAME (NLENGTH), NAFBUF[%NAME (NNAME)])
98      %,
99
100     PADDED_TO_COUNTED (NAME) =
101     PADDED_TO_COUNTED1 (%EXPLODE (NAME))
102     %,
103
104     PADDED_TO_COUNTED1 (C1, C2, C3, C4, C5) =
105     PADDED_TO_COUNTED2 (%STRING ('ONF$$', %REMAINING),
106     %STRING ('ONF$', C5, %REMAINING),
107     %STRING ('NAF$$', %REMAINING),
108     %STRING ('NAF$', C5, %REMAINING))
109     %,
110
111     PADDED_TO_COUNTED2 (OLENGTH, ONAME, NLENGTH, NNAME) =
112     BEGIN
113     LOCAL COUNT, P : REF VECTOR [,BYTE];
114
```

```

115      M 0158 1      P = ONFBUF[%NAME (ONAME)] + %NAME (OLENGTH);
116      M 0159 1      UNTIL .P[-1] NEQ ;
117      M 0160 1      OR .P LEQA ONFBUF[%NAME (ONAME)]
118      M 0161 1      DO P = .P - 1;
119      M 0162 1      COUNT = .P - ONFBUF[%NAME (ONAME)];
120      M 0163 1      VECTOR [NAFBUF[%NAME (NNAME)], 0; , BYTE] = .COUNT;
121      M 0164 1      CH$COPY (, COUNT, ONFBUF[%NAME (ONAME)],
122      M 0165 1      ,
123      M 0166 1      %NAME (NLENGTH)-1, NAFBUF[%NAME (NNAME)]+1)
124      M 0167 1      END
125      M 0168 1      %:
126      M 0169 1
127      M 0170 1      EXTERNAL ROUTINE
128      M 0171 1      LIB$PUT_OUTPUT : ADDRESSING_MODE (GENERAL);
129      M 0172 1
130      M 0173 1      OWN
131      M 0174 1      NAFBUF: BLOCK[NAF$C_LENGTH, BYTE],
132      M 0175 1      GNFBUF: BLOCK[ONF$C_LENGTH, BYTE],
133      M 0176 1
134      P 0177 1      NAFKEY1: $XABKEY ( : XAB for UIC key
135      P 0178 1      KREF = 1, : alternate key
136      P 0179 1      POSO = $BYTEOFFSET (NAF$T_LOCALUSER),
137      P 0180 1      SIZO = NAF$S_LOCALUSER,
138      P 0181 1      FLG = (CHG, DUP),
139      P 0182 1      KNM = UPLIT BYTE ('Local Username '),
140      M 0183 1      ),
141      M 0184 1
142      P 0185 1      NAFKEY0: $XABKEY ( : XAB for USERNAME key
143      P 0186 1      KREF = 0, : primary key
144      P 0187 1      POSO = $BYTEOFFSET (NAF$T_REMNAME),
145      P 0188 1      SIZO = NAF$S_REMNAME,
146      P 0189 1      KNM = UPLIT BYTE ('Remote Node Name and Username '),
147      P 0190 1      NXT = NAFKEY1
148      M 0191 1      ),
149      M 0192 1
150      P 0193 1      NAFALL: $XABALL ( : XAB for NAF allocation
151      P 0194 1      AOP = CBT,
152      P 0195 1      BKZ = 4,
153      P 0196 1      NXT = NAFKEY0
154      M 0197 1      ),
155      M 0198 1
156      P 0199 1      NAFPRO: $XABPRO ( : XAB for file protection
157      P 0200 1      PRO = (RWED, RWED, RWE,), : deny world access
158      P 0201 1      NXT = NAFALL
159      M 0202 1      ),
160      M 0203 1
161      P 0204 1      NAFFAB: $FAB ( : FAB for work file
162      P 0205 1      FAC = PUT, : access types
163      P 0206 1      FNM = 'NEWNAF', : temporary file name
164      P 0207 1      DNM = '.DAT',
165      P 0208 1      SHR = NIL,
166      P 0209 1      ORG = IDX, : indexed operations
167      P 0210 1      RFM = FIX,
168      P 0211 1      MRS = NAF$C_LENGTH, : maximum record size
169      P 0212 1      XAB = NAFPRO : file protection XAB
170      M 0213 1      ),
171      M 0214 1

```

```

: 172 P 0215 1 NAFRAB: $RAB ( . RAB for temporary file
: 173 P 0216 1 RSZ = NAF$C_LENGTH,
: 174 P 0217 1 RBF = NAF$BUF,
: 175 P 0218 1 FAB = NAFFAB
: 176 P 0219 1 ),
: 177 P 0220 1
: 178 P 0221 1 ONFKEY1: $XABKEY ( ! XAB to read old NAF key
: 179 P 0222 1 KREF = 1
: 180 P 0223 1 ),
: 181 P 0224 1
: 182 P 0225 1 ONFKEY0: $XABKEY ( ! XAB to read old NAF key
: 183 P 0226 1 KREF = 0,
: 184 P 0227 1 NXT = ONFKEY1
: 185 P 0228 1 ),
: 186 P 0229 1
: 187 P 0230 1 ONFFAB: $FAB ( ! FAB for old NAF file
: 188 P 0231 1 FAC = GET, ! read only
: 189 P 0232 1 FNM = 'OLDNAF',
: 190 P 0233 1 DNM = '.DAT',
: 191 P 0234 1 XAB = ONFKEY0
: 192 P 0235 1 ),
: 193 P 0236 1
: 194 P 0237 1 ONFRAB: $RAB ( ! RAB for old NAF
: 195 P 0238 1 RAC = SEQ,
: 196 P 0239 1 KRF = 0, ! key of reference
: 197 P 0240 1 UBF = ONF$BUF,
: 198 P 0241 1 USZ = ONF$C_LENGTH,
: 199 P 0242 1 FAB = ONFFAB
: 200 P 0243 1 );
: 201 P 0244 1
: 202 P 0245 1 ROUTINE MAIN =
: 203 P 0246 2 BEGIN
: 204 P 0247 2
: 205 P 0248 2 LOCAL
: 206 P 0249 2 RMSERR;
: 207 P 0250 2
: 208 P 0251 2 !
: 209 P 0252 2 ! Open the old NAF for input
: 210 P 0253 2 !
: 211 P 0254 2 RETURN_IF_ERROR ($OPEN (FAB = ONFFAB));
: 212 P 0255 2 RETURN_IF_ERROR ($CONNECT (RAB = ONFRAB));
: 213 P 0256 2 !
: 214 P 0257 2 ! Validate the keys of the NAF, checking if it is a new format,
: 215 P 0258 2 ! and then for old format validity.
: 216 P 0259 2 !
: 217 P 0260 2 IF .ONFKEY0[XAB$W_POS0] EQL $BYTEOFFSET (NAF$T_REMNAME)
: 218 P 0261 2 AND .ONFKEY0[XAB$B_SIZE] F NAF$S_REMNAME
: 219 P 0262 2 AND .ONFKEY1[XAB$W_POS0] L $BYTEOFFSET (NAF$T_LOCALUSER)
: 220 P 0263 2 AND .ONFKEY1[XAB$B_SIZE] EQL NAF$S_LOCALUSER
: 221 P 0264 2 THEN
: 222 P 0265 3 BEGIN
: 223 P 0266 3 LIB$PUT OUTPUT ($DESCRIPTOR ('Proxy file is already converted'));
: 224 P 0267 3 RETURN ST$M_INHIB_MSG;
: 225 P 0268 2 END;
: 226 P 0269 2 IF .ONFKEY0[XAB$W_POS0] NEQ $BYTEOFFSET (ONF$T_REMNAME)
: 227 P 0270 2 OR .ONFKEY0[XAB$B_SIZE] NEQ ONF$S_REMNAME
: 228 P 0271 2 OR .ONFKEY1[XAB$W_POS0] NEQ $BYTEOFFSET (ONF$T_LOCALUSER)

```

```

229 0272 2 OR .ONFKEY1[XAB$B_SIZE] NEQ ONF$S_LOCALUSER
230 0273 2 THEN
231 0274 2 BEGIN
232 0275 2 LIB$PUT OUTPUT ($DESCRIPTOR ('Proxy file has invalid format'));
233 0276 2 RETURN ST$SM_INHIB_MSG;
234 0277 2 END;
235 0278 2
236 0279 2 ! Create the new NAF.
237 0280 2 !
238 0281 2 NAFALL[XAB$L_ALQ] = .ONFFAB[FAB$L_ALQ] / 2 + .ONFFAB[FAB$L_ALQ];
239 0282 2 RETURN_IF_ERROR ($CREATE (FAB = NAFFAB));
240 0283 2 RETURN_IF_ERROR ($CONNECT (RAB = NAFRAB));
241 0284 2
242 0285 2 WHILE (RMSERR = $GET (RAB = ONFRAB))
243 0286 2 DO
244 0287 2 BEGIN
245 0288 2 CH$FILL (0, NAF$C_LENGTH, NAFBUF);
246 0289 2 COPY_PADDED ('NAF$T_NODE');
247 0290 2 COPY_PADDED ('NAF$T_REMUSER');
248 0291 2 COPY_PADDED ('NAF$T_LOCALUSER');
249 0292 2 RETURN_IF_ERROR ($PUT (RAB = NAFRAB));
250 0293 2 END;
251 0294 2 IF .RMSERR NEQ RM$S_EOF THEN RETURN .RMSERR;
252 0295 2 RETURN_IF_ERROR ($CLOSE (FAB = NAFFAB));
253 0296 3 $CLOSE (FAB = ONFFAB)
254 0297 1 END;

```

														.TITLE	CVTNAF Convert Network Proxy File					
														.IDENT	\V04-000\					
														.PSECT	\$SPLITS,NOVRT,NOEXE,2					
20	65	6D	61	6E	72	65	73	55	20	6C	61	63	6F	4C	00000	P.AAA:	.ASCII	\Local Username	\	
20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	0000F					
6D	61	4E	20	65	64	6F	4E	20	65	74	6F	6D	65	52	00020	P.AAB:	.ASCII	\Remote Node Name and Username	\	
20	65	6D	61	6E	72	65	73	55	20	64	6E	61	20	65	0002F					
															0003E					
										46	41	4E	57	45	4E	00040	P.AAC:	.ASCII	\NEWNAF\	
															00046	P.AAD:	.ASCII	\.DAT\		
										46	41	4E	44	4C	4F	0004A	P.AAE:	.ASCII	\OLDNAF\	
															00050	P.AAF:	.ASCII	\.DAT\		
61	20	73	69	20	65	6C	69	66	20	79	78	6F	72	50	00054	P.AAH:	.ASCII	\Proxy file is already converted\		
65	74	72	65	76	6E	6F	63	20	79	64	61	65	72	6C	00063					
															00072					
															00073					
															00074	P.AAG:	.BLKB	1		
															00078		.LONG	31		
															0007C	P.AAJ:	.ADDRESS	P.AAH		
20	73	61	68	20	65	6C	69	66	20	79	78	6F	72	50	0007C	P.AAJ:	.ASCII	\Proxy file has invalid format\		
74	61	6D	72	6F	66	20	64	69	6C	61	76	6E	69	0008B						
															00099		.BLKB	3		
															0009C	P.AAI:	.LONG	29		
															000A0		.ADDRESS	P.AAJ		
																	.PSECT	\$OWNS,NOEXE,2		

```

00000 NAFBUF: .BLKB 100
00064 ONFBUF: .BLKB 30
00082 .BLKB 2
15 00084 NAFKEY1: .BYTE 21
4C 00085 .BYTE 76
0000 00086 .WORD 0
00000000 00088 .LONG 0
00 0008C .BYTE 0
00 0008D .BYTE 0
00 0008E .BYTE 0
00 0008F .BYTE 0
00 00090 .BYTE 0
00 00091 .BYTE 0
00000000 00092 .LONG 0
03 00096 .BYTE 3
00 00097 .BYTE 0
00 00098 .BYTE 0
00 00099 .BYTE 0
00 0009A .BYTE 0
01 0009B .BYTE 1
0000 0009C .WORD 0
0000 0009E .WORD 0
0000 000A0 .WORD 0
0040 000A2 .WORD 64
0000 000A4 .WORD 0
0000 000A6 .WORD 0
0000 000AB .WORD 0
0000 000AA .WORD 0
0000 000AC .WORD 0
0000 000AE .WORD 0
0000 000B0 .WORD 0
20 000B2 .BYTE 32
00 000B3 .BYTE 0
00 000B4 .BYTE 0
00 000B5 .BYTE 0
00 000B6 .BYTE 0
00 000B7 .BYTE 0
00 000B8 .BYTE 0
00 000B9 .BYTE 0
0000 000BA .WORD 0
00000000 000BC .ADDRESS P.AAA
00000000 000C0 .LONG 0
00 000C4 .BYTE 0
00 000C5 .BYTE 0
00 000C6 .BYTE 0
00 000C7 .BYTE 0
00 000C8 .BYTE 0
00 000C9 .BYTE 0
00 000CA .BYTE 0
00 000CB .BYTE 0
00 000CC .BYTE 0
00 000CD .BYTE 0
0000 000CE .BLKB 2
15 000D0 NAFKEY0: .BYTE 21
4C 000D1 .BYTE 76
0000 000D2 .WORD 0
C0000000 000D4 .ADDRESS NAFKEY1

```

.....

.....



```

00 000D8 .BYTE 0
00 000D9 .BYTE 0
00 000DA .BYTE 0
00 000DB .BYTE 0
00 000DC .BYTE 0
00 000DD .BYTE 0
00000000 000DE .LONG 0
00 000E2 .BYTE 0
00 000E3 .BYTE 0
00 000E4 .BYTE 0
00 000E5 .BYTE 0
00 000E6 .BYTE 0
00 000E7 .BYTE 0
0000 000E8 .WORD 0
0000 000EA .WORD 0
0000 000EC .WORD 0
0000 000EE .WORD 0
0000 000F0 .WORD 0
0000 000F2 .WORD 0
0000 000F4 .WORD 0
0000 000F6 .WORD 0
0000 000F8 .WORD 0
0000 000FA .WORD 0
0000 000FC .WORD 0
00 000FE .BYTE 64
00 000FF .BYTE 0
00 00100 .BYTE 0
00 00101 .BYTE 0
00 00102 .BYTE 0
00 00103 .BYTE 0
00 00104 .BYTE 0
00 00105 .BYTE 0
0000 00106 .WORD 0
00000000 00108 .ADDRESS P.AAB
00000000 0010C .LONG 0
00 00110 .BYTE 0
00 00111 .BYTE 0
00 00112 .BYTE 0
00 00113 .BYTE 0
00 00114 .BYTE 0
00 00115 .BYTE 0
00 00116 .BYTE 0
00 00117 .BYTE 0
00 00118 .BYTE 0
00 00119 .BYTE 0
00 0011A .BLKB 2
00 0011C NAFALL: .BYTE 20
00 0011D .BYTE 32
0000 0011E .WORD 0
00000000 00120 .ADDRESS NAFKEY0
00 00124 .BYTE 32
00 00126 .BYTE 0
0000 00128 .WORD 0
00000000 0012C .LONG 0
00000000 00130 .LONG 0
0000 00132 .WORD 4
00 00132 .BYTE 4

```

.....

CV  
VO  
.....

```

0000 0000 0000 00133 .BYTE 0
0000 0000 0000 00134 .WORD 0, 0, 0
0000 0000 0000 0013A .WORD 0
13 0013C NAFPRO: .BYTE 19
58 0013D .BYTE 88
0000 0013E .WORD 0
00000000 00140 .ADDRESS NAFALL
F800 00144 .WORD -2048
00 00146 .BYTE 0
00 00147 .BYTE 0
0000 0000 00148 .WORD 0, 0
00 0014C .BYTE 0
00 0014D .BYTE 0
0000 0014E .WORD 0
00000000 00150 .LONG 0
00000000 00154 .LONG 0
0000 00158 .WORD 0
0000 0015A .WORD 0
00000000 0015C .LONG 0
00000000 00160 .LONG 0
00164 .BLKB 48
03 00194 NAF F AB: .BYTE 3
50 00195 .BYTE 80
0000 00196 .WORD 0
00000000 00198 .LONG 0
00000000 0019C .LONG 0
00000000 001A0 .LONG 0
00000000 001A4 .LONG 0
0000 001A8 .WORD 0
01 001AA .BYTE 1
20 001AB .BYTE 32
00000000 001AC .LONG 0
00 001B0 .BYTE 0
20 001B1 .BYTE 32
00 001B2 .BYTE 0
01 001B3 .BYTE 1
00000000 001B4 .LONG 0
00000000 001B8 .ADDRESS 'AFPRO
00000000 001BC .LONG 0
00000000 001C0 .ADDRESS P.AAC
00000000 001C4 .ADDRESS P.AAD
06 001C8 .BYTE 6
04 001C9 .BYTE 4
0064 001CA .WORD 100
00000000 001CC .LONG 0
0000 001D0 .WORD 0
00 001D2 .BYTE 0
00 001D3 .BYTE 0
00000000 001D4 .LONG 0
00000000 001D8 .LONG 0
0000 001DC .WORD 0
00 001DE .BYTE 0
00 001DF .BYTE 0
00000000 001E0 .LONG 0
01 001E4 NAF R AB: .BYTE 1
44 001E5 .BYTE 68
0000 001E6 .WORD 0

```

.....

CV  
VO  
.....

00000000	001E8	.LONG	0
00000000	001EC	.LONG	0
00000000	001F0	.LONG	0
0000	001F4	.WORD	0[3]
0000	001FA	.WORD	0
00000000	001FC	.LONG	0
0000	00200	.WORD	0
00	00202	.BYTE	0
00	00203	.BYTE	0
0000	00204	.WORD	0
0064	00206	.WORD	100
00000000	00208	.LONG	0
00000000	0020C	.ADDRESS	NAFBUF
00000000	00210	.LONG	0
00000000	00214	.LONG	0
00	00218	.BYTE	0
00	00219	.BYTE	0
00	0021A	.BYTE	0
00	0021B	.BYTE	0
00000000	0021C	.LONG	0
00000000	00220	.ADDRESS	NAFFAB
00000000	00224	.LONG	0
15	00228	ONFKEY1: .BYTE	21
4C	00229	.BYTE	76
0000	0022A	.WORD	0
00000000	0022C	.LONG	0
00	00230	.BYTE	0
00	00231	.BYTE	0
00	00232	.BYTE	0
00	00233	.BYTE	0
00	00234	.BYTE	0
00	00235	.BYTE	0
00000000	00236	.LONG	0
00	0023A	.BYTE	0
00	0023B	.BYTE	0
00	0023C	.BYTE	0
00	0023D	.BYTE	0
00	0023E	.BYTE	0
01	0023F	.BYTE	1
0000	00240	.WORD	0
0000	00242	.WORD	0
0000	00244	.WORD	0
0000	00246	.WORD	0
0000	00248	.WORD	0
0000	0024A	.WORD	0
0000	0024C	.WORD	0
0000	0024E	.WORD	0
0000	00250	.WORD	0
0000	00252	.WORD	0
0000	00254	.WORD	0
00	00256	.BYTE	0
00	00257	.BYTE	C
00	00258	.BYTE	0
00	00259	.BYTE	0
00	0025A	.BYTE	0
00	0025B	.BYTE	0
00	0025C	.BYTE	0

.....

.....

00	0025D	.BYTE	0
0000	0025E	.WORD	0
00000000	00260	.LONG	0
00000000	00264	.LONG	0
00	00268	.BYTE	0
00	00269	.BYTE	0
00	0026A	.BYTE	0
00	0026B	.BYTE	0
00	0026C	.BYTE	0
00	0026D	.BYTE	0
00	0026E	.BYTE	0
00	0026F	.BYTE	0
00	00270	.BYTE	0
00	00271	.BYTE	0
	00272	.BLKB	2
15	00274	ONFKEY0: .BYTE	21
4C	00275	.BYTE	76
0000	00276	.WORD	0
00000000	00278	.ADDRESS ONFKEY1	
00	0027C	.BYTE	0
00	0027D	.BYTE	0
00	0027E	.BYTE	0
00	0027F	.BYTE	0
00	00280	.BYTE	0
00000000	00281	.BYTE	0
00	00282	.LONG	0
00	00286	.BYTE	0
00	00287	.BYTE	0
00	00288	.BYTE	0
00	00289	.BYTE	0
00	0028A	.BYTE	0
00	0028B	.BYTE	0
0000	0028C	.WORD	0
0000	0028E	.WORD	0
0000	00290	.WORD	0
0000	00292	.WORD	0
0000	00294	.WORD	0
0000	00296	.WORD	0
0000	00298	.WORD	0
0000	0029A	.WORD	0
0000	0029C	.WORD	0
0000	0029E	.WORD	0
0000	002A0	.WORD	0
00	002A2	.BYTE	0
00	002A3	.BYTE	0
00	002A4	.BYTE	0
00	002A5	.BYTE	0
00	002A6	.BYTE	0
00	002A7	.BYTE	0
00	002A8	.BYTE	0
00	002A9	.BYTE	0
0000	002AA	.WORD	0
00000000	002AC	.LONG	0
00000000	002B0	.LONG	0
00	002B4	.BYTE	0
00	002B5	.BYTE	0
00	002B6	.BYTE	0

.....

.....

```

00 002B7 .BYTE 0
00 002B8 .BYTE 0
00 002B9 .BYTE 0
00 002BA .BYTE 0
00 002BB .BYTE 0
00 002BC .BYTE 0
00 002BD .BYTE 0
00 002BE .BLKB 2
03 002C0 ONFFAB: .BYTE 3
50 002C1 .BYTE 80
0000 002C2 .WORD 0
00000000 002C4 .LONG 0
00000000 002C8 .LONG 0
00000000 002CC .LONG 0
00000000 002D0 .LONG 0
0000 002D4 .WORD 0
02 002D6 .BYTE 2
00 002D7 .BYTE 0
00000000 0C2D8 .LONG 0
00 002DC .BYTE 0
00 002DD .BYTE 0
00 002DE .BYTE 0
02 002DF .BYTE 2
00000000 002E0 .LONG 0
00000000 002E4 .ADDRESS ONFKEY0
00000000 002E8 .LONG 0
00000000 002EC .ADDRESS P.AAE
00000000 002F0 .ADDRESS P.AAF
06 002F4 .BYTE 6
04 002F5 .BYTE 4
0000 002F6 .WORD 0
00000000 002F8 .LONG 0
0000 002FC .WORD 0
00 002FE .BYTE 0
00 002FF .BYTE 0
00000000 00300 .LONG 0
00000000 00304 .LONG 0
0000 00308 .WORD 0
00 0030A .BYTE 0
00 0030B .BYTE 0
00000000 0030C .LONG 0
01 00310 ONFRAB: .BYTE 1
44 00311 .BYTE 68
0000 00312 .WORD 0
00000000 00314 .LONG 0
00000000 00318 .LONG 0
00000000 0031C .LONG 0
0000# 00320 .WORD 0[3]
0000 00326 .WORD 0
00000000 00328 .LONG 0
0000 0032C .WORD 0
00 0032E .BYTE 0
00 0032F .BYTE 0
001E 00330 .WORD 30
0000 00332 .WORD 0
00000000 00334 .ADDRESS ONFBUF
00000000 00338 .LONG 0

```

.....

.....

```

00000000 0033C .LONG 0
00000000 00340 .LONG 0
      00 00344 .BYTE 0
      00 00345 .BYTE 0
      00 00346 .BYTE 0
      00 00347 .BYTE 0
00000000 00348 .LONG 0
00000000 0034C .ADDRESS ONFFAB
00000000 00350 .LONG 0

```

```

.EXTRN LIB$PUT OUTPUT, SYSS$OPEN
.EXTRN SYSS$CONNECT, SYSS$CREATE
.EXTRN SYSS$GET, SYSS$PUT
.EXTRN SYSS$CLOSE

```

```
.PSECT $CODE$,NOWRT,2
```

```

      03FC 00000 MAIN: .WORD Save R2,R3,R4,R5,R6,R7,R8,R9      : 0245
59 00000000G 00 9E 00002 MOVAB SYSS$CLOSE, R9
58 00000000G 00 9E 00009 MOVAB SYSS$CONNECT, R8
57 0000' 00 9E 00010 MOVAB ONFFAB, R7
      57 DD 00015 PUSHL R7      : 0254
00000000G 00 01 FB 00017 CALLS #1, SYSS$OPEN
6A 50 E9 0001E BLBC RMSERR, 5$      : 0255
      50 A7 9F 00021 PUSHAB ONFRAB
68 01 FB 00024 CALLS #1, SYSS$CONNECT
61 50 E9 00027 BLBC RMSERR, 5$
      D2 A7 B5 0002A TSTW ONFKEY0+30      : 0260
40 8F E2 A7 91 0002F BNEQ 1$
      14 12 00034 BNEQ 1$
0040 8F 86 A7 B1 00036 CMPW ONFKEY1+30, #64      : 0261
      0C 12 0003C BNEQ 1$
20 96 A7 91 0003E CMPB ONFKEY1+46, #32      : 0262
      06 12 00042 BNEQ 1$
      0000' CF 9F 00044 PUSHAB P.AAG      : 0266
      1B 11 00048 BRB 3$
      D2 A7 B5 0004A 1$: TSTW ONFKEY0+30      : 0269
      12 12 0004D BNEQ 2$
12 E2 A7 91 0004F CMPB ONFKEY0+46, #18      : 0270
      0C 12 00053 BNEQ 2$
12 86 A7 B1 00055 CMPW ONFKEY1+30, #18      : 0271
      06 12 00059 BNEQ 2$
0C 96 A7 91 0005B CMPB ONFKEY1+46, #12      : 0272
      13 13 0005F BEQL 4$
00000000G 00 0000' CF 9F 00061 2$: PUSHAB P.AAI      : 0275
50 10000000 01 FB 00065 3$: CALLS #1, LIB$PUT_OUTPUT
      04 00073 MOVL #268435456, -R0      : 0276
50 10 A7 02 C7 00074 4$: RET
      FE6C C7 10 B740 9E 00079 4$: DIVL3 #2, ONFFAB+16, R0
      FED4 C7 9F 00080 MOVAB @ONFFAB+16[R0], NAFALL+16
00000000G 00 01 FB 00084 PUSHAB NAFFAB      : 0282
6C 50 E9 0008E 5$: CALLS #1, SYSS$CREATE
      FF24 C7 9F 0008E 5$: BLBC RMSERR, 10$
68 01 FB 00092 PUSHAB NAFRAB
      43 11 00095 BRB 7$      : 0283

```

				50	A7 9F 00097 6\$:	PUSHAB ONFRAB		0285
		00000000G	00		01 FB 0009A	CALLS #1, SYSSGET		
			56		50 D0 000A1	MOVL R0, RMSERR		
0064	8F		37		56 E9 000A4	BLBC RMSERR, 8\$		
		00	6E	FD40	00 2C 000A7	MOVCS #0, (SP), #0, #100, NAFBUF		0288
	20				C7 000AE			
		20	FDA4	C7	06 2C 000B1	MOVCS #6, ONFBUF, #32, #32, NAFBUF		0289
	20			FD40	C7 000B8			
		20	FDA4	C7	0C 2C 000BB	MOVCS #12, ONFBUF+6, #32, #32, NAFBUF+32		0290
	20			FD60	C7 000C2			
		20	FDB6	C7	0C 2C 000C5	MOVCS #12, ONFBUF+18, #32, #32, NAFBUF+64		0291
				FD80	C7 000CC			
				FF24	C7 9F 000CF	PUSHAB NAFRAB		0292
		00000000G	00		01 FB 000D3	CALLS #1, SYSSPUT		
			BA		50 EB 000DA 7\$:	BLBS RMSERR, 6\$		
		0001827A	8F		04 000DD	RET		
			50		56 D1 000DE 8\$:	CMPL RMSERR, #98938		0294
					04 13 000E5	BEQL 9\$		
					56 D0 000E7	MOVL RMSERR, R0		
					04 000EA	RET		
				FED4	C7 9F 000E9 9\$:	PUSHAB NAFFAB		0295
			69		01 FB 000EF	CALLS #1, SYSSCLOSE		
			05		50 E9 000F2	BLBC RMSERR, 10\$		
					57 DD 000F5	PUSHL R7		0296
			69		01 FB 000F7	CALLS #1, SYSSCLOSE		
					04 000FA 10\$:	RET		0297

: Routine Size: 251 bytes, Routine Base: \$CODE\$ + 0000

```

: 255      0298 1
: 256      0299 1 END
: 257      0300 0 ELUDOM

```

OBJECT SUMMARY

Name	Bytes	Attributes
\$OWNS	852	NOVEC, WRT, RD, NOEXE, NOSHR, LCL, REL, CON, NOPIC, ALIGN(2)
\$SPLITS	164	NOVEC, NOWRT, RD, NOEXE, NOSHR, LCL, REL, CON, NOPIC, ALIGN(2)
\$CODE\$	251	NOVEC, NOWRT, RD, EXE, NOSHR, LCL, REL, CON, NOPIC, ALIGN(2)

Library Statistics

File	Total	Symbols Loaded	Percent	Pages Mapped	Processing Time
_\$255\$DUA28:[SYSLIB]LIB.L32;1	18619	77	0	1000	00:01.9

CVTNAF  
V04-000

Convert Network Proxy File

M 14  
16-Sep-1984 02:14:17  
14-Sep-1984 13:21:08

VAX-11 Bliss-32 V4.0-742  
DISK\$VMSMASTER:[LAF.SRC]CVTNAF.B32;1 Page 14 (1)

COMMAND QUALIFIERS

BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS\$:CVTNAF/OBJ=OBJ\$:CVTNAF MSRC\$:CVTNAF/UPDATE=(ENHS:CVTNAF)

: Size: 251 code + 1016 data bytes  
: Run Time: 00:17.3  
: Elapsed Time: 00:44.7  
: Lines/CPU Min: 1039  
: Lexemes/CPU-Min: 44979  
: Memory Used: 173 pages  
: Compilation Complete

CV  
V0  
20

64

6E  
65



The image displays a 10x12 grid of 120 terminal windows, each showing a different screen from the VAX/VMS V4.0 software suite. The screens are arranged in a regular grid and contain various system utilities, command-line interfaces, and data displays. Some of the prominent screens include:

- AUTHORIZE CLD**: A screen for authorizing classes.
- ONFDEF SDL**: A screen for defining ONFDEF SDL.
- LAF**: A screen for LAF.
- AUTHORIZE MAP**: A screen for authorizing maps.
- CUTUAF MAP**: A screen for cutting UAF maps.
- LAFREQ SDL**: A screen for LAFREQ SDL.
- YFDRIVER LIS**: A screen for YFDRIVER LIS.
- CUTNAF LIS**: A screen for cutting NAF LIS.
- ONFDEF SDL**: Another instance of the ONFDEF SDL screen.
- CUTNAF MAP**: A screen for cutting NAF maps.
- CUTNAF LIS**: Another instance of the CUTNAF LIS screen.

Each screen typically features a title bar, a header section, and a main content area with text, tables, or command prompts. The overall appearance is that of a multi-user terminal environment from the early 1980s.