```
**********************************************************
   11896 Mon Aug 26 04:14:44 2019
new/usr/src/man/man1m/fmadm.1m
11621 fmadm and fmstat document privileges incorrectly
**********************************************************
'\" te
.\" Copyright (c) 2008, Sun Microsystems, Inc. All Rights Reserved.
.\" Copyright 2012 Joshua M. Clulow <josh@sysmgr.org>
.\" Copyright 2019 Peter Tribble
.\" The contents of this file are subject to the terms of the Common Development
.\" You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
.\" When distributing Covered Code, include this CDDL HEADER in each file and in
.TH FMADM 1M "Aug 26, 2019"
.TH FMADM 1M "Oct 22, 2008"
.SH NAME
fmadm \- fault management configuration tool
.SH SYNOPSIS
.LP
.nf
\fBfmadm\fR [\fB-q\fR] [\fIsubcommand\fR [\fIarguments\fR]]
.fi

.SH DESCRIPTION
.sp
.LP
The \fBfmadm\fR utility can be used by administrators and service personnel to
view and modify system configuration parameters maintained by the Fault
Manager, \fBfmd\fR(1M). \fBfmd\fR receives telemetry information relating to
problems detected by the system software, diagnoses these problems, and
initiates proactive self-healing activities such as disabling faulty
components.
.sp
.LP
\fBfmadm\fR can be used to:
.RS +4
.TP
.ie t \(bu
.el o
view the set of diagnosis engines and agents that are currently participating
in fault management,
.RE
.RS +4
.TP
.ie t \(bu
.el o
view the list of system components that have been diagnosed as faulty, and
.RE
.RS +4
.TP
.ie t \(bu
.el o
perform administrative tasks related to these entities.
.RE
.sp
.LP
The Fault Manager attempts to automate as many activities as possible, so use
of \fBfmadm\fR is typically not required. When the Fault Manager needs help
from a human administrator, it produces a message indicating its needs. It also
refers you to a knowledge article on http://illumos.org/msg/. This web site
might ask you to use \fBfmadm\fR or one of the other fault management utilities
to gather more information or perform additional tasks. The documentation for
\fBfmd\fR(1M), \fBfmdump\fR(1M), and \fBfmstat\fR(1M) describe more about tools
to observe fault management activities.
.sp
.LP
The \fBfmadm\fR utility requires the user to possess the \fBPRIV_SYS_ADMIN\fR
```



```
privilege. See \fBprivileges\fR(5). The \fBfmadm\fR \fBload\fR subcommand
requires that the user possess all privileges.
The \fBfmadm\fR utility requires the user to possess the \fBSYS_CONFIG\fR
privilege. Refer to the \fI\fR for more information about how to configure
privileges. The \fBfmadm\fR \fBload\fR subcommand requires that the
user possess all privileges.
.SS "SUBCOMMANDS"
.sp
.LP
\fBfmadm\fR accepts the following subcommands. Some of the subcommands accept
or require additional options and operands:
.sp
.ne 2
.na
\fB\fBfmadm acquit\fR \fIfmri\fR \fB|\fR \fIlabel\fR [\fIuuid\fR]\fR
.ad
.sp .6
.RS 4n
Notify the Fault Manager that the specified resource is not to be considered to
be a suspect in the fault event identified by \fIuuid\fR, or if no UUID is
specified, then in any fault or faults that have been detected. The \fBfmadm
acquit\fR subcommand should be used only at the direction of a documented
repair procedure. Administrators might need to apply additional commands to
re-enable a previously faulted resource.
.RE

.sp
.ne 2
.na
\fB\fBfmadm acquit\fR \fIuuid\fR\fR
.ad
.sp .6
.RS 4n
Notify the Fault Manager that the fault event identified by \fIuuid\fR can be
safely ignored. The \fBfmadm acquit\fR subcommand should be used only at the
direction of a documented repair procedure. Administrators might need to
apply additional commands to re-enable any previously faulted resources.
.RE

.sp
.ne 2
.na
\fB\fBfmadm config\fR\fR
.ad
.sp .6
.RS 4n
Display the configuration of the Fault Manager itself, including the module
name, version, and description of each component module. Fault Manager modules
provide services such as automated diagnosis, self-healing, and messaging for
hardware and software present on the system.
.RE

.sp
.ne 2
.na
\fB\fBfmadm faulty\fR [\fB-afgiprsv\fR] [\fB-n\fR \fImax\fR] [\fB-u\fR
\fIuid\fR]\fR
.ad
.sp .6
.RS 4n
Display status information for resources that the Fault Manager currently
believes to be faulty.
.sp
The following options are supported:
.sp
.ne 2
```

```
.na
\fB\fB-a\fR\fR
.ad
.RS 10n
Display all faults. By default, the \fBfmadm faulty\fR command only lists
output for resources that are currently present and faulty. If you specify the
\fB-a\fR option, all resource information cached by the Fault Manager is
listed, including faults which have been automatically corrected or where no
recovery action is needed. The listing includes information for resources that
might no longer be present in the system.
.RE

.sp
.ne 2
.na
\fB\fB-f\fR\fR
.ad
.RS 10n
Display faulty \fBfru's\fR (Field replaceable units).
.RE

.sp
.ne 2
.na
\fB\fB-g\fR\fR
.ad
.RS 10n
Group together faults which have the same fru, class and fault message.
.RE

.sp
.ne 2
.na
\fB\fB-i\fR\fR
.ad
.RS 10n
Display persistent cache identifier for each resource in the Fault Manager.
.RE

.sp
.ne 2
.na
\fB\fB-n\fR \fImax\fR\fR
.ad
.RS 10n
If faults or resources are grouped together with the \fB-a\fR or \fB-g\fR
options, limit the output to \fImax\fR entries.
.RE

.sp
.ne 2
.na
\fB\fB-p\fR\fR
.ad
.RS 10n
Pipe output through pager with form feed between each fault.
.RE

.sp
.ne 2
.na
\fB\fB-r\fR\fR
.ad
.RS 10n
Display Fault Management Resource with their Identifier (FMRI) and their fault
management state.
```



```
.RE

.sp
.ne 2
.na
\fB\fB-s\fR\fR
.ad
.RS 10n
Display 1 line fault summary for each fault event.
.RE

.sp
.ne 2
.na
\fB\fB-u\fR \fIuid\fR\fR
.ad
.RS 10n
Only display fault with given \fBuid\fR.
.RE

.sp
.ne 2
.na
\fB\fB-v\fR\fR
.ad
.RS 10n
Display full output.
.RE

The percentage certainty is displayed if a fault has multiple suspects, either
of different classes or on different \fBfru\fR's. If more than one resource is
on the same \fBfru\fR and it is not 100% certain that the fault is associated
with the \fBfru\fR, the maximum percentage certainty of the possible suspects
on the \fBfru\fR is displayed.
.RE

.sp
.LP
The Fault Manager associates the following states with every resource for which
telemetry information has been received:
.sp
.ne 2
.na
\fB\fBok\fR\fR
.ad
.sp .6
.RS 4n
The resource is present and in use and has no known problems so far as the
Fault Manager is concerned.
.RE

.sp
.ne 2
.na
\fB\fBunknown\fR\fR
.ad
.sp .6
.RS 4n
The resource is not present or not usable but has no known problems. This might
indicate the resource has been disabled or deconfigured by an administrator.
Consult appropriate management tools for more information.
.RE

.sp
.ne 2
.na
```

```
250 \fB\fBfaulted\fR\fR
251 .ad
252 .sp .6
253 .RS 4n
254 The resource is present but is not usable because one or more problems have
255 been diagnosed by the Fault Manager. The resource has been disabled to prevent
256 further damage to the system.
257 .RE

259 .sp
260 .ne 2
261 .na
262 \fB\fBdegraded\fR\fR
263 .ad
264 .sp .6
265 .RS 4n
266 The resource is present and usable, but one or more problems have been
267 diagnosed in the resource by the Fault Manager.
268 .sp
269 If all affected resources are in the same state, this is reflected in the
270 message at the end of the list. Otherwise the state is given after each
271 affected resource.
272 .RE

274 .sp
275 .ne 2
276 .na
277 \fB\fBfmadm flush\fR \fIfmri\fR\fR
278 .ad
279 .sp .6
280 .RS 4n
281 Flush the information cached by the Fault Manager for the specified resource,
282 named by its FMRI. This subcommand should only be used when indicated by a
283 documented repair procedure. Typically, the use of this command is not
284 necessary as the Fault Manager keeps its cache up-to-date automatically. If a
285 faulty resource is flushed from the cache, administrators might need to apply
286 additional commands to enable the specified resource.
287 .RE

289 .sp
290 .ne 2
291 .na
292 \fB\fBfmadm load\fR \fIpath\fR\fR
293 .ad
294 .sp .6
295 .RS 4n
296 Load the specified Fault Manager module. \fIpath\fR must be an absolute path
297 and must refer to a module present in one of the defined directories for
298 modules. Typically, the use of this command is not necessary as the Fault
299 Manager loads modules automatically when the operating system initially boots
300 or as needed.
301 .RE

303 .sp
304 .ne 2
305 .na
306 \fB\fBfmadm unload\fR \fImodule\fR\fR
307 .ad
308 .sp .6
309 .RS 4n
310 Unload the specified Fault Manager module. Specify \fImodule\fR using the
311 basename listed in the \fBfmadm config\fR output. Typically, the use of this
312 command is not necessary as the Fault Manager loads and unloads modules
313 automatically based on the system configuration
314 .RE
```



```
316 .sp
317 .ne 2
318 .na
319 \fB\fBfmadm repaired\fR \fIfmri\fR \fB|\fR \fIlabel\fR\fR
320 .ad
321 .sp .6
322 .RS 4n
323 Notify the Fault Manager that a repair procedure has been carried out on the
324 specified resource. The \fBfmadm repaired\fR subcommand should be used only at
325 the direction of a documented repair procedure. Administrators might need
326 to apply additional commands to re-enable a previously faulted resource.
327 .RE

329 .sp
330 .ne 2
331 .na
332 \fB\fBfmadm replaced\fR \fIfmri\fR \fB|\fR \fIlabel\fR\fR
333 .ad
334 .sp .6
335 .RS 4n
336 Notify the Fault Manager that the specified resource has been replaced. This
337 command should be used in those cases where the Fault Manager is unable to
338 automatically detect the replacement. The \fBfmadm replaced\fR subcommand
339 should be used only at the direction of a documented repair procedure.
340 Administrators might need to apply additional commands to re-enable a
341 previously faulted resource.
342 .RE

344 .sp
345 .ne 2
346 .na
347 \fB\fBfmadm reset\fR [\fB-s\fR \fIserd\fR\fB]\fR \fImodule\fR\fR
348 .ad
349 .sp .6
350 .RS 4n
351 Reset the specified Fault Manager module or module subcomponent. If the
352 \fB-s\fR option is present, the specified Soft Error Rate Discrimination (SERD)
353 engine is reset within the module. If the \fB-s\fR option is not present, the
354 entire module is reset and all persistent state associated with the module is
355 deleted. The \fBfmadm reset\fR subcommand should only be used at the direction
356 of a documented repair procedure. The use of this command is typically not
357 necessary as the Fault Manager manages its modules automatically.
358 .RE

360 .sp
361 .ne 2
362 .na
363 \fB\fBfmadm rotate\fR \fBerrlog | fltlog\fR\fR
364 .ad
365 .sp .6
366 .RS 4n
367 The \fBrotate\fR subcommand is a helper command for \fBlogadm\fR(1M), so that
368 \fBlogadm\fR can rotate live log files correctly. It is not intended to be
369 invoked directly. Use one of the following commands to cause the appropriate
370 logfile to be rotated, if the current one is not zero in size:
371 .sp
372 .in +2
373 .nf
374 # \fBlogadm -p now -s 1b /var/fm/fmd/errlog\fR
375 # \fBlogadm -p now -s 1b /var/fm/fmd/fltlog\fR
376 .fi
377 .in -2
378 .sp

380 .RE
```

```
.SH OPTIONS
.sp
.LP
The following options are supported:
.sp
.ne 2
.na
\fB\fB-q\fR\fR
.ad
.RS 6n
Set quiet mode. \fBfmadm\fR does not produce messages indicating the result of
successful operations to standard output.
.RE

.SH OPERANDS
.sp
.LP
The following operands are supported:
.sp
.ne 2
.na
\fB\fIcmd\fR\fR
.ad
.RS 8n
The name of a subcommand listed in \fBSUBCOMMANDS\fR.
.RE

.sp
.ne 2
.na
\fB\fIargs\fR\fR
.ad
.RS 8n
One or more options or arguments appropriate for the selected \fIsubcommand\fR,
as described in \fBSUBCOMMANDS\fR.
.RE

.SH EXIT STATUS
.sp
.LP
The following exit values are returned:
.sp
.ne 2
.na
\fB\fB0\fR\fR
.ad
.RS 5n
Successful completion.
.RE

.sp
.ne 2
.na
\fB\fB1\fR\fR
.ad
.RS 5n
An error occurred. Errors include a failure to communicate with \fBfmd\fR or
insufficient privileges to perform the requested operation.
.RE

.sp
.ne 2
.na
\fB\fB2\fR\fR
.ad
.RS 5n
```



```
Invalid command-line options were specified.
.RE

.SH ATTRIBUTES
.sp
.LP
See \fBattributes\fR(5) for descriptions of the following attributes:
.sp

.sp
.TS
box;
c | c
l | l .
ATTRIBUTE TYPE  ATTRIBUTE VALUE
_
Interface Stability     See below.
.TE

.sp
.LP
The command-line options are Committed. The human-readable output is
not-an-interface.
.SH SEE ALSO
.sp
.LP
\fBfmd\fR(1M), \fBfmdump\fR(1M), \fBfmstat\fR(1M), \fBlogadm\fR(1M),
\fBsyslogd\fR(1M), \fBattributes\fR(5), \fBprivileges\fR(5)
\fIsyslogd\fR(1M), \fIattributes\fR(5)
.sp
.LP
\fI\fR
.sp
.LP
http://illumos.org/msg/
```

```
**********************************************************
    7384 Mon Aug 26 04:14:45 2019
new/usr/src/man/man1m/fmstat.1m
11621 fmadm and fmstat document privileges incorrectly
**********************************************************
'\" te
.\" Copyright (c) 2005, Sun Microsystems, Inc. All Rights Reserved.
.\" Copyright 2019 Peter Tribble
.\" The contents of this file are subject to the terms of the Common Development
.\"  See the License for the specific language governing permissions and limitat
.\" the fields enclosed by brackets "[]" replaced with your own identifying info
.TH FMSTAT 1M "Aug 26, 2019"
.TH FMSTAT 1M "Jun 16, 2009"
.SH NAME
fmstat \- report fault management module statistics
.SH SYNOPSIS
.LP
.nf
\fBfmstat\fR [\fB-astTz\fR] [\fB-d\fR u | d ] [\fB-m\fR \fImodule\fR] [\fIinterv
.fi

.SH DESCRIPTION
.sp
.LP
The \fBfmstat\fR utility can be used by administrators and service personnel to
report statistics associated with the Fault Manager, \fBfmd\fR(1M) and
report statistics associated with the Solaris Fault Manager, \fBfmd\fR(1M) and
its associated set of modules. The Fault Manager runs in the background on each
system. It receives telemetry information relating to problems detected
Solaris system. It receives telemetry information relating to problems detected
by the system software, diagnoses these problems, and initiates proactive
self-healing activities such as disabling faulty components.
.sp
.LP
You can use \fBfmstat\fR to view statistics for diagnosis engines and agents
that are currently participating in fault management. The documentation for
\fBfmd\fR(1M), \fBfmadm\fR(1M), and \fBfmdump\fR(1M) describes more about tools
to observe fault management activities.
.sp
.LP
If the \fB-m\fR option is present or the \fB-t\fR option is present,
\fBfmstat\fR reports any statistics kept by the specified fault management
module. The module list can be obtained using \fBfmadm config\fR.
.sp
.LP
If the \fB-m\fR option is not present, \fBfmstat\fR reports the following
statistics for each of its client modules:
.sp
.ne 2
.na
\fB\fBmodule\fR\fR
.ad
.RS 11n
The name of the fault management module, as reported by \fBfmadm config\fR.
.RE

.sp
.ne 2
.na
\fB\fBev_recv\fR\fR
.ad
.RS 11n
The number of telemetry events received by the module.
.RE

.sp
```



```
.ne 2
.na
\fB\fBev_acpt\fR\fR
.ad
.RS 11n
The number of events accepted by the module as relevant to a diagnosis.
.RE

.sp
.ne 2
.na
\fB\fBwait\fR\fR
.ad
.RS 11n
The average number of telemetry events waiting to be examined by the module.
.RE

.sp
.ne 2
.na
\fB\fBsvc_t\fR\fR
.ad
.RS 11n
The average service time for telemetry events received by the module, in
milliseconds.
.RE

.sp
.ne 2
.na
\fB\fB%w\fR\fR
.ad
.RS 11n
The percentage of time that there were telemetry events waiting to be examined
by the module.
.RE

.sp
.ne 2
.na
\fB\fB%b\fR\fR
.ad
.RS 11n
The percentage of time that the module was busy processing telemetry events.
.RE

.sp
.ne 2
.na
\fB\fBopen\fR\fR
.ad
.RS 11n
The number of active cases (open problem investigations) owned by the module.
.RE

.sp
.ne 2
.na
\fB\fBsolve\fR\fR
.ad
.RS 11n
The total number of cases solved by this module since it was loaded.
.RE

.sp
.ne 2
```

```
.na
\fB\fBmemsz\fR\fR
.ad
.RS 11n
The amount of dynamic memory currently allocated by this module.
.RE

.sp
.ne 2
.na
\fB\fBbufsz\fR\fR
.ad
.RS 11n
The amount of persistent buffer space currently allocated by this module.
.RE

.sp
.LP
The \fBfmstat\fR utility requires the user to possess the \fBPRIV_SYS_ADMIN\fR
privilege. See \fBprivileges\fR(5).
The \fBfmstat\fR utility requires the user to posses the \fBSYS_CONFIG\fR
privilege. Refer to the \fI\fR for more information about how to configure
Solaris privileges.
.SH OPTIONS
.sp
.LP
The following options are supported:
.sp
.ne 2
.na
\fB\fB-a\fR\fR
.ad
.RS 13n
Print all statistics for a module, including those kept on its behalf by
\fBfmd\fR. If the \fB-a\fR option is not present, only those statistics kept by
the module are reported. If the \fB-a\fR option is used without the \fB-m\fR
\fImodule\fR, a set of global statistics associated with \fBfmd\fR are
displayed.
.RE

.sp
.ne 2
.na
\fB\fB-d\fR \fBu\fR | \fBd\fR\fR
.ad
.RS 13n
Display a time stamp.
.sp
Specify \fBu\fR for a printed representation of the internal representation of
time. See \fBtime\fR(2). Specify \fBd\fR for standard date format. See
\fBdate\fR(1).
.RE

.sp
.ne 2
.na
\fB\fB-m\fR \fImodule\fR\fR
.ad
.RS 13n
Print a report on the statistics associated with the specified fault management
module, instead of the default statistics report. Modules can publish an
arbitrary set of statistics to help Sun service the fault management software
itself. The module statistics constitute a Private interface. See
\fBattributes\fR(5) for information on Sun's rules for Private interfaces.
Scripts should not be written that depend upon the values of fault management
module statistics as they can change without notice.
```



```
.RE

.sp
.ne 2
.na
\fB\fB-s\fR\fR
.ad
.RS 13n
Print a report on Soft Error Rate Discrimination (SERD) engines associated with
the module instead of the default module statistics report. A SERD engine is a
construct used by fault management software to determine if a statistical
threshold measured as \fIN\fR events in some time \fIT\fR has been exceeded.
The \fB-s\fR option can only be used in combination with the \fB-m\fR option.
.RE

.sp
.ne 2
.na
\fB\fB-t\fR\fR
.ad
.RS 13n
Print a report on the statistics associated with each fault management event
transport. Each fault management module can provide the implementation of one
or more event transports.
.RE

.sp
.ne 2
.na
\fB\fB-T\fR\fR
.ad
.RS 13n
Print a table of the authority information associated with each fault
management event transport. If the \fB-m\fR option is present, only transports
associated with the specified module are displayed.
.RE

.sp
.ne 2
.na
\fB\fB-z\fR\fR
.ad
.RS 13n
Omit statistics with a zero value from the report associated with the specified
fault management module. The \fB-z\fR option can only be used in combination
with the \fB-m\fR option.
.RE

.SH OPERANDS
.sp
.LP
The following operands are supported:
.sp
.ne 2
.na
\fB\fIcount\fR\fR
.ad
.RS 12n
Print only count reports, and then exit.
.RE

.sp
.ne 2
.na
\fB\fIinterval\fR\fR
.ad
```

```
.RS 12n
Print a new report every \fIinterval\fR seconds.
.RE

.sp
.LP
If no \fIinterval\fR and no \fIcount\fR are specified, a single report is
printed and \fBfmstat\fR exits. If an \fIinterval\fR is specified but no
\fIcount\fR is specified, \fBfmstat\fR prints reports every \fIinterval\fR
seconds indefinitely until the command is interrupted.
.SH EXIT STATUS
.sp
.LP
The following exit values are returned:
.sp
.ne 2
.na
\fB\fB0\fR\fR
.ad
.RS 5n
Successful completion.
.RE

.sp
.ne 2
.na
\fB\fB1\fR\fR
.ad
.RS 5n
A fatal error occurred. A fatal error could be the failure to communicate with
\fBfmd\fR(1M). It could also be that insufficient privileges were available to
perform the requested operation.
.RE

.sp
.ne 2
.na
\fB\fB2\fR\fR
.ad
.RS 5n
Invalid command-line options were specified.
.RE

.SH ATTRIBUTES
.sp
.LP
See \fBattributes\fR(5) for descriptions of the following attributes:
.sp

.sp
.TS
box;
c | c
l | l .
ATTRIBUTE TYPE  ATTRIBUTE VALUE
_
Interface Stability     See below.
.TE

.sp
.LP
The command-line options are Evolving. The human-readable default report is
Unstable. The human-readable module report is Private.
.SH SEE ALSO
\fBfmadm\fR(1M), \fBfmd\fR(1M), \fBfmdump\fR(1M), \fBattributes\fR(5),
\fBprivileges\fR(5)
```



```
.sp
.LP
\fBfmadm\fR(1M), \fBfmd\fR(1M), \fBfmdump\fR(1M), \fBattributes\fR(5)
.sp
.LP
\fI\fR
```

```
**********************************************************
   33871 Mon Aug 26 04:14:45 2019
new/usr/src/man/man5/privileges.5
11621 fmadm and fmstat document privileges incorrectly
**********************************************************
'\" te
.\" Copyright (c) 2009, Sun Microsystems, Inc. All Rights Reserved.
.\" Copyright 2015, Joyent, Inc. All Rights Reserved.
.\" Copyright 2019 Peter Tribble
.\" The contents of this file are subject to the terms of the Common Development
.\"  See the License for the specific language governing permissions and limitat
.\" the fields enclosed by brackets "[]" replaced with your own identifying info
.TH PRIVILEGES 5 "Aug 26, 2019"
.TH PRIVILEGES 5 "Feb 28, 2018"
.SH NAME
privileges \- process privilege model
.SH DESCRIPTION
.LP
In illumos, software implements a set of privileges that provide fine-grained
.LP
Solaris software implements a set of privileges that provide fine-grained
control over the actions of processes. The possession of a certain privilege
allows a process to perform a specific set of restricted operations.
.sp
.LP
The change to a primarily privilege-based security model in the
The change to a primarily privilege-based security model in the Solaris
operating system gives developers an opportunity to restrict processes to those
privileged operations actually needed instead of all (super-user) or no
privileges (non-zero UIDs). Additionally, a set of previously unrestricted
operations now requires a privilege; these privileges are dubbed the "basic"
privileges and are by default given to all processes.
.sp
.LP
Taken together, all defined privileges with the exception of the "basic"
privileges compose the set of privileges that are traditionally associated with
the root user. The "basic" privileges are "privileges" unprivileged processes
were accustomed to having.
.sp
.LP
The defined privileges are:
.sp
.ne 2
.na
\fB\fBPRIV_CONTRACT_EVENT\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to request reliable delivery of events to an event endpoint.
.sp
Allow a process to include events in the critical event set term of a template
which could be generated in volume by the user.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_CONTRACT_IDENTITY\fR\fR
.ad
.sp .6
.RS 4n
Allows a process to set the service FMRI value of a process contract template.
.RE

.sp
.ne 2
.na
```



```
\fB\fBPRIV_CONTRACT_OBSERVER\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to observe contract events generated by contracts created and
owned by users other than the process's effective user ID.
.sp
Allow a process to open contract event endpoints belonging to contracts created
and owned by users other than the process's effective user ID.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_CPC_CPU\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to access per-CPU hardware performance counters.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_DTRACE_KERNEL\fR\fR
.ad
.sp .6
.RS 4n
Allow DTrace kernel-level tracing.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_DTRACE_PROC\fR\fR
.ad
.sp .6
.RS 4n
Allow DTrace process-level tracing. Allow process-level tracing probes to be
placed and enabled in processes to which the user has permissions.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_DTRACE_USER\fR\fR
.ad
.sp .6
.RS 4n
Allow DTrace user-level tracing. Allow use of the syscall and profile DTrace
providers to examine processes to which the user has permissions.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_FILE_CHOWN\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to change a file's owner user ID. Allow a process to change a
file's group ID to one other than the process's effective group ID or one of
the process's supplemental group IDs.
.RE

.sp
```

```
.ne 2
.na
\fB\fBPRIV_FILE_CHOWN_SELF\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to give away its files. A process with this privilege runs as
if {\fB_POSIX_CHOWN_RESTRICTED\fR} is not in effect.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_FILE_DAC_EXECUTE\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to execute an executable file whose permission bits or ACL
would otherwise disallow the process execute permission.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_FILE_DAC_READ\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to read a file or directory whose permission bits or ACL would
otherwise disallow the process read permission.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_FILE_DAC_SEARCH\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to search a directory whose permission bits or ACL would not
otherwise allow the process search permission.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_FILE_DAC_WRITE\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to write a file or directory whose permission bits or ACL do
not allow the process write permission. All privileges are required to write
files owned by UID 0 in the absence of an effective UID of 0.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_FILE_DOWNGRADE_SL\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to set the sensitivity label of a file or directory to a
sensitivity label that does not dominate the existing sensitivity label.
.sp
This privilege is interpreted only if the system is configured with Trusted
```



```
Extensions.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_FILE_FLAG_SET\fR\fR
.ad
.sp .6
.RS 4n
Allows a process to set immutable, nounlink or appendonly file attributes.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_FILE_LINK_ANY\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to create hardlinks to files owned by a UID different from the
process's effective UID.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_FILE_OWNER\fR\fR
.ad
.sp .6
.RS 4n
Allow a process that is not the owner of a file to modify that file's access
and modification times. Allow a process that is not the owner of a directory to
modify that directory's access and modification times. Allow a process that is
not the owner of a file or directory to remove or rename a file or directory
whose parent directory has the "save text image after execution" (sticky) bit
set. Allow a process that is not the owner of a file to mount a \fBnamefs\fR
upon that file. Allow a process that is not the owner of a file or directory to
modify that file's or directory's permission bits or ACL.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_FILE_READ\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to open objects in the filesystem for reading. This
privilege is not necessary to read from an already open file which was opened
before dropping the \fBPRIV_FILE_READ\fR privilege.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_FILE_SETID\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to change the ownership of a file or write to a file without
the set-user-ID and set-group-ID bits being cleared. Allow a process to set the
set-group-ID bit on a file or directory whose group is not the process's
effective group or one of the process's supplemental groups. Allow a process to
set the set-user-ID bit on a file with different ownership in the presence of
\fBPRIV_FILE_OWNER\fR. Additional restrictions apply when creating or modifying
```

```
256 a setuid 0 file.
257 .RE

259 .sp
260 .ne 2
261 .na
262 \fB\fBPRIV_FILE_UPGRADE_SL\fR\fR
263 .ad
264 .sp .6
265 .RS 4n
266 Allow a process to set the sensitivity label of a file or directory to a
267 sensitivity label that dominates the existing sensitivity label.
268 .sp
269 This privilege is interpreted only if the system is configured with Trusted
270 Extensions.
271 .RE

273 .sp
274 .ne 2
275 .na
276 \fB\fBPRIV_FILE_WRITE\fR\fR
277 .ad
278 .sp .6
279 .RS 4n
280 Allow a process to open objects in the filesystem for writing, or otherwise
281 modify them. This privilege is not necessary to write to an already open file
282 which was opened before dropping the \fBPRIV_FILE_WRITE\fR privilege.
283 .RE

285 .sp
286 .ne 2
287 .na
288 \fB\fBPRIV_GRAPHICS_ACCESS\fR\fR
289 .ad
290 .sp .6
291 .RS 4n
292 Allow a process to make privileged ioctls to graphics devices. Typically only
293 an xserver process needs to have this privilege. A process with this privilege
294 is also allowed to perform privileged graphics device mappings.
295 .RE

297 .sp
298 .ne 2
299 .na
300 \fB\fBPRIV_GRAPHICS_MAP\fR\fR
301 .ad
302 .sp .6
303 .RS 4n
304 Allow a process to perform privileged mappings through a graphics device.
305 .RE

307 .sp
308 .ne 2
309 .na
310 \fB\fBPRIV_IPC_DAC_READ\fR\fR
311 .ad
312 .sp .6
313 .RS 4n
314 Allow a process to read a System V IPC Message Queue, Semaphore Set, or Shared
315 Memory Segment whose permission bits would not otherwise allow the process read
316 permission.
317 .RE

319 .sp
320 .ne 2
321 .na
```



```
322 \fB\fBPRIV_IPC_DAC_WRITE\fR\fR
323 .ad
324 .sp .6
325 .RS 4n
326 Allow a process to write a System V IPC Message Queue, Semaphore Set, or Shared
327 Memory Segment whose permission bits would not otherwise allow the process
328 write permission.
329 .RE

331 .sp
332 .ne 2
333 .na
334 \fB\fBPRIV_IPC_OWNER\fR\fR
335 .ad
336 .sp .6
337 .RS 4n
338 Allow a process that is not the owner of a System V IPC Message Queue,
339 Semaphore Set, or Shared Memory Segment to remove, change ownership of, or
340 change permission bits of the Message Queue, Semaphore Set, or Shared Memory
341 Segment.
342 .RE

344 .sp
345 .ne 2
346 .na
347 \fB\fBPRIV_NET_ACCESS\fR\fR
348 .ad
349 .sp .6
350 .RS 4n
351 Allow a process to open a TCP, UDP, SDP, or SCTP network endpoint. This
352 privilege is not necessary to communicate using an existing endpoint already
353 opened before dropping the \fBPRIV_NET_ACCESS\fR privilege.
354 .RE

356 .sp
357 .ne 2
358 .na
359 \fB\fBPRIV_NET_BINDMLP\fR\fR
360 .ad
361 .sp .6
362 .RS 4n
363 Allow a process to bind to a port that is configured as a multi-level port
364 (MLP) for the process's zone. This privilege applies to both shared address and
365 zone-specific address MLPs. See \fBtnzonecfg\fR(\fB4\fR) from the Trusted
366 Extensions manual pages for information on configuring MLP ports.
367 .sp
368 This privilege is interpreted only if the system is configured with Trusted
369 Extensions.
370 .RE

372 .sp
373 .ne 2
374 .na
375 \fB\fBPRIV_NET_ICMPACCESS\fR\fR
376 .ad
377 .sp .6
378 .RS 4n
379 Allow a process to send and receive ICMP packets.
380 .RE

382 .sp
383 .ne 2
384 .na
385 \fB\fBPRIV_NET_MAC_AWARE\fR\fR
386 .ad
387 .sp .6
```

```
388 .RS 4n
389 Allow a process to set the \fBNET_MAC_AWARE\fR process flag by using
390 \fBsetpflags\fR(2). This privilege also allows a process to set the
391 \fBSO_MAC_EXEMPT\fR socket option by using \fBsetsockopt\fR(3SOCKET). The
392 \fBNET_MAC_AWARE\fR process flag and the \fBSO_MAC_EXEMPT\fR socket option both
393 allow a local process to communicate with an unlabeled peer if the local
394 process's label dominates the peer's default label, or if the local process
395 runs in the global zone.
396 .sp
397 This privilege is interpreted only if the system is configured with Trusted
398 Extensions.
399 .RE

401 .sp
402 .ne 2
403 .na
404 \fB\fBPRIV_NET_MAC_IMPLICIT\fR\fR
405 .ad
406 .sp .6
407 .RS 4n
408 Allow a process to set \fBSO_MAC_IMPLICIT\fR option by using
409 \fBsetsockopt\fR(3SOCKET).  This allows a privileged process to transmit
410 implicitly-labeled packets to a peer.
411 .sp
412 This privilege is interpreted only if the system is configured with
413 Trusted Extensions.
414 .RE

416 .sp
417 .ne 2
418 .na
419 \fB\fBPRIV_NET_OBSERVABILITY\fR\fR
420 .ad
421 .sp .6
422 .RS 4n
423 Allow a process to open a device for just receiving network traffic, sending
424 traffic is disallowed.
425 .RE

427 .sp
428 .ne 2
429 .na
430 \fB\fBPRIV_NET_PRIVADDR\fR\fR
431 .ad
432 .sp .6
433 .RS 4n
434 Allow a process to bind to a privileged port number. The privilege port numbers
435 are 1-1023 (the traditional UNIX privileged ports) as well as those ports
436 marked as "\fBudp/tcp_extra_priv_ports\fR" with the exception of the ports
437 reserved for use by NFS and SMB.
438 .RE

440 .sp
441 .ne 2
442 .na
443 \fB\fBPRIV_NET_RAWACCESS\fR\fR
444 .ad
445 .sp .6
446 .RS 4n
447 Allow a process to have direct access to the network layer.
448 .RE

450 .sp
451 .ne 2
452 .na
453 \fB\fBPRIV_PROC_AUDIT\fR\fR
```



```
454 .ad
455 .sp .6
456 .RS 4n
457 Allow a process to generate audit records. Allow a process to get its own audit
458 pre-selection information.
459 .RE

461 .sp
462 .ne 2
463 .na
464 \fB\fBPRIV_PROC_CHROOT\fR\fR
465 .ad
466 .sp .6
467 .RS 4n
468 Allow a process to change its root directory.
469 .RE

471 .sp
472 .ne 2
473 .na
474 \fB\fBPRIV_PROC_CLOCK_HIGHRES\fR\fR
475 .ad
476 .sp .6
477 .RS 4n
478 Allow a process to use high resolution timers.
479 .RE

481 .sp
482 .ne 2
483 .na
484 \fB\fBPRIV_PROC_EXEC\fR\fR
485 .ad
486 .sp .6
487 .RS 4n
488 Allow a process to call \fBexec\fR(2).
489 .RE

491 .sp
492 .ne 2
493 .na
494 \fB\fBPRIV_PROC_FORK\fR\fR
495 .ad
496 .sp .6
497 .RS 4n
498 Allow a process to call \fBfork\fR(2), \fBfork1\fR(2), or \fBvfork\fR(2).
499 .RE

501 .sp
502 .ne 2
503 .na
504 \fB\fBPRIV_PROC_INFO\fR\fR
505 .ad
506 .sp .6
507 .RS 4n
508 Allow a process to examine the status of processes other than those to which it
509 can send signals. Processes that cannot be examined cannot be seen in
510 \fB/proc\fR and appear not to exist.
511 .RE

513 .sp
514 .ne 2
515 .na
516 \fB\fBPRIV_PROC_LOCK_MEMORY\fR\fR
517 .ad
518 .sp .6
519 .RS 4n
```

```
520 Allow a process to lock pages in physical memory.
521 .RE

523 .sp
524 .ne 2
525 .na
526 \fB\fBPRIV_PROC_MEMINFO\fR\fR
527 .ad
528 .sp .6
529 .RS 4n
530 Allow a process to access physical memory information.
531 .RE

533 .sp
534 .ne 2
535 .na
536 \fB\fBPRIV_PROC_OWNER\fR\fR
537 .ad
538 .sp .6
539 .RS 4n
540 Allow a process to send signals to other processes and inspect and modify the
541 process state in other processes, regardless of ownership. When modifying
542 another process, additional restrictions apply: the effective privilege set of
543 the attaching process must be a superset of the target process's effective,
544 permitted, and inheritable sets; the limit set must be a superset of the
545 target's limit set; if the target process has any UID set to 0 all privilege
546 must be asserted unless the effective UID is 0. Allow a process to bind
547 arbitrary processes to CPUs.
548 .RE

550 .sp
551 .ne 2
552 .na
553 \fB\fBPRIV_PROC_PRIOUP\fR\fR
554 .ad
555 .sp .6
556 .RS 4n
557 Allow a process to elevate its priority above its current level.
558 .RE

560 .sp
561 .ne 2
562 .na
563 \fB\fBPRIV_PROC_PRIOCNTL\fR\fR
564 .ad
565 .sp .6
566 .RS 4n
567 Allows all that PRIV_PROC_PRIOUP allows.
568 Allow a process to change its scheduling class to any scheduling class,
569 including the RT class.
570 .RE

572 .sp
573 .ne 2
574 .na
575 \fBPRIV_PROC_SECFLAGS\fR
576 .ad
577 .sp .6
578 .RS 4n
579 Allow a process to manipulate the secflags of processes (subject to,
580 additionally, the ability to signal that process).
581 .RE

583 .sp
584 .ne 2
585 .na
```



```
586 \fB\fBPRIV_PROC_SESSION\fR\fR
587 .ad
588 .sp .6
589 .RS 4n
590 Allow a process to send signals or trace processes outside its session.
591 .RE

593 .sp
594 .ne 2
595 .na
596 \fB\fBPRIV_PROC_SETID\fR\fR
597 .ad
598 .sp .6
599 .RS 4n
600 Allow a process to set its UIDs at will, assuming UID 0 requires all privileges
601 to be asserted.
602 .RE

604 .sp
605 .ne 2
606 .na
607 \fB\fBPRIV_PROC_TASKID\fR\fR
608 .ad
609 .sp .6
610 .RS 4n
611 Allow a process to assign a new task ID to the calling process.
612 .RE

614 .sp
615 .ne 2
616 .na
617 \fB\fBPRIV_PROC_ZONE\fR\fR
618 .ad
619 .sp .6
620 .RS 4n
621 Allow a process to trace or send signals to processes in other zones. See
622 \fBzones\fR(5).
623 .RE

625 .sp
626 .ne 2
627 .na
628 \fB\fBPRIV_SYS_ACCT\fR\fR
629 .ad
630 .sp .6
631 .RS 4n
632 Allow a process to enable and disable and manage accounting through
633 \fBacct\fR(2).
634 .RE

636 .sp
637 .ne 2
638 .na
639 \fB\fBPRIV_SYS_ADMIN\fR\fR
640 .ad
641 .sp .6
642 .RS 4n
643 Allow a process to perform system administration tasks such as setting node and
644 domain name and managing \fBfmd\fR(1M) and \fBnscd\fR(1M).
644 domain name and specifying \fBcoreadm\fR(1M) and \fBnscd\fR(1M) settings
645 .RE

647 .sp
648 .ne 2
649 .na
650 \fB\fBPRIV_SYS_AUDIT\fR\fR
```

```
.ad
.sp .6
.RS 4n
Allow a process to start the (kernel) audit daemon. Allow a process to view and
set audit state (audit user ID, audit terminal ID, audit sessions ID, audit
pre-selection mask). Allow a process to turn off and on auditing. Allow a
process to configure the audit parameters (cache and queue sizes, event to
class mappings, and policy options).
.RE

.sp
.ne 2
.na
\fB\fBPRIV_SYS_CONFIG\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to perform various system configuration tasks. Allow
filesystem-specific administrative procedures, such as filesystem configuration
ioctls, quota calls, creation and deletion of snapshots, and manipulating the
PCFS bootsector.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_SYS_DEVICES\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to create device special files. Allow a process to successfully
call a kernel module that calls the kernel \fBdrv_priv\fR(9F) function to check
for allowed access. Allow a process to open the real console device directly.
Allow a process to open devices that have been exclusively opened.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_SYS_DL_CONFIG\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to configure a system’s datalink interfaces.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_SYS_IP_CONFIG\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to configure a system’s IP interfaces and routes. Allow a
process to configure network parameters for \fBTCP/IP\fR using \fBndd\fR. Allow
a process access to otherwise restricted \fBTCP/IP\fR information using
\fBndd\fR. Allow a process to configure \fBIPsec\fR. Allow a process to pop
anchored \fBSTREAM\fRs modules with matching \fBzoneid\fR.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_SYS_IPC_CONFIG\fR\fR
.ad
.sp .6
```



```
.RS 4n
Allow a process to increase the size of a System V IPC Message Queue buffer.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_SYS_IPTUN_CONFIG\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to configure IP tunnel links.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_SYS_LINKDIR\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to unlink and link directories.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_SYS_MOUNT\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to mount and unmount filesystems that would otherwise be
restricted (that is, most filesystems except \fBnamefs\fR). Allow a process to
add and remove swap devices.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_SYS_NET_CONFIG\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to do all that \fBPRIV_SYS_IP_CONFIG\fR,
\fBPRIV_SYS_DL_CONFIG\fR, and \fBPRIV_SYS_PPP_CONFIG\fR allow, plus the
following: use the \fBrpcmod\fR STREAMS module and insert/remove STREAMS
modules on locations other than the top of the module stack.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_SYS_NFS\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to provide NFS service: start NFS kernel threads, perform NFS
locking operations, bind to NFS reserved ports: ports 2049 (\fBnfs\fR) and port
4045 (\fBlockd\fR).
.RE

.sp
.ne 2
.na
\fB\fBPRIV_SYS_PPP_CONFIG\fR\fR
.ad
```

```
783 .sp .6
784 .RS 4n
785 Allow a process to create, configure, and destroy PPP instances with pppd(1M)
786 \fBpppd\fR(1M) and control PPPoE plumbing with \fBspppтun\fR(1M)spppтun(1M).
787 This privilege is granted by default to exclusive IP stack instance zones.
788 .RE

790 .sp
791 .ne 2
792 .na
793 \fB\fBPRIV_SYS_RES_BIND\fR\fR
794 .ad
795 .sp .6
796 .RS 4n
797 Allows a process to bind processes to processor sets.
798 .RE

800 .sp
801 .ne 2
802 .na
803 \fB\fBPRIV_SYS_RES_CONFIG\fR\fR
804 .ad
805 .sp .6
806 .RS 4n
807 Allows all that PRIV_SYS_RES_BIND allows.
808 Allow a process to create and delete processor sets, assign CPUs to processor
809 sets and override the \fBPSET_NOESCAPE\fR property. Allow a process to change
810 the operational status of CPUs in the system using \fBp_online\fR(2). Allow a
811 process to configure filesystem quotas. Allow a process to configure resource
812 pools and bind processes to pools.
813 .RE

815 .sp
816 .ne 2
817 .na
818 \fB\fBPRIV_SYS_RESOURCE\fR\fR
819 .ad
820 .sp .6
821 .RS 4n
822 Allow a process to exceed the resource limits imposed on it by
823 \fBsetrlimit\fR(2) and \fBsetrctl\fR(2).
824 .RE

826 .sp
827 .ne 2
828 .na
829 \fB\fBPRIV_SYS_SMB\fR\fR
830 .ad
831 .sp .6
832 .RS 4n
833 Allow a process to provide NetBIOS or SMB services: start SMB kernel threads or
834 bind to NetBIOS or SMB reserved ports: ports 137, 138, 139 (NetBIOS) and 445
835 (SMB).
836 .RE

838 .sp
839 .ne 2
840 .na
841 \fB\fBPRIV_SYS_SUSER_COMPAT\fR\fR
842 .ad
843 .sp .6
844 .RS 4n
845 Allow a process to successfully call a third party loadable module that calls
846 the kernel \fBsuser()\fR function to check for allowed access. This privilege
847 exists only for third party loadable module compatibility and is not used by
848 illumos.
```



```
848 Solaris proper.
849 .RE

851 .sp
852 .ne 2
853 .na
854 \fB\fBPRIV_SYS_TIME\fR\fR
855 .ad
856 .sp .6
857 .RS 4n
858 Allow a process to manipulate system time using any of the appropriate system
859 calls: \fBstime\fR(2), \fBadjtime\fR(2), and \fBntp_adjtime\fR(2).
860 .RE

862 .sp
863 .ne 2
864 .na
865 \fB\fBPRIV_SYS_TRANS_LABEL\fR\fR
866 .ad
867 .sp .6
868 .RS 4n
869 Allow a process to translate labels that are not dominated by the process's
870 sensitivity label to and from an external string form.
871 .sp
872 This privilege is interpreted only if the system is configured with Trusted
873 Extensions.
874 .RE

876 .sp
877 .ne 2
878 .na
879 \fB\fBPRIV_VIRT_MANAGE\fR\fR
880 .ad
881 .sp .6
882 .RS 4n
883 Allows a process to manage virtualized environments such as \fBxVM\fR(5).
884 .RE

886 .sp
887 .ne 2
888 .na
889 \fB\fBPRIV_WIN_COLORMAP\fR\fR
890 .ad
891 .sp .6
892 .RS 4n
893 Allow a process to override colormap restrictions.
894 .sp
895 Allow a process to install or remove colormaps.
896 .sp
897 Allow a process to retrieve colormap cell entries allocated by other processes.
898 .sp
899 This privilege is interpreted only if the system is configured with Trusted
900 Extensions.
901 .RE

903 .sp
904 .ne 2
905 .na
906 \fB\fBPRIV_WIN_CONFIG\fR\fR
907 .ad
908 .sp .6
909 .RS 4n
910 Allow a process to configure or destroy resources that are permanently retained
911 by the X server.
912 .sp
913 Allow a process to use SetScreenSaver to set the screen saver timeout value
```

```
.sp
Allow a process to use ChangeHosts to modify the display access control list.
.sp
Allow a process to use GrabServer.
.sp
Allow a process to use the SetCloseDownMode request that can retain window,
pixmap, colormap, property, cursor, font, or graphic context resources.
.sp
This privilege is interpreted only if the system is configured with Trusted
Extensions.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_WIN_DAC_READ\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to read from a window resource that it does not own (has a
different user ID).
.sp
This privilege is interpreted only if the system is configured with Trusted
Extensions.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_WIN_DAC_WRITE\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to write to or create a window resource that it does not own
(has a different user ID). A newly created window property is created with the
window's user ID.
.sp
This privilege is interpreted only if the system is configured with Trusted
Extensions.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_WIN_DEVICES\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to perform operations on window input devices.
.sp
Allow a process to get and set keyboard and pointer controls.
.sp
Allow a process to modify pointer button and key mappings.
.sp
This privilege is interpreted only if the system is configured with Trusted
Extensions.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_WIN_DGA\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to use the direct graphics access (DGA) X protocol extensions.
```



```
Direct process access to the frame buffer is still required. Thus the process
must have MAC and DAC privileges that allow access to the frame buffer, or the
frame buffer must be allocated to the process.
.sp
This privilege is interpreted only if the system is configured with Trusted
Extensions.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_WIN_DOWNGRADE_SL\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to set the sensitivity label of a window resource to a
sensitivity label that does not dominate the existing sensitivity label.
.sp
This privilege is interpreted only if the system is configured with Trusted
Extensions.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_WIN_FONTPATH\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to set a font path.
.sp
This privilege is interpreted only if the system is configured with Trusted
Extensions.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_WIN_MAC_READ\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to read from a window resource whose sensitivity label is not
equal to the process sensitivity label.
.sp
This privilege is interpreted only if the system is configured with Trusted
Extensions.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_WIN_MAC_WRITE\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to create a window resource whose sensitivity label is not
equal to the process sensitivity label. A newly created window property is
created with the window's sensitivity label.
.sp
This privilege is interpreted only if the system is configured with Trusted
Extensions.
.RE

.sp
.ne 2
```

```
.na
\fB\fBPRIV_WIN_SELECTION\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to request inter-window data moves without the intervention of
the selection confirmer.
.sp
This privilege is interpreted only if the system is configured with Trusted
Extensions.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_WIN_UPGRADE_SL\fR\fR
.ad
.sp .6
.RS 4n
Allow a process to set the sensitivity label of a window resource to a
sensitivity label that dominates the existing sensitivity label.
.sp
This privilege is interpreted only if the system is configured with Trusted
Extensions.
.RE

.sp
.ne 2
.na
\fB\fBPRIV_XVM_CONTROL\fR\fR
.ad
.sp .6
.RS 4n
Allows a process access to the \fBxVM\fR(5) control devices for managing guest
domains and the hypervisor. This privilege is used only if booted into xVM on
x86 platforms.
.RE

.sp
.LP
Of the privileges listed above, the privileges \fBPRIV_FILE_LINK_ANY\fR,
\fBPRIV_PROC_INFO\fR, \fBPRIV_PROC_SESSION\fR, \fBPRIV_PROC_FORK\fR,
\fBPRIV_FILE_READ\fR, \fBPRIV_FILE_WRITE\fR, \fBPRIV_NET_ACCESS\fR and
\fBPRIV_PROC_EXEC\fR are considered "basic" privileges. These are privileges
that used to be always available to unprivileged processes. By default,
processes still have the basic privileges.
.sp
.LP
The privileges \fBPRIV_PROC_SETID\fR and \fBPRIV_PROC_AUDIT\fR must be present
in the Limit set (see below) of a process in order for set-uid root \fBexec\fRs
to be successful, that is, get an effective UID of 0 and additional privileges.
.sp
.LP
The privilege implementation in illumos extends the process credential with
The privilege implementation in Solaris extends the process credential with
four privilege sets:
.sp
.ne 2
.na
\fBI, the inheritable set\fR
.ad
.RS 26n
The privileges inherited on \fBexec\fR.
.RE

.sp
```



```
.ne 2
.na
\fBP, the permitted set\fR
.ad
.RS 26n
The maximum set of privileges for the process.
.RE

.sp
.ne 2
.na
\fBE, the effective set\fR
.ad
.RS 26n
The privileges currently in effect.
.RE

.sp
.ne 2
.na
\fBL, the limit set\fR
.ad
.RS 26n
The upper bound of the privileges a process and its offspring can obtain.
Changes to L take effect on the next \fBexec\fR.
.RE

.sp
.LP
The sets I, P and E are typically identical to the basic set of privileges for
unprivileged processes. The limit set is typically the full set of privileges.
.sp
.LP
Each process has a Privilege Awareness State (PAS) that can take the value PA
(privilege-aware) and NPA (not-PA). PAS is a transitional mechanism that allows
a choice between full compatibility with the old superuser model and completely
ignoring the effective UID.
.sp
.LP
To facilitate the discussion, we introduce the notion of "observed effective
set" (oE) and "observed permitted set" (oP) and the implementation sets iE and
iP.
.sp
.LP
A process becomes privilege-aware either by manipulating the effective,
permitted, or limit privilege sets through \fBsetppriv\fR(2) or by using
\fBsetpflags\fR(2). In all cases, oE and oP are invariant in the process of
becoming privilege-aware. In the process of becoming privilege-aware, the
following assignments take place:
.sp
.in +2
.nf
iE = oE
iP = oP
.fi
.in -2

.sp
.LP
When a process is privilege-aware, oE and oP are invariant under UID changes.
When a process is not privilege-aware, oE and oP are observed as follows:
.sp
.in +2
.nf
oE = euid == 0 ? L : iE
oP = (euid == 0 || ruid == 0 || suid == 0) ? L : iP
```

```
.fi
.in -2

.sp
.LP
When a non-privilege-aware process has an effective UID of 0, it can exercise
the privileges contained in its limit set, the upper bound of its privileges.
If a non-privilege-aware process has any of the UIDs 0, it appears to be
capable of potentially exercising all privileges in L.
.sp
.LP
It is possible for a process to return to the non-privilege aware state using
\fBsetpflags()\fR. The kernel always attempts this on \fBexec\fR(2). This
operation is permitted only if the following conditions are met:
.RS +4
.TP
.ie t \(bu
.el o
If any of the UIDs is equal to 0, P must be equal to L.
.RE
.RS +4
.TP
.ie t \(bu
.el o
If the effective UID is equal to 0, E must be equal to L.
.RE
.sp
.LP
When a process gives up privilege awareness, the following assignments take
place:
.sp
.in +2
.nf
if (euid == 0) iE = L & I
if (any uid == 0) iP = L & I
.fi
.in -2

.sp
.LP
The privileges obtained when not having a UID of \fB0\fR are the inheritable
set of the process restricted by the limit set.
.sp
.LP
Only privileges in the process's (observed) effective privilege set allow the
process to perform restricted operations. A process can use any of the
privilege manipulation functions to add or remove privileges from the privilege
sets. Privileges can be removed always. Only privileges found in the permitted
set can be added to the effective and inheritable set. The limit set cannot
grow. The inheritable set can be larger than the permitted set.
.sp
.LP
When a process performs an \fBexec\fR(2), the kernel first tries to relinquish
privilege awareness before making the following privilege set modifications:
.sp
.in +2
.nf
E' = P' = I' = L & I
L is unchanged
.fi
.in -2

.sp
.LP
If a process has not manipulated its privileges, the privilege sets effectively
remain the same, as E, P and I are already identical.
```



```
.sp
.LP
The limit set is enforced at \fBexec\fR time.
.sp
.LP
To run a non-privilege-aware application in a backward-compatible manner, a
privilege-aware application should start the non-privilege-aware application
with I=basic.
.sp
.LP
For most privileges, absence of the privilege simply results in a failure. In
some instances, the absence of a privilege can cause system calls to behave
differently. In other instances, the removal of a privilege can force a set-uid
application to seriously malfunction. Privileges of this type are considered
"unsafe". When a process is lacking any of the unsafe privileges from its limit
set, the system does not honor the set-uid bit of set-uid root applications.
The following unsafe privileges have been identified: \fBproc_setid\fR,
\fBsys_resource\fR and \fBproc_audit\fR.
.SS "Privilege Escalation"
.LP
In certain circumstances, a single privilege could lead to a process gaining
one or more additional privileges that were not explicitly granted to that
process. To prevent such an escalation of privileges, the security policy
requires explicit permission for those additional privileges.
.sp
.LP
Common examples of escalation are those mechanisms that allow modification of
system resources through "raw" interfaces; for example, changing kernel data
system resources through "raw'' interfaces; for example, changing kernel data
structures through \fB/dev/kmem\fR or changing files through \fB/dev/dsk/*\fR.
Escalation also occurs when a process controls processes with more privileges
than the controlling process. A special case of this is manipulating or
creating objects owned by UID 0 or trying to obtain UID 0 using
\fBsetuid\fR(2). The special treatment of UID 0 is needed because the UID 0
owns all system configuration files and ordinary file protection mechanisms
allow processes with UID 0 to modify the system configuration. With appropriate
file modifications, a given process running with an effective UID of 0 can gain
all privileges.
.sp
.LP
In situations where a process might obtain UID 0, the security policy requires
additional privileges, up to the full set of privileges. Such restrictions
could be relaxed or removed at such time as additional mechanisms for
protection of system files became available. There are no such mechanisms in
the current release.
the current Solaris release.
.sp
.LP
The use of UID 0 processes should be limited as much as possible. They should
be replaced with programs running under a different UID but with exactly the
privileges they need.
.sp
.LP
Daemons that never need to \fBexec\fR subprocesses should remove the
\fBPRIV_PROC_EXEC\fR privilege from their permitted and limit sets.
.SS "Assigned Privileges and Safeguards"
.LP
When privileges are assigned to a user, the system administrator could give
that user more powers than intended. The administrator should consider whether
safeguards are needed. For example, if the \fBPRIV_PROC_LOCK_MEMORY\fR
privilege is given to a user, the administrator should consider setting the
\fBproject.max-locked-memory\fR resource control as well, to prevent that user
from locking all memory.
.SS "Privilege Debugging"
.LP
When a system call fails with a permission error, it is not always immediately
```

```
1304 obvious what caused the problem. To debug such a problem, you can use a tool
1305 called \fBprivilege debugging\fR. When privilege debugging is enabled for a
1306 process, the kernel reports missing privileges on the controlling terminal of
1307 the process. (Enable debugging for a process with the \fB-D\fR option of
1308 \fBppriv\fR(1).) Additionally, the administrator can enable system-wide
1309 privilege debugging by setting the \fBsystem\fR(4) variable \fBpriv_debug\fR
1310 using:
1311 .sp
1312 .in +2
1313 .nf
1314 set priv_debug = 1
1315 .fi
1316 .in -2

1318 .sp
1319 .LP
1320 On a running system, you can use \fBmdb\fR(1) to change this variable.
1321 .SS "Privilege Administration"
1325 .LP
1322 Use \fBusermod\fR(1M) or \fBrolemod\fR(1M)
1323 to assign privileges to or modify privileges for, respectively, a user or a
1324 role. Use \fBppriv\fR(1) to enumerate the privileges supported on a system and
1325 \fBtruss\fR(1) to determine which privileges a program requires.
1326 .SH SEE ALSO
1331 .LP
1327 \fBmdb\fR(1), \fBppriv\fR(1), \fBadd_drv\fR(1M), \fBifconfig\fR(1M),
1328 \fBlockd\fR(1M), \fBnfsd\fR(1M), \fBpppd\fR(1M), \fBrem_drv\fR(1M),
1329 \fBsmbd\fR(1M), \fBsppptun\fR(1M), \fBupdate_drv\fR(1M), \fBIntro\fR(2),
1330 \fBaccess\fR(2), \fBacct\fR(2), \fBacl\fR(2), \fBadjtime\fR(2), \fBaudit\fR(2),
1331 \fBauditon\fR(2), \fBchmod\fR(2), \fBchown\fR(2), \fBchroot\fR(2),
1332 \fBcreat\fR(2), \fBexec\fR(2), \fBfcntl\fR(2), \fBfork\fR(2),
1333 \fBfpathconf\fR(2), \fBgetacct\fR(2), \fBgetpflags\fR(2), \fBgetppriv\fR(2),
1334 \fBgetsid\fR(2), \fBkill\fR(2), \fBlink\fR(2), \fBmemcntl\fR(2),
1335 \fBmknod\fR(2), \fBmount\fR(2), \fBmsgctl\fR(2), \fBnice\fR(2),
1336 \fBntp_adjtime\fR(2), \fBopen\fR(2), \fBp_online\fR(2), \fBpriocntl\fR(2),
1337 \fBpriocntlset\fR(2), \fBprocessor_bind\fR(2), \fBpset_bind\fR(2),
1338 \fBpset_create\fR(2), \fBreadlink\fR(2), \fBresolvepath\fR(2), \fBrmdir\fR(2),
1339 \fBsemctl\fR(2), \fBsetauid\fR(2), \fBsetegid\fR(2), \fBseteuid\fR(2),
1340 \fBsetgid\fR(2), \fBsetgroups\fR(2), \fBsetpflags\fR(2), \fBsetppriv\fR(2),
1341 \fBsetrctl\fR(2), \fBsetregid\fR(2), \fBsetreuid\fR(2), \fBsetrlimit\fR(2),
1342 \fBsettaskid\fR(2), \fBsetuid\fR(2), \fBshmctl\fR(2), \fBshmget\fR(2),
1343 \fBshmop\fR(2), \fBsigsend\fR(2), \fBstat\fR(2), \fBstatvfs\fR(2),
1344 \fBstime\fR(2), \fBswapctl\fR(2), \fBsysinfo\fR(2), \fBuadmin\fR(2),
1345 \fBulimit\fR(2), \fBumount\fR(2), \fBunlink\fR(2), \fButime\fR(2),
1346 \fButimes\fR(2), \fBbind\fR(3SOCKET), \fBdoor_ucred\fR(3C),
1347 \fBpriv_addset\fR(3C), \fBpriv_set\fR(3C), \fBpriv_getbyname\fR(3C),
1348 \fBpriv_getbynum\fR(3C), \fBpriv_set_to_str\fR(3C), \fBpriv_str_to_set\fR(3C),
1349 \fBsocket\fR(3SOCKET), \fBt_bind\fR(3NSL), \fBtimer_create\fR(3C),
1350 \fBucred_get\fR(3C), \fBexec_attr\fR(4), \fBproc\fR(4), \fBsystem\fR(4),
1351 \fBuser_attr\fR(4), \fBxVM\fR(5), \fBddi_cred\fR(9F), \fBdrv_priv\fR(9F),
1352 \fBpriv_getbyname\fR(9F), \fBpriv_policy\fR(9F), \fBpriv_policy_choice\fR(9F),
1353 \fBpriv_policy_only\fR(9F)
1354 .sp
1355 .LP
1356 \fISystem Administration Guide: Security Services\fR
```