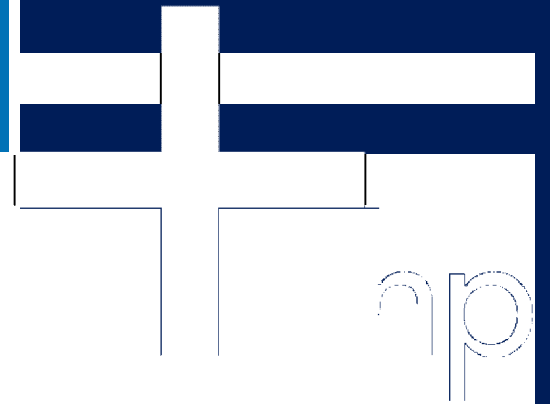




# Using System Service Logging Technical Update

**Ruth Goldenberg**  
**OpenVMS Engineering, HP**

© 2006 Hewlett-Packard Development Company, L.P.  
The information contained herein is subject to change without notice



# Overview

- Mechanism to record information about system service activity in a process
  - Exec and kernel mode services
  - Execlet and privileged shareable image services
  - For a specific process
  - Alpha and Itanium Version 8.2 and later versions
- Main goal was troubleshooting

# Recording Information

- For each service
  - Service identification
  - Location of service request – image and offset
  - Access mode of requestor
  - Service arguments, optionally, depending on whether process has privilege
  - Time stamp
  - Completion status
  - CPU ID, kernel thread ID, and Pthread ID
  - SSLOG entry number

# Enabling Logging

- Systemwide first, through dynamic SYSGEN parameter SYSSER\_LOGGING, 1 by default
- For a particular process, through DCL command, e.g.

```
$SET PROC/SSL=(STATE=ON,COUNT=4)
```

- By default, affects current process
- /NAME or /ID can target another process
- Also can create a process with logging
  - \$RUN <process>/SSLOG\_ENABLE[=(COUNT=x)]
  - Spawned process inherits parent's logging state

# P2 Space Buffers

- Information recorded
  - In a ring of P2 space buffers initially
  - Full buffer written to file SSLOG.DAT in current default disk and directory
- Buffers
  - Number specified by COUNT
  - Are pageable and thus charged against PGFLQUOTA
  - Not deallocated until process deletion
  - Theoretically, the size in bytes specified by SIZE but size is really always  $FE00_{16}$

# Disabling Logging

- Through DCL command
  - \$ SET PROC/SSL=(STATE=OFF) or
  - \$ SET PROC/SSL=(STATE=UNLOAD)
- STATE=OFF stops logging, but it can be re-enabled again
- STATE=UNLOAD stops logging and closes SSLOG.DAT
- Enable/disable done by \$SET\_PROCESS\_PROPERTIES system service
  - PPROP\$C\_SS\_LOG\_ENABLE
  - PPROP\$C\_SS\_LOG\_DISABLE
  - PPROP\$C\_SS\_LOG\_UNLOAD

# Basic Mechanisms

- Enabling logging
  - Create kernel-owned P2 va and init it
  - Allocate space and write a start message
  - Queue supervisor mode AST to create the log file
  - After file is opened, set `PCB$M_SS_LOGGING_ENABLE`
- Logging a system service request
  - Hooks in the change mode dispatchers
  - At entry, increment entry number, allocate buffer space, record info
  - At exit, record service status and current entry number
  - Full buffer written asynchronously to `SSLOG.DAT` and switch to next buffer in ring

## Basic Mechanisms (continued)

- Stopping logging
  - Clears PCB\$M\_SS\_LOGGING\_ENABLE
  - Writes stop message in buffer
- Restarting logging
  - Sets PCB\$M\_SS\_LOGGING\_ENABLE
  - Writes start message in buffer
- Unloading logging
  - Clears PCB\$M\_SS\_LOGGING\_ENABLE
  - Writes stop message and filler record
  - Writes buffers to SSLOG.DAT
  - Closes SSLOG.DAT



# Limitations

- Mode of caller services not logged at all
- Real service requestor obscured in some cases
  - Mode of caller services calling internal services
  - Services called through `AMAC$EMUL_CALL_NATIVE`
  - RMS services
- Service status (and entry) not always available
  - If service exits at above IPL 2
  - If buffer written to disk before service completes

# Displaying Logged Information

- Through DCL command ANALYZE/SSL <file>
  - /WIDE, /BRIEF, or /NORMAL (default)
  - /FULL to get error statuses interpreted and sequence numbers
  - /SELECT to see subset of entries
  - /STATISTICS to see summary
- Example brief output

```
START 1.2 2040022e HERE ALPHA !22-MAY-2006 18:27:38.80
      TOM::GRISWOLD 2 65024
SYS$EXIT_INT sts: ----- acmode: U !18:27:39.17
      image: IMAGE_MANAGEMENT+00014d28 argct: 01
SYS$RMSRUNDN sts: 00010001 acmode: S !18:27:39.17
      image: DCL+00083664 argct: 02
SYS$DCLAST sts: 00000001 acmode: E !18:27:39.17
      image: RMS+0005c354 argct: 03
SYS$RMS_CLOSE sts: 00010001 acmode: E !18:27:39.17
      image: RMS+00055e28 argct: 03
```



# Displaying Logged Information (cont.)

- Example full output

```
START version: 1.2 process: 2040022e HERE !22-MAY-2006 18:27:38.80
      username: GRISWOLD node: TOM platform: ALPHA
      buffer count: 2 size: 65024 start_flags: 00000003
SYS$EXIT_INT sts: ----- acmode: U !18:27:39.17
      image: IMAGE_MANAGEMENT+00014d28 argct: 01
      arg 1:0000000010000001
      entry number: 00000002 number at completion: 00000000
      cpu id: 000 kernel thread ID: 0000 Pthread ID: 1
[...]
```

```
SYS$DASSGN acmode: K !18:27:39.17
      sts: %SYSTEM-S-NORMAL, normal successful completion
      image: IMAGE_MANAGEMENT+0000fe6c argct: 01
      arg 1:000000000000000b0
      entry number: 00000018 number at completion: 00000019
      cpu id: 000 kernel thread ID: 0000 Pthread ID: 0
```

```
SYS$QIO acmode: K !18:27:39.17
      sts: %SYSTEM-S-NORMAL, normal successful completion
      image: IO_ROUTINES_MON+0002d3c4 argct: 12
      arg 1:00000000000000080 2:000000000000000b0 3:00000000000000034
      arg 4:000000007ff87e28 5:00000000000000000 6:00000000000000000
      arg 7:000000007ff87e10 8:00000000000000000 9:00000000000000000
      arg 10:00000000000000000 11:00000000000000000 12:00000000000000000
      entry number: 00000019 number at completion: 00000019
      cpu id: 000 kernel thread ID: 0000 Pthread ID: 0
```



# ANALYZE/SSLOG/STAT

IPL31> ana/ssl/stat

START version: 1.1 process: 26800426 GUY !24-NOV-2004 12:43:52.54  
username: GUY node: IPL31 platform: IA64  
buffer count: 6 size: 65024 start\_flags: 00000003

Service	Count	User	Super	Exec	Kernel	Rate/sec
-----	-----	-----	-----	-----	-----	-----
SYS\$RMS_GET	185874	0	185874	0	0	939.7
SYS\$\$ENQ	111680	0	0	1	111679	564.6
SYS\$\$DEQ	109722	0	0	1	109721	554.7
SYS\$\$QIO	76406	0	36996	36622	2788	386.3
SYS\$RMS_FIND	37180	0	37180	0	0	188.0
SYS\$PERSONA_EXTENSION_LOOKUP	37146	0	0	0	37146	187.8
SYS\$GETDVI	37018	0	37000	18	0	187.1
SYS\$ASSIGN_LOCAL	36970	0	36952	18	0	186.9
SYS\$FILESCAN	36856	0	36856	0	0	186.3
[. . .]						



# ANALYZ/SSL/STAT/SELE=IMAGE

• IPL31> ana/ssl/stat/select=image=dcl

• START version: 1.1 process: 26800426 GUY !24-NOV-2004 12:43:52.54  
• username: GUY node: IPL31 platform: IA64  
• buffer count: 6 size: 65024 start\_flags: 00000003

Service	Count	User	Super	Exec	Kernel	Rate/sec
-----	-----	-----	-----	-----	-----	-----
• SYS\$FILESCAN	36856	0	36856	0	0	186.3
• SYS\$DASSGN	36778	0	36778	0	0	185.9
• SYS\$DCLEXH	1	0	1	0	0	0.0
• SYS\$IMGACT	1	0	1	0	0	0.0
• SYS\$SETEXV	1	1	0	0	0	0.0
• SYS\$RMSRUNDN	1	0	1	0	0	0.0



# ANALY/SSL/STAT/SELECT=ACCESS

• IPL31> ana/ssl/stat/select=access=user

• START version: 1.1 process: 26800426 GUY !24-NOV-2004 12:43:52.54  
• username: GUY node: IPL31 platform: IA64  
• buffer count: 6 size: 65024 start\_flags: 00000003

Service	Count	User	Super	Exec	Kernel	Rate/sec
-----	-----	-----	-----	-----	-----	-----
• SYS\$EXPREG	3	3	0	0	0	0.0
• SYS\$GETJPI	2	2	0	0	0	0.0
• SYS\$SET_PROCESS_PROPERTIESW	1	1	0	0	0	0.0
• SYS\$SETEXV	1	1	0	0	0	0.0
• SYS\$IMGACT	1	1	0	0	0	0.0
• SYS\$GETSYI	1	1	0	0	0	0.0
• SYS\$EXIT_INT	1	1	0	0	0	0.0

• IPL31>ana/ssl/select=access=user ! Looking at all the entries

• SYS\$GETSYI sts: 00000001 acmode: U !12:47:10.34  
• image: SYSTEM\_PRIMITIVES\_MIN+00199920 argct: 07  
• arg 1:000000000000000000000000 2:000000000000000000000000 3:000000000000000000000000  
• arg 4:000000007ad178b0 5:000000007ad178c0 6:000000000000000000000000  
• arg 7:000000000000000000000000  
• entry number: 0065CF6A number at completion: 0065CF6A



# One Example of its Use

- Customer reported reproducible problem with DCL procedures after upgrading to V8.2
- Problem reduced to SEARCH returning SS\$\_IVLOGNAM if 16-character logical name used
- Engineer reproduced the problem with logging enabled and looked at the log for status 154<sub>16</sub>

```
IPL8> ana/ssl/sele=status=154
START  version: 1.2 process: 34a0042f _VTA1:           !16-APR-2006 19:53:31.37
       username: XYZ                               node: IPL8   platform: IA64
       buffer count: 2   size: 65024   start_flags: 00000003
missing entry numbers: curr:1166 prev:1037

SYS$GETDVI                               sts: 00000154  acmode: U !19:53:34.93
  image:                                SYSTEM_PRIMITIVES+001a8250  argct: 08
  arg 1:0000000000000000  2:0000000000000000  3:000000007ac31b00
  arg 4:000000007ac31ad8  5:0000000000000000  6:0000000000000000
  arg 7:0000000000000000  8:0000000000000000
  entry number: 000006DB   number at completion: 000006DB
  cpu id:    000   kernel thread ID: 0000   Pthread ID:0
```

# Another Example

- TCP/IP Network Time Protocol (NTP) server would not start up on two different clusters— log extract:

```
2 Dec 19:01:10 ntp[538970507]: no IPv6 interfaces found
getconfig: Couldn't open <SYS$SPECIFIC:[TCPIP$NTP]TCPIP$NTP.CONF>
 2 Dec 19:01:10 ntp[538970507]: getconfig: Couldn't open <SYS$SPECIFIC:[TCPIP$NTP]TCPIP$NTP.CONF>
2 Dec 19:01:11 ntp[538970507]: adj_sysstime: not owner
2 Dec 19:01:12 ntp[538970507]: adj_sysstime: not owner
```

After support engineer tried SET WATCH/FILE and enabling all privileges without success, he enabled logging:

```
SYS$CMKRNL                sts: 00000194  acmode: U
  image:                  TCPIP$NTP+000780a0  argct:02
  arg  1:0000000000018f58  2:000000007adc15e8
  entry number: 00000BA7   number at completion: 00000BA8
SYS$LKWSET                sts: 00000194  acmode: K
  image:                  TCPIP$NTP+00078158  argct:03
  arg  1:0000000000018ef0  2:0000000000000000  3:0000000000000000
  entry number: 00000BA8   number at completion: 00000BA8

[...]
SYS6$ exit %x194
%SYSTEM-F-LKWSETFUL, locked portion of working set is full
```



# System Service Logging at work

- When we started using ANALYZE/SSLOG internally we found some interesting “features”..

```
• BLUSKY> ty loop.com
• $ start=f$time()
• $ i=25000
• $ loop:
• $ if i .eq. 0 then goto exit
• $ i=i-1
• $ goto loop
• $ exit:
• $ end=f$time()
• $ write sys$output f$delta(start,end)
```

- Alpha personal workstation without the fix

```
• BLUSKY> @loop
• 0 00:00:06.61
```

- Same system with the new DCL image:

```
• BLUSKY> @loop
• 0 00:00:04.70
```

# Caveats

- System service log can take a lot of disk space
- If logging is stopped and restarted, the existing buffers are used
- Must analyze log on same architecture as one on which it was created
- No service name for privileged shareable image service – just image name and CHMx code
- Do not be misled by SYSTEM-S-ACCVIO

```
SYS$SETEF                               acmode: U   image:
      LIBRTL+00091e6c                    !16:24:12.71
      sts: %SYSTEM-S-ACCVIO, access violation, reason mask=!XB, virtual address=!XH, PC=!XH, PS=!XL
argct:01  1:000000000000000080
[...]
```



# For Further Information

- OpenVMS System Analysis Tools Manual



i n v e n t