

[OpenVMS] How to Read Mailbox, MBAn:, Data Without Removing the Messages  
Last Technical Review: 8-DEC-1992  
Copyright (c) 1997, 1998 Compaq Computer Corporation. All rights reserved.

PRODUCT: DIGITAL OpenVMS VAX, All Versions  
COMPONENT: Driver, MBDRIVER  
SOURCE: Compaq Computer Corporation

#### OVERVIEW:

Programmers may use mailboxes for inter-process communication. If you need to troubleshoot such an application, you may need to determine the content of a mailbox.

For example, one process sends a message that another process reads to determine its next task. If the second process is not performing the task, then perhaps the mailbox message is incorrect or missing. You can determine this only by reading the mailbox.

You might also need to read a mailbox when analyzing a crash dump.

However, reading the mailbox removes the message, thus interfering with troubleshooting.

#### QUESTION:

How can you read the contents of a mailbox without removing the messages?

#### ANSWER:

You can use the System Dump Analyzer (SDA) to read the contents of a mailbox without removing the messages from the mailbox.

#### Note:

You need CMKRNL privilege to invoke SDA with ANALYZE/SYSTEM to read a mailbox on the running system. If you need to read a mailbox in a crash dump, you need READ access to the dumpfile.

1. First, create a command procedure which will define the I/O buffer (IOB), extract the length of the message, and examine the mailbox as described below. The Control-Z (^Z) returns you to DCL. (You could also invoke an editor if desired.)

There are 2 versions of the command procedure. One for Pre-5.5 systems and one for systems running VMS versions 5.5 and later.

```
(File for Pre 5.5 systems)
$CREATE SEE_MBX.COM
DEFINE IOB=@IOB
DEFINE LEN=(@(IOB+C)@10)@-10
EXAMINE IOB+16;LEN
^Z
$
```

```

(File for 5.5 and later systems)
$CREATE SEE_MBX.COM
DEFINE IOB=@IOB
DEFINE LEN=(@(IOB+20)@10)@-10
EXAMINE IOB+24;LEN
^Z
$

```

- Next, invoke SDA with \$ANALYZE/SYSTEM (to look at a mailbox on the running system) or \$ANALYZE/CRASH (to look at a mailbox in a crash dump). This example is from a running system, but you would use the same SDA commands in a crash dump.

```
$ANALYZE/SYSTEM
```

```
VAX/VMS System Analyzer
```

- Define data structures and symbols for SDA:

```
SDA> READ SYS$SYSTEM:SYSDEF
```

The analysis may implicate a mailbox device; for example, if a process is in RWMBX state (Resource Wait for MailBoX), you can use the SDA command SHOW PROCESS/CHANNELS to get the mailbox device name - MBA2 in this example.

- Show the mailbox device you want to read; this step defines the UCB (Unit Control Block) symbol and address:

```

SDA> SHOW DEVICE MBA2
I/O data structures
-----
MBA2                Unknown                UCB address:  8018A878
                                     ^
                                     |
:                                     +-----+
:                                     |
:                                     SDA will use this address
:                                     in the next command

```

- Define the symbol IOB=UCB; this will be used by the command procedure.

```
SDA> DEFINE IOB=UCB
```

- Finally, activate the SEE\_MBX.COM procedure to see the first message.

```

SDA> @SEE_MBX      ! read the first message in MBA2:
00120040 00000040 00000000 00000008  .....@...@...  80380B06
20204957 20202020 20595255 424C4957 WILBURY      WI  80380B16
00020000 00000001 00070000 20202020  .....  80380B26
6C400000 00900004 00500200 01010003  .....P.....@l  80380B36
20202020 45464646 46464646 C1B4AA68 h^a^A..FFFFFFE  80380B46
20202059 5255424C 49572020 20202020 WILBURY      80380B56
20202059 5255424C 49572020 20202020 WILBURY      80380B66
0000000C 00122020 20202020 50492020 IP           .....  80380B76
41414141 4141000B 000A0000 00000800  .....AAAAAA  80380B86
EB600000 04200000 00000000 AAAA004D M.^a.....`e  80380B96

```

7. Continue to activate SEE\_MBX.COM until no more messages appear:

```
SDA> @SEE_MBX      ! second message in MBA2:
00010004 00000040 00000000 00000008 .....@..... 802248C6
6F6C3C00 20202020 20204644 414A4C3B ;LJADF      .<lo 802248D6
00040000 00000001 00070000 3E6E6967 gin>..... 802248E6
04A00000 00920004 00500500 01010003 .....P..... 802248F6
20202020 45464646 46464646 C1B4AA68 h^`Á..FFFFFFE 80380B46
20202020 3A314154 525F2020 20202020      _RTA1:      80224916
20202020 4644414A 4C3B2020 20202020      ;LJADF      80224926
7084000C 00103E6E 69676F6C 3C002020      .<login>.....p 80224936
444A4C3B 413B000B 000B0000 00000000 .....;A;LJD 80224946
48000A00 093A3141 54525F00 0A000546 F...._RTA1:....H 80224956
20202059 5255424C 49572020 20202020      WILBURY      80380B56
```

```
SDA> @SEE_MBX      ! When the procedure returns the SDA prompt
                  ! instead of a message, there are no more
SDA> EXIT          ! messages to read, and you can exit from SDA.
```