

```

new/usr/src/cmd/cmd-inet/usr.sbin/socket-filter-kssl.xml 1
*****
5206 Sat Nov 22 13:16:57 2014
new/usr/src/cmd/cmd-inet/usr.sbin/socket-filter-kssl.xml
1403 Support deferring TCP accept()s
*****
1 <?xml version="1.0"?>
2 <!DOCTYPE service_bundle SYSTEM "/usr/share/lib/xml/dtd/service_bundle.dtd.1">
3 <!--
4 Copyright (c) 2010, Oracle and/or its affiliates. All rights reserved.
5 Copyright 2012, Nexenta Systems, Inc. All rights reserved.

7 CDDL HEADER START

9 The contents of this file are subject to the terms of the
10 Common Development and Distribution License (the "License").
11 You may not use this file except in compliance with the License.

13 You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
14 or http://www.opensolaris.org/os/licensing.
15 See the License for the specific language governing permissions
16 and limitations under the License.

18 When distributing Covered Code, include this CDDL HEADER in each
19 file and include the License file at usr/src/OPENSOLARIS.LICENSE.
20 If applicable, add the following below this CDDL HEADER, with the
21 fields enclosed by brackets "[]" replaced with your own identifying
22 information: Portions Copyright [yyyy] [name of copyright owner]

24 CDDL HEADER END

26 NOTE: This service manifest is not editable; its contents will
27 be overwritten by package or patch operations, including
28 operating system upgrade. Make customizations in a different
29 file.
30 -->

32 <service_bundle type='manifest' name='SUNWcs:socket-filter-kssl'>

34 <service
35 name='network/socket-filter'
36 type='service'
37 version='1'>

39 <instance name='kssl' enabled='false'>
40 <dependency
41 name='kssl-filter-filesystem-root'
42 grouping='require_all'
43 restart_on='none'
44 type='service'>
45 <service_fmri value='svc:/system/filesystem/root' />
46 </dependency>

48 <exec_method
49 type='method'
50 name='start'
51 exec='/lib/svc/method/svc-sockfilter start'
52 timeout_seconds='60' />

54 <exec_method
55 type='method'
56 name='stop'
57 exec='/lib/svc/method/svc-sockfilter stop'
58 timeout_seconds='60' />

60 <property_group name='startd' type='framework'>
61 <propval name='duration' type='astring'

```

```

new/usr/src/cmd/cmd-inet/usr.sbin/socket-filter-kssl.xml 2
62 value='transient' />
63 </property_group>

65 <property_group name='socket-filter' type='framework'>
66 <propval name='name' type='astring' value='ksslif' />
67 <propval name='module_name' type='astring'
68 value='ksslif' />
69 <propval name='attach_semantics' type='astring'
70 value='auto' />
71 <propval name='socket_tuples' type='astring'
72 value='2:2:0,2:2:6,26:2:0,26:2:6' />
73 </property_group>

74
75 <template>
76 <common_name>
77 <loctext xml:lang='C'>
78 kernel SSL socket filter
79 </loctext>
80 </common_name>
81 <documentation>
82 <manpage title='ksslcfg' section='1M'
83 manpath='/usr/share/man' />
84 </documentation>
85 </template>
86 </instance>

88 <instance name='http' enabled='true'>
89 <dependency
90 name='http-filter-filesystem-root'
91 grouping='require_all'
92 restart_on='none'
93 type='service'>
94 <service_fmri value='svc:/system/filesystem/root' />
95 </dependency>

97 <exec_method
98 type='method'
99 name='start'
100 exec='/lib/svc/method/svc-sockfilter start'
101 timeout_seconds='60' />

103 <exec_method
104 type='method'
105 name='stop'
106 exec='/lib/svc/method/svc-sockfilter stop'
107 timeout_seconds='60' />

109 <property_group name='startd' type='framework'>
110 <propval name='duration' type='astring'
111 value='transient' />
112 </property_group>

114 <property_group name='socket-filter' type='framework'>
115 <propval name='name' type='astring' value='httpfilt' />
116 <propval name='module_name' type='astring'
117 value='httpfilt' />
118 <propval name='attach_semantics' type='astring'
119 value='prog' />
120 <propval name='socket_tuples' type='astring'
121 value='2:2:0,2:2:6,26:2:0,26:2:6' />
122 </property_group>

124 <template>
125 <common_name>
126 <loctext xml:lang='C'>
127 kernel inbound HTTP request socket filter

```

```
128         </loctext>
129         </common_name>
130         <documentation>
131             <manpage title='httpfilt' section='7M'
132                 manpath='/usr/share/man' />
133         </documentation>
134     </template>
135 </instance>
136
137 <instance name='data' enabled='true'>
138     <dependency
139         name='data-filter-filesystem-root'
140         grouping='require_all'
141         restart_on='none'
142         type='service'>
143         <service_fmri value='svc:/system/filesystem/root' />
144     </dependency>
145
146     <exec_method
147         type='method'
148         name='start'
149         exec='/lib/svc/method/svc-sockfilter start'
150         timeout_seconds='60' />
151
152     <exec_method
153         type='method'
154         name='stop'
155         exec='/lib/svc/method/svc-sockfilter stop'
156         timeout_seconds='60' />
157
158     <property_group name='startd' type='framework'>
159         <propval name='duration' type='astring'
160             value='transient' />
161     </property_group>
162
163     <property_group name='socket-filter' type='framework'>
164         <propval name='name' type='astring' value='datafilt' />
165         <propval name='module_name' type='astring'
166             value='datafilt' />
167         <propval name='attach_semantics' type='astring'
168             value='prog' />
169         <propval name='socket_tuples' type='astring'
170             value='2:2:0,2:2:6,26:2:0,26:2:6' />
171     </property_group>
172
173     <template>
174         <common_name>
175             <loctext xml:lang='C'>
176                 TCP inbound data socket filter
177             </loctext>
178         </common_name>
179         <documentation>
180             <manpage title='datafilt' section='7M'
181                 manpath='/usr/share/man' />
182         </documentation>
183     </template>
184 </instance>
185
186     <stability value='Unstable' />
187 </service>
188
189 </service_bundle>
```

```

*****
6497 Sat Nov 22 13:16:58 2014
new/usr/src/man/man7/Intro.7
1403 Support deferring TCP accept()s
*****
1 \" te
2 .\" Copyright 2014 Ryan Zezeski
3 .\" Copyright (c) 1999, Sun Microsystems, Inc. All Rights Reserved.
4 .\" Copyright 1989 AT&T
5 .\" The contents of this file are subject to the terms of the Common Development
6 .\" You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
7 .\" When distributing Covered Code, include this CDDL HEADER in each file and in
8 .TH INTRO 7 "Nov 20, 2014"
9 .TH INTRO 7 "Sep 29, 1994"
10 .SH NAME
11 Intro, intro \- introduction to special files
12 .SH DESCRIPTION
13 .sp
14 .LP
15 This section describes various device and network interfaces available on the
16 system. The types of interfaces described include character and block
17 devices, STREAMS modules, network protocols, file systems, and ioctl requests
18 for driver subsystems and classes.
19 .sp
20 .LP
21 This section contains the following major collections:
22 .sp
23 .ne 2
24 .na
25 \fB(7D)\fR
26 .ad
27 .RS 9n
28 The system provides drivers for a variety of hardware devices, such as disk,
29 magnetic tapes, serial communication lines, mice, and frame buffers, as well
30 as virtual devices such as pseudo-terminals and windows.
31 .sp
32 This section describes special files that refer to specific hardware
33 peripherals and device drivers. STREAMS device drivers are also described.
34 Characteristics of both the hardware device and the corresponding device driver
35 are discussed where applicable.
36 .sp
37 An application accesses a device through that device's special file. This
38 section specifies the device special file to be used to access the device as
39 well as application programming interface (API) information relevant to the use
40 of the device driver.
41 .sp
42 All device special files are located under the \fB/devices\fR directory. The
43 \fB/devices\fR directory hierarchy attempts to mirror the hierarchy of system
44 busses, controllers, and devices configured on the system. Logical device
45 names for special files in \fB/devices\fR are located under the \fB/dev\fR
46 directory. Although not every special file under \fB/devices\fR will have a
47 corresponding logical entry under \fB/dev\fR, whenever possible, an
48 application should reference a device using the logical name for the device.
49 Logical device names are listed in the \fBFILES\fR section of the page for the
50 device in question.
51 .sp
52 This section also describes driver configuration where applicable. Many device
53 drivers have a driver configuration file of the form
54 \fBdriver_name\fR\fB&.conf\fR associated with them (see
55 \fBdriver.conf\fR(4)). The configuration information stored in the driver
56 configuration file is used to configure the driver and the device. Driver
57 configuration files are located in \fB/kernel/drv\fR and
58 \fB/usr/kernel/drv\fR. Driver configuration files for platform dependent
59 drivers are located in \fB/platform/uname\fR \fB-i'\fR/kernel/drv\fR where
60 \fBuname\fR \fB-i'\fR is the output of the \fBuname\fR(1) command with the
61 \fB-i'\fR option.

```

```

60 .sp
61 Some driver configuration files may contain user configurable properties.
62 Changes in a driver's configuration file will not take effect until the system
63 is rebooted or the driver has been removed and re-added (see \fBrem_drv\fR(1M))
64 and \fBadd_drv\fR(1M)).
65 .RE
66 .sp
67 .ne 2
68 .na
69 \fB(7F)\fR
70 .ad
71 .RS 9n
72 This section describes the programmatic interface for several file systems
73 supported by SunOS.
74 .RE
75 .sp
76 .ne 2
77 .na
78 \fB(7I)\fR
79 .ad
80 .RS 9n
81 This section describes ioctl requests which apply to a class of drivers or
82 subsystems. For example, ioctl requests which apply to most tape devices are
83 discussed in \fBmrio\fR(7I). Ioctl requests relevant to only a specific
84 device are described on the man page for that device. The page for the device
85 in question should still be examined for exceptions to the ioctls listed in
86 section 7I.
87 .RE
88 .sp
89 .ne 2
90 .na
91 \fB(7M)\fR
92 .ad
93 .RS 9n
94 This section describes \fBSTREAMS\fR and socket filter modules. Note
95 that \fBSTREAMS\fR drivers are discussed in section 7D.
96 \fBstreamio\fR(7I) contains a list of ioctl requests used to
97 manipulate \fBSTREAMS\fR modules and interface with the \fBSTREAMS\fR
98 framework. Ioctl requests specific to a \fBSTREAMS\fR module will be
99 discussed on the man page for that module.
100 This section describes \fBSTREAMS\fR modules. Note that \fBSTREAMS\fR
101 drivers are discussed in section 7D. \fBstreamio\fR(7I) contains a list of
102 ioctl requests used to manipulate \fBSTREAMS\fR modules and interface with the
103 \fBSTREAMS\fR framework. Ioctl requests specific to a \fBSTREAMS\fR module
104 will be discussed on the man page for that module.
105 .RE
106 .sp
107 .ne 2
108 .na
109 \fB(7P)\fR
110 .ad
111 .RS 9n
112 This section describes various network protocols available in SunOS.
113 .sp
114 SunOS supports both socket-based and \fBSTREAMS\fR-based network
115 communications. The Internet protocol family, described in \fBinet\fR(7P), is
116 the primary protocol family supported by SunOS, although the system can support
117 a number of others. The raw interface provides low-level services, such as
118 packet fragmentation and reassembly, routing, addressing, and basic transport
119 for socket-based implementations. Facilities for communicating using an
120 Internet-family protocol are generally accessed by specifying the \fBINET\fR
121 address family when binding a socket; see \fBsocket\fR(3SOCKET) for details.

```

```
121 .sp
122 Major protocols in the Internet family include:
123 .RS +4
124 .TP
125 .ie t \(\bu
126 .el o
127 The Internet Protocol (IP) itself, which supports the universal datagram
128 format, as described in \fBip\fR(7P). This is the default protocol for
129 \fB SOCK_RAW\fR type sockets within the \fBAF_INET\fR domain.
130 .RE
131 .RS +4
132 .TP
133 .ie t \(\bu
134 .el o
135 The Transmission Control Protocol (TCP); see \fBtcp\fR(7P). This is the default
136 protocol for \fB SOCK_STREAM\fR type sockets.
137 .RE
138 .RS +4
139 .TP
140 .ie t \(\bu
141 .el o
142 The User Datagram Protocol (UDP); see \fBudp\fR(7P). This is the default
143 protocol for \fB SOCK_DGRAM\fR type sockets.
144 .RE
145 .RS +4
146 .TP
147 .ie t \(\bu
148 .el o
149 The Address Resolution Protocol (ARP); see \fBarp\fR(7P).
150 .RE
151 .RS +4
152 .TP
153 .ie t \(\bu
154 .el o
155 The Internet Control Message Protocol (ICMP); see \fBicmp\fR(7P).
156 .RE
157 .RE

159 .SH SEE ALSO
159 .sp
160 .LP
160 \fBadd_drv\fR(1M), \fBrem_drv\fR(1M), \fBIntro\fR(3), \fBioctl\fR(2),
161 \fBsocket\fR(3SOCKET), \fBdriver.conf\fR(4), \fBarp\fR(7P), \fBicmp\fR(7P),
162 \fBinet\fR(7P), \fBip\fR(7P), \fBmtio\fR(7I), \fBst\fR(7D), \fBstreamio\fR(7I),
163 \fBtcp\fR(7P), \fBudp\fR(7P)
164 .sp
165 .LP
166 \fISystem Administration Guide: IP Services\fR
167 .sp
168 .LP
169 \fISTREAMS Programming Guide\fR
170 .sp
171 .LP
172 \fIWriting Device Drivers\fR
```

new/usr/src/man/man7m/datafilt.7m

1

\*\*\*\*\*

1312 Sat Nov 22 13:16:58 2014

new/usr/src/man/man7m/datafilt.7m

1403 Support deferring TCP accept()  
s

\*\*\*\*\*

```
1 .\"
2 .\" This file and its contents are supplied under the terms of the
3 .\" Common Development and Distribution License ("CDDL"), version 1.0.
4 .\" You may only use this file in accordance with the terms of version
5 .\" 1.0 of the CDDL.
6 .\"
7 .\" A full copy of the text of the CDDL should have accompanied this
8 .\" source. A copy of the CDDL is also available via the Internet at
9 .\" http://www.illumos.org/license/CDDL.
10 .\"
11 .\"
12 .\" Copyright 2014 Ryan Zezeski
13 .\"
14 .Dd Nov 20, 2014
15 .Dt DATAFILT 7M
16 .Os
17 .Sh NAME
18 .Nm datafilt
19 .Nd socket filter module for deferred TCP connections
20 .Sh DESCRIPTION
21 The
22 .Nm datafilt
23 socket filter provides deferment of
24 .Xr accept 3SOCKET
25 for TCP connections. The accept call will not return until at least
26 one byte has been buffered by the kernel. Deferment assures the
27 application that the first call to
28 .Xr read 2 or
29 .Xr recv 3SOCKET
30 will not block. It reduces unnecessary switching between user and
31 kernel.
32 .Sh EXAMPLES
33 .Ss Example 1
34 Enable deferment on the listening socket.
35 .Bd -literal
36 setsockopt(lssock, SOL_FILTER, FIL_ATTACH, "datafilt", 8);
37 .Ed
38 .Ss Example 2
39 Disable deferment on the listening socket.
40 .Bd -literal
41 char filt[] = "datafilt";
42 setsockopt(lssock, SOL_FILTER, FIL_DETACH, filt, strlen(filt) + 1);
43 .Ed
44 .Sh SEE ALSO
45 .Xr setsockopt 3SOCKET ,
46 .Xr httpfilt 7M
```

```
*****
2191 Sat Nov 22 13:16:58 2014
new/usr/src/man/man7m/httpfilt.7m
1403 Support deferring TCP accept()s
*****
```

```
1 .\"
2 .\" This file and its contents are supplied under the terms of the
3 .\" Common Development and Distribution License ("CDDL"), version 1.0.
4 .\" You may only use this file in accordance with the terms of version
5 .\" 1.0 of the CDDL.
6 .\"
7 .\" A full copy of the text of the CDDL should have accompanied this
8 .\" source. A copy of the CDDL is also available via the Internet at
9 .\" http://www.illumos.org/license/CDDL.
10 .\"
11 .\"
12 .\" Copyright 2014 Ryan Zezeski
13 .\"
14 .Dd Nov 20, 2014
15 .Dt HTTPFILT 7M
16 .Os
17 .Sh NAME
18 .Nm httpfilt
19 .Nd socket filter module for deferred HTTP connections
20 .Sh DESCRIPTION
21 The
22 .Nm httpfilt
23 socket filter provides deferment of
24 .Xr accept 3SOCKET
25 for HTTP connections. The accept call will not return until the CRLF
26 line that designates the end of the headers has been seen, or until 8K
27 has been buffered. Deferment assures the application that the first
28 call to
29 .Xr read 2 or
30 .Xr recv 3SOCKET
31 will not block. It reduces unnecessary switching between user and
32 kernel.
33 .Sh IMPLEMENTATION NOTES
34 It's important to understand that this filter only defers HTTP
35 requests, it doesn't parse them. It's not meant to be an in-kernel
36 HTTP server, rejecting or accepting requests. That's still the job of
37 the application. It simply does its best to buffer data until the HTTP
38 request is complete, sans the body.
39 .Pp
40 As the filter does not implement a complete HTTP request parser it can
41 be fooled. It is possible to create bogus requests which keep the
42 socket deferred in the kernel indefinitely. These deferred sockets are
43 allowed to stay open until the socketfilter system feels pressure and
44 starts closing them.
45 .Pp
46 If you want to avoid indefinite deferral then
47 .Xr datafilt 7M
48 may be used instead. It defers the connection until at least one byte
49 arrives.
50 .Sh EXAMPLES
51 .Ss Example 1
52 Enable HTTP deferment on the listening socket.
53 .Bd -literal
54 char filt[] = "httpfilt";
55 setsockopt(lsock, SOL_FILTER, FIL_ATTACH, filt, strlen(filt) + 1);
56 .Ed
57 .Ss Example 2
58 Disable HTTP deferment on the listening socket.
59 .Bd -literal
60 char filt[] = "httpfilt";
61 setsockopt(lsock, SOL_FILTER, FIL_DETACH, filt, strlen(filt) + 1);
```

```
62 .Ed
63 .Sh SEE ALSO
64 .Xr setsockopt 3SOCKET ,
65 .Xr datafilt 7M
```

new/usr/src/pkg/manifests/system-kernel.mf

1

```
*****
45516 Sat Nov 22 13:16:58 2014
new/usr/src/pkg/manifests/system-kernel.mf
1403 Support deferring TCP accept()s
*****
1 #
2 # CDDL HEADER START
3 #
4 # The contents of this file are subject to the terms of the
5 # Common Development and Distribution License (the "License").
6 # You may not use this file except in compliance with the License.
7 #
8 # You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9 # or http://www.opensolaris.org/os/licensing.
10 # See the License for the specific language governing permissions
11 # and limitations under the License.
12 #
13 # When distributing Covered Code, include this CDDL HEADER in each
14 # file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 # If applicable, add the following below this CDDL HEADER, with the
16 # fields enclosed by brackets "[]" replaced with your own identifying
17 # information: Portions Copyright [yyyy] [name of copyright owner]
18 #
19 # CDDL HEADER END
20 #
21 #
22 #
23 # Copyright (c) 2010, Oracle and/or its affiliates. All rights reserved.
24 # Copyright 2012, Nexenta Systems, Inc. All rights reserved.
25 #
26 #
27 #
28 # The default for payload-bearing actions in this package is to appear in the
29 # global zone only. See the include file for greater detail, as well as
30 # information about overriding the defaults.
31 #
32 <include global_zone_only_component>
33 <include system-kernel.man1m.inc>
34 <include system-kernel.man2.inc>
35 <include system-kernel.man4.inc>
36 <include system-kernel.man5.inc>
37 <include system-kernel.man7.inc>
38 <include system-kernel.man7d.inc>
39 <include system-kernel.man7fs.inc>
40 <include system-kernel.man7m.inc>
41 <include system-kernel.man7p.inc>
42 <include system-kernel.man9.inc>
43 <include system-kernel.man9e.inc>
44 <include system-kernel.man9f.inc>
45 <include system-kernel.man9p.inc>
46 <include system-kernel.man9s.inc>
47 set name=pkg.fmri value=pkg:/system/kernel@$(PKGVERS)
48 set name=pkg.description \
49     value="core kernel software for a specific instruction-set architecture"
50 set name=pkg.summary value="Core Solaris Kernel"
51 set name=info.classification value=org.opensolaris.category.2008:System/Core
52 set name=variant.arch value=$(ARCH)
53 dir path=boot group=sys
54 $(i386_ONLY)dir path=boot/acpi group=sys
55 $(i386_ONLY)dir path=boot/acpi/tables group=sys
56 dir path=boot/solaris group=sys
57 dir path=boot/solaris/bin group=sys
58 dir path=etc group=sys
59 dir path=etc/crypto group=sys
60 dir path=etc/sock2path.d group=sys
61 dir path=kernel group=sys
```

new/usr/src/pkg/manifests/system-kernel.mf

2

```
62 $(i386_ONLY)dir path=kernel/$(ARCH64) group=sys
63 dir path=kernel/crypto group=sys
64 dir path=kernel/crypto/$(ARCH64) group=sys
65 dir path=kernel/dacf group=sys
66 dir path=kernel/dacf/$(ARCH64) group=sys
67 dir path=kernel/drv group=sys
68 dir path=kernel/drv/$(ARCH64) group=sys
69 dir path=kernel/exec group=sys
70 dir path=kernel/exec/$(ARCH64) group=sys
71 dir path=kernel/fs group=sys
72 dir path=kernel/fs/$(ARCH64) group=sys
73 dir path=kernel/ipp group=sys
74 dir path=kernel/ipp/$(ARCH64) group=sys
75 dir path=kernel/kiconv group=sys
76 dir path=kernel/kiconv/$(ARCH64) group=sys
77 dir path=kernel/mac group=sys
78 dir path=kernel/mac/$(ARCH64) group=sys
79 dir path=kernel/misc group=sys
80 dir path=kernel/misc/$(ARCH64) group=sys
81 dir path=kernel/misc/scsi_vhci group=sys
82 dir path=kernel/misc/scsi_vhci/$(ARCH64) group=sys
83 dir path=kernel/sched group=sys
84 dir path=kernel/sched/$(ARCH64) group=sys
85 dir path=kernel/socketmod group=sys
86 dir path=kernel/socketmod/$(ARCH64) group=sys
87 dir path=kernel/strmod group=sys
88 dir path=kernel/strmod/$(ARCH64) group=sys
89 dir path=kernel/sys group=sys
90 dir path=kernel/sys/$(ARCH64) group=sys
91 dir path=lib
92 dir path=lib/svc
93 dir path=lib/svc/manifest group=sys
94 dir path=lib/svc/manifest/system group=sys
95 dir path=lib/svc/method
96 dir path=usr/share/man
97 dir path=usr/share/man/man1m
98 dir path=usr/share/man/man2
99 dir path=usr/share/man/man3
100 dir path=usr/share/man/man4
101 dir path=usr/share/man/man5
102 dir path=usr/share/man/man7d
103 dir path=usr/share/man/man7fs
104 dir path=usr/share/man/man7m
105 dir path=usr/share/man/man7p
106 dir path=usr/share/man/man9
107 dir path=usr/share/man/man9e
108 dir path=usr/share/man/man9f
109 dir path=usr/share/man/man9p
110 dir path=usr/share/man/man9s
111 $(i386_ONLY)driver name=acpi_drv perms="* 0666 root sys"
112 driver name=aggr perms="* 0666 root sys"
113 driver name=arp perms="arp 0666 root sys"
114 driver name=bl perms="* 0666 root sys"
115 driver name=bridge clone_perms="bridge 0666 root sys" \
116     policy="read_priv_set=net_rawaccess write_priv_set=net_rawaccess"
117 $(sparc_ONLY)driver name=bscibus alias=SUNW,bscibus
118 $(i386_ONLY)driver name=bscibus alias=SVI0101
119 $(sparc_ONLY)driver name=bscv alias=SUNW,bscv perms="* 0644 root sys"
120 $(i386_ONLY)driver name=bscv
121 driver name=clone
122 driver name=cn perms="* 0620 root tty"
123 driver name=conskbd perms="kbd 0666 root sys"
124 driver name=consm mouse perms="mouse 0666 root sys"
125 driver name=cpuid perms="self 0644 root sys"
126 $(i386_ONLY)driver name=cpunex alias=cpus
127 driver name=crypto perms="crypto 0666 root sys"
```

```

128 driver name=cryptoadm perms="cryptoadm 0644 root sys"
129 $(sparc_ONLY)driver name=dad alias=ide-disk perms="* 0640 root sys"
130 driver name=devinfo perms="devinfo 0640 root sys" \
131 perms="devinfo,ro 0444 root sys"
132 driver name=dld perms="* 0666 root sys"
133 driver name=dlpistub perms="* 0666 root sys"
134 $(sparc_ONLY)driver name=i8042 alias=8042
135 $(i386_ONLY)driver name=i8042
136 driver name=icmp perms="icmp 0666 root sys" \
137 policy="read_priv_set=net_icmpaccess write_priv_set=net_icmpaccess"
138 driver name=icmp6 perms="icmp6 0666 root sys" \
139 policy="read_priv_set=net_icmpaccess write_priv_set=net_icmpaccess"
140 $(i386_ONLY)driver name=intel_nb5000 \
141 alias=pci8086,25c0 \
142 alias=pci8086,25d0 \
143 alias=pci8086,25d4 \
144 alias=pci8086,25d8 \
145 alias=pci8086,3600 \
146 alias=pci8086,4000 \
147 alias=pci8086,4001 \
148 alias=pci8086,4003 \
149 alias=pci8086,65c0
150 $(i386_ONLY)driver name=intel_nhm \
151 alias=pci8086,3423 \
152 alias=pci8086,372a
153 $(i386_ONLY)driver name=intel_nhmex alias=pci8086,3438
154 driver name=ip perms="ip 0666 root sys" \
155 policy="read_priv_set=net_rawaccess write_priv_set=net_rawaccess"
156 driver name=ip6 perms="ip6 0666 root sys" \
157 policy="read_priv_set=net_rawaccess write_priv_set=net_rawaccess"
158 driver name=ipnet perms="lo0 0666 root sys" \
159 policy="read_priv_set=net_observability write_priv_set=net_observability"
160 driver name=ippctl
161 driver name=ipsecah perms="ipsecah 0666 root sys" \
162 policy="read_priv_set=sys_ip_config write_priv_set=sys_ip_config"
163 driver name=ipsecesp perms="ipsecesp 0666 root sys" \
164 policy="read_priv_set=sys_ip_config write_priv_set=sys_ip_config"
165 driver name=iptun
166 driver name=iwscn
167 driver name=kb8042 alias=pnpPNP,303
168 driver name=keysock perms="keysock 0666 root sys" \
169 policy="read_priv_set=sys_ip_config write_priv_set=sys_ip_config"
170 driver name=kmdb
171 driver name=kssl perms="* 0666 root sys"
172 driver name=llcl clone_perms="llcl 0666 root sys"
173 driver name=lofi perms="* 0600 root sys" perms="ctl 0644 root sys"
174 driver name=log perms="conslog 0666 root sys" perms="log 0640 root sys"
175 $(i386_ONLY)driver name=mc-amd \
176 alias=pci1022,1100 \
177 alias=pci1022,1101 \
178 alias=pci1022,1102
179 driver name=mm perms="allkmem 0600 root sys" perms="kmem 0640 root sys" \
180 perms="mem 0640 root sys" perms="null 0666 root sys" \
181 perms="zero 0666 root sys" \
182 policy="allkmem read_priv_set=all write_priv_set=all" \
183 policy="kmem read_priv_set=none write_priv_set=all" \
184 policy="mem read_priv_set=none write_priv_set=all"
185 driver name=mouse8042 alias=pnpPNP,f03
186 $(i386_ONLY)driver name=mpt class=scsi \
187 alias=pci1000,30 \
188 alias=pci1000,50 \
189 alias=pci1000,54 \
190 alias=pci1000,56 \
191 alias=pci1000,58 \
192 alias=pci1000,62 \
193 alias=pciex1000,56 \

```

```

194 alias=pciex1000,58 \
195 alias=pciex1000,62
196 driver name=nulldriver \
197 alias=scsa,nodev \
198 alias=scsa,probe
199 driver name=openeeprom perms="openprom 0640 root sys" policy=write_priv_set=all
200 driver name=options
201 $(sparc_ONLY)driver name=pci_pci class=pci \
202 alias=pci1011,1 \
203 alias=pci1011,21 \
204 alias=pci1011,24 \
205 alias=pci1011,25 \
206 alias=pci1011,26 \
207 alias=pci1014,22 \
208 alias=pciclass,060400
209 $(i386_ONLY)driver name=pci_pci class=pci \
210 alias=pci1011,1 \
211 alias=pci1011,21 \
212 alias=pci1014,22 \
213 alias=pciclass,060400 \
214 alias=pciclass,060401
215 $(sparc_ONLY)driver name=pcieb \
216 alias=pciex108e,9010 \
217 alias=pciex108e,9020 \
218 alias=pciex10b5,8114 \
219 alias=pciex10b5,8516 \
220 alias=pciex10b5,8517 \
221 alias=pciex10b5,8518 \
222 alias=pciex10b5,8532 \
223 alias=pciex10b5,8533 \
224 alias=pciex10b5,8548 \
225 alias=pciexclass,060400
226 $(i386_ONLY)driver name=pcieb \
227 alias=pciexclass,060400 \
228 alias=pciexclass,060401
229 $(sparc_ONLY)driver name=pcieb_bcm alias=pciex1166,103
230 driver name=phymem perms="* 0600 root sys"
231 driver name=poll perms="* 0666 root sys"
232 $(sparc_ONLY)driver name=power alias=ali1535d+-power
233 $(i386_ONLY)driver name=power
234 driver name=pseudo alias=zconsnex
235 driver name=ptc perms="* 0666 root sys"
236 driver name=ptsl perms="* 0666 root sys"
237 $(sparc_ONLY)driver name=ramdisk alias=SUNW,ramdisk perms="* 0600 root sys" \
238 perms="ctl 0644 root sys"
239 $(i386_ONLY)driver name=ramdisk perms="* 0600 root sys" \
240 perms="ctl 0644 root sys"
241 driver name=random perms="* 0644 root sys" policy=write_priv_set=sys_devices
242 driver name=rts perms="rts 0666 root sys"
243 driver name=sad perms="admin 0666 root sys" perms="user 0666 root sys"
244 driver name=scsi_vhci class=scsi-self-identifying perms="* 0666 root sys" \
245 policy="devctl write_priv_set=sys_devices"
246 $(sparc_ONLY)driver name=sd perms="* 0640 root sys" \
247 alias=ide-cdrom \
248 alias=scsiclass,00 \
249 alias=scsiclass,05
250 $(i386_ONLY)driver name=sd perms="* 0640 root sys" \
251 alias=scsiclass,00 \
252 alias=scsiclass,05
253 driver name=sgen perms="* 0600 root sys" \
254 alias=scsa,08.bfcp \
255 alias=scsa,08.bvhci
256 driver name=simnet clone_perms="simnet 0666 root sys" perms="* 0666 root sys"
257 $(i386_ONLY)driver name=smbios perms="smbios 0444 root sys"
258 driver name=softmac
259 driver name=spdsoc perms="spdsoc 0666 root sys" \

```



```

260 policy="read_priv_set=sys_ip_config write_priv_set=sys_ip_config"
261 driver name=st alias=scsiclass,01 perms="* 0666 root sys"
262 driver name=sy perms="tty 0666 root tty"
263 driver name=sysevent perms="* 0600 root sys"
264 driver name=sysmsg perms="msglog 0600 root sys" perms="sysmsg 0600 root sys"
265 driver name=tcp perms="tcp 0666 root sys"
266 driver name=tcp6 perms="tcp6 0666 root sys"
267 driver name=tl perms="* 0666 root sys" clone_perms="ticlts 0666 root sys" \
268   clone_perms="ticots 0666 root sys" clone_perms="ticotsord 0666 root sys"
269 $(sparc_ONLY)driver name=ttymux alias=multiplexer
270 $(i386_ONLY)driver name=tzmon
271 $(sparc_ONLY)driver name=uata \
272   alias=pci1095,646 \
273   alias=pci1095,649 \
274   alias=pci1095,680 \
275   alias=pci10b9,5229 \
276   alias=pci10b9,5288 class=dada class=scsi
277 $(i386_ONLY)driver name=ucode perms="* 0644 root sys"
278 driver name=udp perms="udp 0666 root sys"
279 driver name=udp6 perms="udp6 0666 root sys"
280 $(i386_ONLY)driver name=vgatext \
281   alias=pciclass,000100 \
282   alias=pciclass,030000 \
283   alias=pciclass,030001 \
284   alias=pnpPNP,900
285 driver name=vnics clone_perms="vnics 0666 root sys" perms="* 0666 root sys"
286 driver name=wc perms="* 0600 root sys"
287 $(i386_ONLY)file path=boot/solaris/bin/create_diskmap group=sys mode=0555
288 file path=boot/solaris/bin/create_ramdisk group=sys mode=0555
289 file path=boot/solaris/bin/extract_boot_filelist group=sys mode=0555
290 $(i386_ONLY)file path=boot/solaris/bin/mbr group=sys mode=0555
291 $(i386_ONLY)file path=boot/solaris/bin/symdef group=sys mode=0555
292 $(i386_ONLY)file path=boot/solaris/bin/update_grub group=sys mode=0555
293 file path=boot/solaris/filelist.ramdisk group=sys
294 file path=boot/solaris/filelist.safe group=sys
295 file path=etc/crypto/kcf.conf group=sys \
296   original_name=SUNWckr:etc/crypto/kcf.conf preserve=true
297 $(i386_ONLY)file path=etc/mach group=sys original_name=SUNWos86r:etc/mach \
298   preserve=true
299 file path=etc/name_to_sysnum group=sys \
300   original_name=SUNWckr:etc/name_to_sysnum preserve=renameold
301 file path=etc/sock2path.d/system%2Fkernel group=sys
302 file path=etc/system group=sys original_name=SUNWckr:etc/system preserve=true
303 $(i386_ONLY)file path=kernel/$(ARCH64)/genunix group=sys mode=0755
304 file path=kernel/crypto/$(ARCH64)/aes group=sys mode=0755
305 file path=kernel/crypto/$(ARCH64)/arcfour group=sys mode=0755
306 file path=kernel/crypto/$(ARCH64)/blowfish group=sys mode=0755
307 file path=kernel/crypto/$(ARCH64)/des group=sys mode=0755
308 file path=kernel/crypto/$(ARCH64)/ecc group=sys mode=0755
309 file path=kernel/crypto/$(ARCH64)/md4 group=sys mode=0755
310 file path=kernel/crypto/$(ARCH64)/md5 group=sys mode=0755
311 file path=kernel/crypto/$(ARCH64)/rsa group=sys mode=0755
312 file path=kernel/crypto/$(ARCH64)/shal group=sys mode=0755
313 file path=kernel/crypto/$(ARCH64)/sha2 group=sys mode=0755
314 file path=kernel/crypto/$(ARCH64)/swrand group=sys mode=0755
315 $(i386_ONLY)file path=kernel/crypto/aes group=sys mode=0755
316 $(i386_ONLY)file path=kernel/crypto/arcfour group=sys mode=0755
317 $(i386_ONLY)file path=kernel/crypto/blowfish group=sys mode=0755
318 $(i386_ONLY)file path=kernel/crypto/des group=sys mode=0755
319 $(i386_ONLY)file path=kernel/crypto/ecc group=sys mode=0755
320 $(i386_ONLY)file path=kernel/crypto/md4 group=sys mode=0755
321 $(i386_ONLY)file path=kernel/crypto/md5 group=sys mode=0755
322 $(i386_ONLY)file path=kernel/crypto/rsa group=sys mode=0755
323 $(i386_ONLY)file path=kernel/crypto/shal group=sys mode=0755
324 $(i386_ONLY)file path=kernel/crypto/sha2 group=sys mode=0755
325 $(i386_ONLY)file path=kernel/crypto/swrand group=sys mode=0755

```

```

326 $(sparc_ONLY)file path=kernel/dacf/$(ARCH64)/consconfig_dacf group=sys \
327   mode=0755
328 file path=kernel/dacf/$(ARCH64)/net_dacf group=sys mode=0755
329 $(i386_ONLY)file path=kernel/dacf/net_dacf group=sys mode=0755
330 $(i386_ONLY)file path=kernel/drv/$(ARCH64)/acpi_drv group=sys
331 $(i386_ONLY)file path=kernel/drv/$(ARCH64)/acpi_toshiba group=sys
332 file path=kernel/drv/$(ARCH64)/aggr group=sys
333 file path=kernel/drv/$(ARCH64)/arp group=sys
334 file path=kernel/drv/$(ARCH64)/bl group=sys
335 file path=kernel/drv/$(ARCH64)/bridge group=sys
336 $(i386_ONLY)file path=kernel/drv/$(ARCH64)/bscbs group=sys
337 $(i386_ONLY)file path=kernel/drv/$(ARCH64)/bscv group=sys
338 file path=kernel/drv/$(ARCH64)/clone group=sys
339 file path=kernel/drv/$(ARCH64)/cn group=sys
340 file path=kernel/drv/$(ARCH64)/conskbd group=sys
341 file path=kernel/drv/$(ARCH64)/consm group=sys
342 file path=kernel/drv/$(ARCH64)/cpuid group=sys
343 $(i386_ONLY)file path=kernel/drv/$(ARCH64)/cpunex group=sys
344 file path=kernel/drv/$(ARCH64)/crypto group=sys
345 file path=kernel/drv/$(ARCH64)/cryptoadm group=sys
346 $(sparc_ONLY)file path=kernel/drv/$(ARCH64)/dad group=sys
347 file path=kernel/drv/$(ARCH64)/devinfo group=sys
348 file path=kernel/drv/$(ARCH64)/dld group=sys
349 file path=kernel/drv/$(ARCH64)/dlpistub group=sys
350 file path=kernel/drv/$(ARCH64)/i8042 group=sys
351 file path=kernel/drv/$(ARCH64)/icmp group=sys
352 file path=kernel/drv/$(ARCH64)/icmp6 group=sys
353 $(i386_ONLY)file path=kernel/drv/$(ARCH64)/intel_nb5000 group=sys
354 $(i386_ONLY)file path=kernel/drv/$(ARCH64)/intel_nhm group=sys
355 $(i386_ONLY)file path=kernel/drv/$(ARCH64)/intel_nhmex group=sys
356 file path=kernel/drv/$(ARCH64)/ip group=sys
357 file path=kernel/drv/$(ARCH64)/ip6 group=sys
358 file path=kernel/drv/$(ARCH64)/ipnet group=sys
359 file path=kernel/drv/$(ARCH64)/ippctl group=sys
360 file path=kernel/drv/$(ARCH64)/ipsec group=sys
361 file path=kernel/drv/$(ARCH64)/ipseccp group=sys
362 file path=kernel/drv/$(ARCH64)/iptun group=sys
363 file path=kernel/drv/$(ARCH64)/iwscn group=sys
364 file path=kernel/drv/$(ARCH64)/kb8042 group=sys
365 file path=kernel/drv/$(ARCH64)/keysock group=sys
366 file path=kernel/drv/$(ARCH64)/kmdb group=sys
367 file path=kernel/drv/$(ARCH64)/kssl group=sys
368 file path=kernel/drv/$(ARCH64)/llcl group=sys
369 file path=kernel/drv/$(ARCH64)/lofi group=sys
370 file path=kernel/drv/$(ARCH64)/log group=sys
371 $(i386_ONLY)file path=kernel/drv/$(ARCH64)/mc-amd group=sys
372 file path=kernel/drv/$(ARCH64)/mm group=sys
373 file path=kernel/drv/$(ARCH64)/mouse8042 group=sys
374 $(i386_ONLY)file path=kernel/drv/$(ARCH64)/mpt group=sys
375 file path=kernel/drv/$(ARCH64)/nulldriver group=sys
376 file path=kernel/drv/$(ARCH64)/openeep group=sys
377 file path=kernel/drv/$(ARCH64)/options group=sys
378 file path=kernel/drv/$(ARCH64)/pci_pci group=sys
379 file path=kernel/drv/$(ARCH64)/pcieb group=sys
380 $(sparc_ONLY)file path=kernel/drv/$(ARCH64)/pcieb_bcm group=sys
381 file path=kernel/drv/$(ARCH64)/physmem group=sys
382 file path=kernel/drv/$(ARCH64)/poll group=sys
383 $(i386_ONLY)file path=kernel/drv/$(ARCH64)/power group=sys
384 file path=kernel/drv/$(ARCH64)/pseudo group=sys
385 file path=kernel/drv/$(ARCH64)/ptc group=sys
386 file path=kernel/drv/$(ARCH64)/ptsl group=sys
387 file path=kernel/drv/$(ARCH64)/ramdisk group=sys
388 file path=kernel/drv/$(ARCH64)/random group=sys
389 file path=kernel/drv/$(ARCH64)/rts group=sys
390 file path=kernel/drv/$(ARCH64)/sad group=sys
391 file path=kernel/drv/$(ARCH64)/scsi_vhci group=sys

```

```

392 file path=kernel/drv/$(ARCH64)/sd group=sys
393 file path=kernel/drv/$(ARCH64)/sgen group=sys
394 file path=kernel/drv/$(ARCH64)/simnet group=sys
395 $(i386_ONLY)file path=kernel/drv/$(ARCH64)/smbios group=sys
396 file path=kernel/drv/$(ARCH64)/softmac group=sys
397 file path=kernel/drv/$(ARCH64)/spsock group=sys
398 file path=kernel/drv/$(ARCH64)/st group=sys
399 file path=kernel/drv/$(ARCH64)/sy group=sys
400 file path=kernel/drv/$(ARCH64)/sysevent group=sys
401 file path=kernel/drv/$(ARCH64)/sysmsg group=sys
402 file path=kernel/drv/$(ARCH64)/tcp group=sys
403 file path=kernel/drv/$(ARCH64)/tcp6 group=sys
404 file path=kernel/drv/$(ARCH64)/tl group=sys
405 $(sparc_ONLY)file path=kernel/drv/$(ARCH64)/ttymux group=sys
406 $(i386_ONLY)file path=kernel/drv/$(ARCH64)/tzmon group=sys
407 $(sparc_ONLY)file path=kernel/drv/$(ARCH64)/uata group=sys
408 $(i386_ONLY)file path=kernel/drv/$(ARCH64)/ucode group=sys
409 file path=kernel/drv/$(ARCH64)/udp group=sys
410 file path=kernel/drv/$(ARCH64)/udp6 group=sys
411 $(i386_ONLY)file path=kernel/drv/$(ARCH64)/vgatext group=sys
412 file path=kernel/drv/$(ARCH64)/vnic group=sys
413 file path=kernel/drv/$(ARCH64)/wc group=sys
414 $(i386_ONLY)file path=kernel/drv/acpi_drv group=sys
415 $(i386_ONLY)file path=kernel/drv/acpi_drv.conf group=sys
416 $(i386_ONLY)file path=kernel/drv/acpi_toshiba group=sys
417 $(i386_ONLY)file path=kernel/drv/aggr group=sys
418 file path=kernel/drv/aggr.conf group=sys
419 $(i386_ONLY)file path=kernel/drv/arp group=sys
420 file path=kernel/drv/arp.conf group=sys
421 $(i386_ONLY)file path=kernel/drv/bl group=sys
422 file path=kernel/drv/bl.conf group=sys
423 $(i386_ONLY)file path=kernel/drv/bridge group=sys
424 file path=kernel/drv/bridge.conf group=sys
425 $(i386_ONLY)file path=kernel/drv/bscbus group=sys
426 $(i386_ONLY)file path=kernel/drv/bscbus.conf group=sys
427 $(i386_ONLY)file path=kernel/drv/bscv group=sys
428 $(i386_ONLY)file path=kernel/drv/bscv.conf group=sys
429 $(i386_ONLY)file path=kernel/drv/clone group=sys
430 file path=kernel/drv/clone.conf group=sys
431 $(i386_ONLY)file path=kernel/drv/cn group=sys
432 file path=kernel/drv/cn.conf group=sys
433 $(i386_ONLY)file path=kernel/drv/conskbd group=sys
434 file path=kernel/drv/conskbd.conf group=sys
435 $(i386_ONLY)file path=kernel/drv/consms group=sys
436 file path=kernel/drv/consms.conf group=sys
437 $(i386_ONLY)file path=kernel/drv/cpuid group=sys
438 file path=kernel/drv/cpuid.conf group=sys
439 $(i386_ONLY)file path=kernel/drv/cpunex group=sys
440 $(i386_ONLY)file path=kernel/drv/crypto group=sys
441 file path=kernel/drv/crypto.conf group=sys
442 $(i386_ONLY)file path=kernel/drv/cryptoadm group=sys
443 file path=kernel/drv/cryptoadm.conf group=sys
444 $(sparc_ONLY)file path=kernel/drv/dad.conf group=sys
445 $(i386_ONLY)file path=kernel/drv/devinfo group=sys
446 file path=kernel/drv/devinfo.conf group=sys
447 $(i386_ONLY)file path=kernel/drv/dld group=sys
448 file path=kernel/drv/dld.conf group=sys
449 $(i386_ONLY)file path=kernel/drv/dlpistub group=sys
450 file path=kernel/drv/dlpistub.conf group=sys
451 $(i386_ONLY)file path=kernel/drv/i8042 group=sys
452 $(i386_ONLY)file path=kernel/drv/icmp group=sys
453 file path=kernel/drv/icmp.conf group=sys
454 $(i386_ONLY)file path=kernel/drv/icmp6 group=sys
455 file path=kernel/drv/icmp6.conf group=sys
456 $(i386_ONLY)file path=kernel/drv/intel_nb5000 group=sys
457 $(i386_ONLY)file path=kernel/drv/intel_nb5000.conf group=sys

```

```

458 $(i386_ONLY)file path=kernel/drv/intel_nhm group=sys
459 $(i386_ONLY)file path=kernel/drv/intel_nhm.conf group=sys
460 $(i386_ONLY)file path=kernel/drv/intel_nhmex group=sys
461 $(i386_ONLY)file path=kernel/drv/intel_nhmex.conf group=sys
462 $(i386_ONLY)file path=kernel/drv/ip group=sys
463 file path=kernel/drv/ip.conf group=sys
464 $(i386_ONLY)file path=kernel/drv/ip6 group=sys
465 file path=kernel/drv/ip6.conf group=sys
466 $(i386_ONLY)file path=kernel/drv/ipnet group=sys
467 file path=kernel/drv/ipnet.conf group=sys
468 $(i386_ONLY)file path=kernel/drv/ippctl group=sys
469 file path=kernel/drv/ippctl.conf group=sys
470 $(i386_ONLY)file path=kernel/drv/ipsec group=sys
471 file path=kernel/drv/ipsec.conf group=sys
472 $(i386_ONLY)file path=kernel/drv/ipsecesp group=sys
473 file path=kernel/drv/ipsecesp.conf group=sys
474 $(i386_ONLY)file path=kernel/drv/iptun group=sys
475 file path=kernel/drv/iptun.conf group=sys
476 $(i386_ONLY)file path=kernel/drv/iwscn group=sys
477 file path=kernel/drv/iwscn.conf group=sys
478 $(i386_ONLY)file path=kernel/drv/kb8042 group=sys
479 $(i386_ONLY)file path=kernel/drv/keysock group=sys
480 file path=kernel/drv/keysock.conf group=sys
481 $(i386_ONLY)file path=kernel/drv/kmdb group=sys
482 file path=kernel/drv/kmdb.conf group=sys
483 $(i386_ONLY)file path=kernel/drv/kssl group=sys
484 file path=kernel/drv/kssl.conf group=sys
485 $(i386_ONLY)file path=kernel/drv/llcl group=sys
486 file path=kernel/drv/llcl.conf group=sys
487 $(i386_ONLY)file path=kernel/drv/lofi group=sys
488 file path=kernel/drv/lofi.conf group=sys
489 $(i386_ONLY)file path=kernel/drv/log group=sys
490 file path=kernel/drv/log.conf group=sys \
491     original_name=SUNWckr:kernel/drv/log.conf preserve=true
492 $(i386_ONLY)file path=kernel/drv/mc-amd group=sys
493 $(i386_ONLY)file path=kernel/drv/mc-amd.conf group=sys
494 $(i386_ONLY)file path=kernel/drv/mm group=sys
495 file path=kernel/drv/mm.conf group=sys
496 $(i386_ONLY)file path=kernel/drv/mouse8042 group=sys
497 $(i386_ONLY)file path=kernel/drv/mpt group=sys
498 $(i386_ONLY)file path=kernel/drv/mpt.conf group=sys \
499     original_name=SUNWckr:kernel/drv/mpt.conf preserve=true
500 $(i386_ONLY)file path=kernel/drv/nulldriver group=sys
501 $(i386_ONLY)file path=kernel/drv/openeep group=sys
502 file path=kernel/drv/openeep.conf group=sys
503 $(i386_ONLY)file path=kernel/drv/options group=sys
504 file path=kernel/drv/options.conf group=sys
505 $(i386_ONLY)file path=kernel/drv/pci_pci group=sys
506 $(i386_ONLY)file path=kernel/drv/pcieb group=sys
507 file path=kernel/drv/pcieb.conf group=sys
508 $(i386_ONLY)file path=kernel/drv/phymem group=sys
509 file path=kernel/drv/phymem.conf group=sys
510 $(i386_ONLY)file path=kernel/drv/poll group=sys
511 file path=kernel/drv/poll.conf group=sys
512 $(i386_ONLY)file path=kernel/drv/power group=sys
513 $(i386_ONLY)file path=kernel/drv/power.conf group=sys
514 $(i386_ONLY)file path=kernel/drv/pseudo group=sys
515 file path=kernel/drv/pseudo.conf group=sys
516 $(i386_ONLY)file path=kernel/drv/ptc group=sys
517 file path=kernel/drv/ptc.conf group=sys
518 $(i386_ONLY)file path=kernel/drv/ptsl group=sys
519 file path=kernel/drv/ptsl.conf group=sys
520 $(i386_ONLY)file path=kernel/drv/ramdisk group=sys
521 file path=kernel/drv/ramdisk.conf group=sys
522 $(i386_ONLY)file path=kernel/drv/random group=sys
523 file path=kernel/drv/random.conf group=sys

```

```

524 $(i386_ONLY)file path=kernel/drv/rts group=sys
525 file path=kernel/drv/rts.conf group=sys
526 $(i386_ONLY)file path=kernel/drv/sad group=sys
527 file path=kernel/drv/sad.conf group=sys
528 $(i386_ONLY)file path=kernel/drv/scsi_vhci group=sys
529 file path=kernel/drv/scsi_vhci.conf group=sys \
530     original_name=SUNWckr:kernel/drv/scsi_vhci.conf preserve=true
531 $(i386_ONLY)file path=kernel/drv/sd group=sys
532 $(sparc_ONLY)file path=kernel/drv/sd.conf group=sys \
533     original_name=SUNWckr:kernel/drv/sd.conf preserve=true
534 $(i386_ONLY)file path=kernel/drv/sd.conf group=sys \
535     original_name=SUNWos86r:kernel/drv/sd.conf preserve=true
536 $(i386_ONLY)file path=kernel/drv/sgen group=sys
537 file path=kernel/drv/sgen.conf group=sys \
538     original_name=SUNWckr:kernel/drv/sgen.conf preserve=true
539 $(i386_ONLY)file path=kernel/drv/simnet group=sys
540 file path=kernel/drv/simnet.conf group=sys
541 $(i386_ONLY)file path=kernel/drv/smbios group=sys
542 $(i386_ONLY)file path=kernel/drv/smbios.conf group=sys
543 $(i386_ONLY)file path=kernel/drv/softmac group=sys
544 file path=kernel/drv/softmac.conf group=sys
545 $(i386_ONLY)file path=kernel/drv/spdsock group=sys
546 file path=kernel/drv/spdsock.conf group=sys
547 $(i386_ONLY)file path=kernel/drv/st group=sys
548 file path=kernel/drv/st.conf group=sys \
549     original_name=SUNWckr:kernel/drv/st.conf preserve=true
550 $(i386_ONLY)file path=kernel/drv/sy group=sys
551 file path=kernel/drv/sy.conf group=sys
552 $(i386_ONLY)file path=kernel/drv/sysevent group=sys
553 file path=kernel/drv/sysevent.conf group=sys
554 $(i386_ONLY)file path=kernel/drv/sysmsg group=sys
555 file path=kernel/drv/sysmsg.conf group=sys
556 $(i386_ONLY)file path=kernel/drv/tcp group=sys
557 file path=kernel/drv/tcp.conf group=sys
558 $(i386_ONLY)file path=kernel/drv/tcp6 group=sys
559 file path=kernel/drv/tcp6.conf group=sys
560 $(i386_ONLY)file path=kernel/drv/tl group=sys
561 file path=kernel/drv/tl.conf group=sys
562 $(i386_ONLY)file path=kernel/drv/tzmon group=sys
563 $(i386_ONLY)file path=kernel/drv/tzmon.conf group=sys
564 $(sparc_ONLY)file path=kernel/drv/uata.conf group=sys \
565     original_name=SUNWckr:kernel/drv/uata.conf preserve=true
566 $(i386_ONLY)file path=kernel/drv/ucode group=sys
567 $(i386_ONLY)file path=kernel/drv/ucode.conf group=sys
568 $(i386_ONLY)file path=kernel/drv/udp group=sys
569 file path=kernel/drv/udp.conf group=sys
570 $(i386_ONLY)file path=kernel/drv/udp6 group=sys
571 file path=kernel/drv/udp6.conf group=sys
572 $(i386_ONLY)file path=kernel/drv/vgatext group=sys
573 $(i386_ONLY)file path=kernel/drv/vnic group=sys
574 file path=kernel/drv/vnic.conf group=sys
575 $(i386_ONLY)file path=kernel/drv/wc group=sys
576 file path=kernel/drv/wc.conf group=sys
577 $(sparc_ONLY)file path=kernel/exec/$(ARCH64)/aoutexec group=sys mode=0755
578 file path=kernel/exec/$(ARCH64)/elfexec group=sys mode=0755
579 file path=kernel/exec/$(ARCH64)/intpexec group=sys mode=0755
580 $(i386_ONLY)file path=kernel/exec/elfexec group=sys mode=0755
581 $(i386_ONLY)file path=kernel/exec/intpexec group=sys mode=0755
582 file path=kernel/fs/$(ARCH64)/autofs group=sys mode=0755
583 file path=kernel/fs/$(ARCH64)/cacheafs group=sys mode=0755
584 file path=kernel/fs/$(ARCH64)/ctfs group=sys mode=0755
585 file path=kernel/fs/$(ARCH64)/dcfs group=sys mode=0755
586 file path=kernel/fs/$(ARCH64)/dev group=sys mode=0755
587 file path=kernel/fs/$(ARCH64)/devfs group=sys mode=0755
588 file path=kernel/fs/$(ARCH64)/fifofs group=sys mode=0755
589 file path=kernel/fs/$(ARCH64)/hsfs group=sys mode=0755

```

```

590 file path=kernel/fs/$(ARCH64)/lofs group=sys mode=0755
591 file path=kernel/fs/$(ARCH64)/mntfs group=sys mode=0755
592 file path=kernel/fs/$(ARCH64)/namefs group=sys mode=0755
593 file path=kernel/fs/$(ARCH64)/objfs group=sys mode=0755
594 file path=kernel/fs/$(ARCH64)/procfs group=sys mode=0755
595 file path=kernel/fs/$(ARCH64)/sharefs group=sys mode=0755
596 file path=kernel/fs/$(ARCH64)/sockfs group=sys mode=0755
597 file path=kernel/fs/$(ARCH64)/specfs group=sys mode=0755
598 file path=kernel/fs/$(ARCH64)/tmpfs group=sys mode=0755
599 file path=kernel/fs/$(ARCH64)/ufs group=sys mode=0755
600 $(i386_ONLY)file path=kernel/fs/autofs group=sys mode=0755
601 $(i386_ONLY)file path=kernel/fs/cacheafs group=sys mode=0755
602 $(i386_ONLY)file path=kernel/fs/ctfs group=sys mode=0755
603 $(i386_ONLY)file path=kernel/fs/dcfs group=sys mode=0755
604 $(i386_ONLY)file path=kernel/fs/dev group=sys mode=0755
605 $(i386_ONLY)file path=kernel/fs/devfs group=sys mode=0755
606 $(i386_ONLY)file path=kernel/fs/fifofs group=sys mode=0755
607 $(i386_ONLY)file path=kernel/fs/hsfs group=sys mode=0755
608 $(i386_ONLY)file path=kernel/fs/lofs group=sys mode=0755
609 $(i386_ONLY)file path=kernel/fs/mntfs group=sys mode=0755
610 $(i386_ONLY)file path=kernel/fs/namefs group=sys mode=0755
611 $(i386_ONLY)file path=kernel/fs/objfs group=sys mode=0755
612 $(i386_ONLY)file path=kernel/fs/procfs group=sys mode=0755
613 $(i386_ONLY)file path=kernel/fs/sharefs group=sys mode=0755
614 $(i386_ONLY)file path=kernel/fs/sockfs group=sys mode=0755
615 $(i386_ONLY)file path=kernel/fs/specfs group=sys mode=0755
616 $(i386_ONLY)file path=kernel/fs/tmpfs group=sys mode=0755
617 $(i386_ONLY)file path=kernel/fs/ufs group=sys mode=0755
618 $(i386_ONLY)file path=kernel/genunix group=sys mode=0755
619 file path=kernel/ipp/$(ARCH64)/ipgpc group=sys mode=0755
620 $(i386_ONLY)file path=kernel/ipp/ipgpc group=sys mode=0755
621 file path=kernel/kiconv/$(ARCH64)/kiconv_emea group=sys mode=0755
622 file path=kernel/kiconv/$(ARCH64)/kiconv_ja group=sys mode=0755
623 file path=kernel/kiconv/$(ARCH64)/kiconv_ko group=sys mode=0755
624 file path=kernel/kiconv/$(ARCH64)/kiconv_sc group=sys mode=0755
625 file path=kernel/kiconv/$(ARCH64)/kiconv_tc group=sys mode=0755
626 $(i386_ONLY)file path=kernel/kiconv/kiconv_emea group=sys mode=0755
627 $(i386_ONLY)file path=kernel/kiconv/kiconv_ja group=sys mode=0755
628 $(i386_ONLY)file path=kernel/kiconv/kiconv_ko group=sys mode=0755
629 $(i386_ONLY)file path=kernel/kiconv/kiconv_sc group=sys mode=0755
630 $(i386_ONLY)file path=kernel/kiconv/kiconv_tc group=sys mode=0755
631 file path=kernel/mac/$(ARCH64)/mac_6to4 group=sys mode=0755
632 file path=kernel/mac/$(ARCH64)/mac_ether group=sys mode=0755
633 file path=kernel/mac/$(ARCH64)/mac_ib group=sys mode=0755
634 file path=kernel/mac/$(ARCH64)/mac_ipv4 group=sys mode=0755
635 file path=kernel/mac/$(ARCH64)/mac_ipv6 group=sys mode=0755
636 file path=kernel/mac/$(ARCH64)/mac_wifi group=sys mode=0755
637 $(i386_ONLY)file path=kernel/mac/mac_6to4 group=sys mode=0755
638 $(i386_ONLY)file path=kernel/mac/mac_ether group=sys mode=0755
639 $(i386_ONLY)file path=kernel/mac/mac_ib group=sys mode=0755
640 $(i386_ONLY)file path=kernel/mac/mac_ipv4 group=sys mode=0755
641 $(i386_ONLY)file path=kernel/mac/mac_ipv6 group=sys mode=0755
642 $(i386_ONLY)file path=kernel/mac/mac_wifi group=sys mode=0755
643 $(i386_ONLY)file path=kernel/misc/$(ARCH64)/acpica group=sys mode=0755
644 $(i386_ONLY)file path=kernel/misc/$(ARCH64)/agpmaster group=sys mode=0755
645 file path=kernel/misc/$(ARCH64)/bignum group=sys mode=0755
646 $(i386_ONLY)file path=kernel/misc/$(ARCH64)/bootdev group=sys mode=0755
647 file path=kernel/misc/$(ARCH64)/busra group=sys mode=0755
648 file path=kernel/misc/$(ARCH64)/cardbus group=sys mode=0755
649 file path=kernel/misc/$(ARCH64)/cmlb group=sys mode=0755
650 file path=kernel/misc/$(ARCH64)/consconfig group=sys mode=0755
651 file path=kernel/misc/$(ARCH64)/ctf group=sys mode=0755
652 $(sparc_ONLY)file path=kernel/misc/$(ARCH64)/dada group=sys mode=0755
653 file path=kernel/misc/$(ARCH64)/dls group=sys mode=0755
654 file path=kernel/misc/$(ARCH64)/fssnap_if group=sys mode=0755
655 file path=kernel/misc/$(ARCH64)/gld group=sys mode=0755

```

```

656 file path=kernel/misc/$(ARCH64)/hook group=sys mode=0755
657 file path=kernel/misc/$(ARCH64)/hpcsvc group=sys mode=0755
658 file path=kernel/misc/$(ARCH64)/idmap group=sys mode=0755
659 $(i386_ONLY)file path=kernel/misc/$(ARCH64)/iommulib group=sys mode=0755
660 file path=kernel/misc/$(ARCH64)/ipc group=sys mode=0755
661 file path=kernel/misc/$(ARCH64)/kbtrans group=sys mode=0755
662 file path=kernel/misc/$(ARCH64)/kcf group=sys mode=0755
663 $(i386_ONLY)file path=kernel/misc/$(ARCH64)/kmdbmod group=sys mode=0755
664 file path=kernel/misc/$(ARCH64)/ksocket group=sys mode=0755
665 file path=kernel/misc/$(ARCH64)/mac group=sys mode=0755
666 file path=kernel/misc/$(ARCH64)/mii group=sys mode=0755
667 $(i386_ONLY)file path=kernel/misc/$(ARCH64)/net80211 group=sys mode=0755
668 file path=kernel/misc/$(ARCH64)/neti group=sys mode=0755
669 $(i386_ONLY)file path=kernel/misc/$(ARCH64)/pci_autoconfig group=sys mode=0755
670 $(i386_ONLY)file path=kernel/misc/$(ARCH64)/pcicfg group=sys mode=0755
671 $(i386_ONLY)file path=kernel/misc/$(ARCH64)/pcie group=sys mode=0755
672 file path=kernel/misc/$(ARCH64)/pcihp group=sys mode=0755
673 file path=kernel/misc/$(ARCH64)/pcmcia group=sys mode=0755
674 file path=kernel/misc/$(ARCH64)/rpcsec group=sys mode=0755
675 $(i386_ONLY)file path=kernel/misc/$(ARCH64)/sata group=sys mode=0755
676 file path=kernel/misc/$(ARCH64)/scsi group=sys mode=0755
677 file path=kernel/misc/$(ARCH64)/strplumb group=sys mode=0755
678 $(sparc_ONLY)file path=kernel/misc/$(ARCH64)/swapgeneric group=sys mode=0755
679 file path=kernel/misc/$(ARCH64)/tem group=sys mode=0755
680 file path=kernel/misc/$(ARCH64)/tlimod group=sys mode=0755
681 $(i386_ONLY)file path=kernel/misc/acpica group=sys mode=0755
682 $(i386_ONLY)file path=kernel/misc/agpmaster group=sys mode=0755
683 $(i386_ONLY)file path=kernel/misc/bignum group=sys mode=0755
684 $(i386_ONLY)file path=kernel/misc/bootdev group=sys mode=0755
685 $(i386_ONLY)file path=kernel/misc/busra group=sys mode=0755
686 $(i386_ONLY)file path=kernel/misc/cardbus group=sys mode=0755
687 $(i386_ONLY)file path=kernel/misc/cmlb group=sys mode=0755
688 $(i386_ONLY)file path=kernel/misc/consconfig group=sys mode=0755
689 $(i386_ONLY)file path=kernel/misc/ctf group=sys mode=0755
690 $(i386_ONLY)file path=kernel/misc/dls group=sys mode=0755
691 $(i386_ONLY)file path=kernel/misc/fssnap_if group=sys mode=0755
692 $(i386_ONLY)file path=kernel/misc/gld group=sys mode=0755
693 $(i386_ONLY)file path=kernel/misc/hook group=sys mode=0755
694 $(i386_ONLY)file path=kernel/misc/hpcsvc group=sys mode=0755
695 $(i386_ONLY)file path=kernel/misc/idmap group=sys mode=0755
696 $(i386_ONLY)file path=kernel/misc/iommulib group=sys mode=0755
697 $(i386_ONLY)file path=kernel/misc/ipc group=sys mode=0755
698 $(i386_ONLY)file path=kernel/misc/kbtrans group=sys mode=0755
699 $(i386_ONLY)file path=kernel/misc/kcf group=sys mode=0755
700 $(i386_ONLY)file path=kernel/misc/kmdbmod group=sys mode=0755
701 $(i386_ONLY)file path=kernel/misc/ksocket group=sys mode=0755
702 $(i386_ONLY)file path=kernel/misc/mac group=sys mode=0755
703 $(i386_ONLY)file path=kernel/misc/mii group=sys mode=0755
704 $(i386_ONLY)file path=kernel/misc/net80211 group=sys mode=0755
705 $(i386_ONLY)file path=kernel/misc/neti group=sys mode=0755
706 $(i386_ONLY)file path=kernel/misc/pci_autoconfig group=sys mode=0755
707 $(i386_ONLY)file path=kernel/misc/pcicfg group=sys mode=0755
708 $(i386_ONLY)file path=kernel/misc/pcie group=sys mode=0755
709 $(i386_ONLY)file path=kernel/misc/pcihp group=sys mode=0755
710 $(i386_ONLY)file path=kernel/misc/pcmcia group=sys mode=0755
711 $(i386_ONLY)file path=kernel/misc/rpcsec group=sys mode=0755
712 $(i386_ONLY)file path=kernel/misc/sata group=sys mode=0755
713 $(i386_ONLY)file path=kernel/misc/scsi group=sys mode=0755
714 file path=kernel/misc/scsi_vhci/$(ARCH64)/scsi_vhci_f_asym_emc group=sys \
715     mode=0755
716 file path=kernel/misc/scsi_vhci/$(ARCH64)/scsi_vhci_f_asym_lsi group=sys \
717     mode=0755
718 file path=kernel/misc/scsi_vhci/$(ARCH64)/scsi_vhci_f_asym_sun group=sys \
719     mode=0755
720 file path=kernel/misc/scsi_vhci/$(ARCH64)/scsi_vhci_f_sym group=sys mode=0755
721 file path=kernel/misc/scsi_vhci/$(ARCH64)/scsi_vhci_f_sym_emc group=sys \

```

```

722     mode=0755
723 file path=kernel/misc/scsi_vhci/$(ARCH64)/scsi_vhci_f_sym_hds group=sys \
724     mode=0755
725 file path=kernel/misc/scsi_vhci/$(ARCH64)/scsi_vhci_f_tape group=sys mode=0755
726 file path=kernel/misc/scsi_vhci/$(ARCH64)/scsi_vhci_f_tpgs group=sys mode=0755
727 file path=kernel/misc/scsi_vhci/$(ARCH64)/scsi_vhci_f_tpgs_tape group=sys \
728     mode=0755
729 $(i386_ONLY)file path=kernel/misc/scsi_vhci/scsi_vhci_f_asym_emc group=sys \
730     mode=0755
731 $(i386_ONLY)file path=kernel/misc/scsi_vhci/scsi_vhci_f_asym_lsi group=sys \
732     mode=0755
733 $(i386_ONLY)file path=kernel/misc/scsi_vhci/scsi_vhci_f_asym_sun group=sys \
734     mode=0755
735 $(i386_ONLY)file path=kernel/misc/scsi_vhci/scsi_vhci_f_sym group=sys \
736     mode=0755
737 $(i386_ONLY)file path=kernel/misc/scsi_vhci/scsi_vhci_f_sym_emc group=sys \
738     mode=0755
739 $(i386_ONLY)file path=kernel/misc/scsi_vhci/scsi_vhci_f_sym_hds group=sys \
740     mode=0755
741 $(i386_ONLY)file path=kernel/misc/scsi_vhci/scsi_vhci_f_tape group=sys \
742     mode=0755
743 $(i386_ONLY)file path=kernel/misc/scsi_vhci/scsi_vhci_f_tpgs group=sys \
744     mode=0755
745 $(i386_ONLY)file path=kernel/misc/scsi_vhci/scsi_vhci_f_tpgs_tape group=sys \
746     mode=0755
747 $(i386_ONLY)file path=kernel/misc/strplumb group=sys mode=0755
748 $(i386_ONLY)file path=kernel/misc/tem group=sys mode=0755
749 $(i386_ONLY)file path=kernel/misc/tlimod group=sys mode=0755
750 file path=kernel/sched/$(ARCH64)/SDC group=sys mode=0755
751 file path=kernel/sched/$(ARCH64)/TS group=sys mode=0755
752 file path=kernel/sched/$(ARCH64)/TS_DPTBL group=sys mode=0755
753 $(i386_ONLY)file path=kernel/sched/SDC group=sys mode=0755
754 $(i386_ONLY)file path=kernel/sched/TS group=sys mode=0755
755 $(i386_ONLY)file path=kernel/sched/TS_DPTBL group=sys mode=0755
756 file path=kernel/socketmod/$(ARCH64)/datafilt group=sys mode=0755
757 file path=kernel/socketmod/$(ARCH64)/httpfilt group=sys mode=0755
758 file path=kernel/socketmod/$(ARCH64)/ksslf group=sys mode=0755
759 file path=kernel/socketmod/$(ARCH64)/socksctp group=sys mode=0755
760 file path=kernel/socketmod/$(ARCH64)/trill group=sys mode=0755
761 $(i386_ONLY)file path=kernel/socketmod/datafilt group=sys mode=0755
762 $(i386_ONLY)file path=kernel/socketmod/httpfilt group=sys mode=0755
763 $(i386_ONLY)file path=kernel/socketmod/ksslf group=sys mode=0755
764 $(i386_ONLY)file path=kernel/socketmod/socksctp group=sys mode=0755
765 $(i386_ONLY)file path=kernel/socketmod/trill group=sys mode=0755
766 file path=kernel/strmod/$(ARCH64)/bufmod group=sys mode=0755
767 file path=kernel/strmod/$(ARCH64)/connld group=sys mode=0755
768 file path=kernel/strmod/$(ARCH64)/dedump group=sys mode=0755
769 file path=kernel/strmod/$(ARCH64)/drcompat group=sys mode=0755
770 file path=kernel/strmod/$(ARCH64)/ldterm group=sys mode=0755
771 $(sparc_ONLY)file path=kernel/strmod/$(ARCH64)/ms group=sys mode=0755
772 file path=kernel/strmod/$(ARCH64)/pckt group=sys mode=0755
773 file path=kernel/strmod/$(ARCH64)/pfmod group=sys mode=0755
774 file path=kernel/strmod/$(ARCH64)/pipemod group=sys mode=0755
775 file path=kernel/strmod/$(ARCH64)/ptem group=sys mode=0755
776 file path=kernel/strmod/$(ARCH64)/redirmod group=sys mode=0755
777 file path=kernel/strmod/$(ARCH64)/rpcmod group=sys mode=0755
778 file path=kernel/strmod/$(ARCH64)/timod group=sys mode=0755
779 file path=kernel/strmod/$(ARCH64)/tirdwr group=sys mode=0755
780 file path=kernel/strmod/$(ARCH64)/ttcompat group=sys mode=0755
781 $(sparc_ONLY)file path=kernel/strmod/$(ARCH64)/vuid3ps2 group=sys mode=0755
782 $(i386_ONLY)file path=kernel/strmod/bufmod group=sys mode=0755
783 $(i386_ONLY)file path=kernel/strmod/connld group=sys mode=0755
784 $(i386_ONLY)file path=kernel/strmod/dedump group=sys mode=0755
785 $(i386_ONLY)file path=kernel/strmod/drcompat group=sys mode=0755
786 $(i386_ONLY)file path=kernel/strmod/ldterm group=sys mode=0755
787 $(i386_ONLY)file path=kernel/strmod/pckt group=sys mode=0755

```

```

788 $(i386_ONLY)file path=kernel/strmod/pfmod group=sys mode=0755
789 $(i386_ONLY)file path=kernel/strmod/pipemod group=sys mode=0755
790 $(i386_ONLY)file path=kernel/strmod/ptem group=sys mode=0755
791 $(i386_ONLY)file path=kernel/strmod/redirmod group=sys mode=0755
792 $(i386_ONLY)file path=kernel/strmod/rpcmod group=sys mode=0755
793 $(i386_ONLY)file path=kernel/strmod/timod group=sys mode=0755
794 $(i386_ONLY)file path=kernel/strmod/tirdwr group=sys mode=0755
795 $(i386_ONLY)file path=kernel/strmod/ttcompat group=sys mode=0755
796 file path=kernel/sys/$(ARCH64)/c2audit group=sys mode=0755
797 file path=kernel/sys/$(ARCH64)/doorfs group=sys mode=0755
798 file path=kernel/sys/$(ARCH64)/inst_sync group=sys mode=0755
799 file path=kernel/sys/$(ARCH64)/kaio group=sys mode=0755
800 file path=kernel/sys/$(ARCH64)/msgsys group=sys mode=0755
801 file path=kernel/sys/$(ARCH64)/pipe group=sys mode=0755
802 file path=kernel/sys/$(ARCH64)/portfs group=sys mode=0755
803 file path=kernel/sys/$(ARCH64)/pset group=sys mode=0755
804 file path=kernel/sys/$(ARCH64)/semsys group=sys mode=0755
805 file path=kernel/sys/$(ARCH64)/shmsys group=sys mode=0755
806 $(i386_ONLY)file path=kernel/sys/c2audit group=sys mode=0755
807 $(i386_ONLY)file path=kernel/sys/doorfs group=sys mode=0755
808 $(i386_ONLY)file path=kernel/sys/inst_sync group=sys mode=0755
809 $(i386_ONLY)file path=kernel/sys/kaio group=sys mode=0755
810 $(i386_ONLY)file path=kernel/sys/msgsys group=sys mode=0755
811 $(i386_ONLY)file path=kernel/sys/pipe group=sys mode=0755
812 $(i386_ONLY)file path=kernel/sys/portfs group=sys mode=0755
813 $(i386_ONLY)file path=kernel/sys/pset group=sys mode=0755
814 $(i386_ONLY)file path=kernel/sys/semsys group=sys mode=0755
815 $(i386_ONLY)file path=kernel/sys/shmsys group=sys mode=0755
816 file path=lib/svc/manifest/system/dumpadm.xml group=sys mode=0444
817 file path=lib/svc/manifest/system/intrd.xml group=sys mode=0444
818 file path=lib/svc/manifest/system/scheduler.xml group=sys mode=0444
819 file path=lib/svc/method/svc-dumpadm mode=0555
820 file path=lib/svc/method/svc-intrd mode=0555
821 file path=lib/svc/method/svc-scheduler mode=0555
822 $(sparc_ONLY)file path=usr/share/man/man1m/monitor.1m
823 $(sparc_ONLY)file path=usr/share/man/man1m/obpsym.1m
824 # On SPARC driver/bcscv is Serverbladel specific, and in system/kernel/platform
825 # We keep the manual page generic
826 $(sparc_ONLY)file path=usr/share/man/man7d/dad.7d
827 $(i386_ONLY)file path=usr/share/man/man7d/smbios.7d
828 # Sadly void mouse support is in different packages on different platforms
829 # While kstat(7D) is in SUNWcs, the structures are general
830 hardlink path=kernel/misc/$(ARCH64)/md5 \
831 target=../../../../kernel/crypto/$(ARCH64)/md5
832 hardlink path=kernel/misc/$(ARCH64)/sha1 \
833 target=../../../../kernel/crypto/$(ARCH64)/sha1
834 hardlink path=kernel/misc/$(ARCH64)/sha2 \
835 target=../../../../kernel/crypto/$(ARCH64)/sha2
836 $(i386_ONLY)hardlink path=kernel/misc/md5 target=../../../../kernel/crypto/md5
837 $(i386_ONLY)hardlink path=kernel/misc/sha1 target=../../../../kernel/crypto/sha1
838 $(i386_ONLY)hardlink path=kernel/misc/sha2 target=../../../../kernel/crypto/sha2
839 hardlink path=kernel/socketmod/$(ARCH64)/icmp \
840 target=../../../../kernel/drv/$(ARCH64)/icmp
841 hardlink path=kernel/socketmod/$(ARCH64)/rts \
842 target=../../../../kernel/drv/$(ARCH64)/rts
843 hardlink path=kernel/socketmod/$(ARCH64)/tcp \
844 target=../../../../kernel/drv/$(ARCH64)/tcp
845 hardlink path=kernel/socketmod/$(ARCH64)/udp \
846 target=../../../../kernel/drv/$(ARCH64)/udp
847 $(i386_ONLY)hardlink path=kernel/socketmod/icmp target=../../../../kernel/drv/icmp
848 $(i386_ONLY)hardlink path=kernel/socketmod/rts target=../../../../kernel/drv/rts
849 $(i386_ONLY)hardlink path=kernel/socketmod/tcp target=../../../../kernel/drv/tcp
850 $(i386_ONLY)hardlink path=kernel/socketmod/udp target=../../../../kernel/drv/udp
851 hardlink path=kernel/strmod/$(ARCH64)/arp \
852 target=../../../../kernel/drv/$(ARCH64)/arp
853 hardlink path=kernel/strmod/$(ARCH64)/icmp \

```

```

854 target=../../../../kernel/drv/$(ARCH64)/icmp
855 hardlink path=kernel/strmod/$(ARCH64)/ip \
856 target=../../../../kernel/drv/$(ARCH64)/ip
857 hardlink path=kernel/strmod/$(ARCH64)/ipsecah \
858 target=../../../../kernel/drv/$(ARCH64)/ipsecah
859 hardlink path=kernel/strmod/$(ARCH64)/ipsecsep \
860 target=../../../../kernel/drv/$(ARCH64)/ipsecsep
861 hardlink path=kernel/strmod/$(ARCH64)/keysock \
862 target=../../../../kernel/drv/$(ARCH64)/keysock
863 hardlink path=kernel/strmod/$(ARCH64)/tcp \
864 target=../../../../kernel/drv/$(ARCH64)/tcp
865 hardlink path=kernel/strmod/$(ARCH64)/udp \
866 target=../../../../kernel/drv/$(ARCH64)/udp
867 $(i386_ONLY)hardlink path=kernel/strmod/arp target=../../../../kernel/drv/arp
868 $(i386_ONLY)hardlink path=kernel/strmod/icmp target=../../../../kernel/drv/icmp
869 $(i386_ONLY)hardlink path=kernel/strmod/ip target=../../../../kernel/drv/ip
870 $(i386_ONLY)hardlink path=kernel/strmod/ipsecah \
871 target=../../../../kernel/drv/ipsecah
872 $(i386_ONLY)hardlink path=kernel/strmod/ipsecsep \
873 target=../../../../kernel/drv/ipsecsep
874 $(i386_ONLY)hardlink path=kernel/strmod/keysock \
875 target=../../../../kernel/drv/keysock
876 $(i386_ONLY)hardlink path=kernel/strmod/tcp target=../../../../kernel/drv/tcp
877 $(i386_ONLY)hardlink path=kernel/strmod/udp target=../../../../kernel/drv/udp
878 hardlink path=kernel/sys/$(ARCH64)/autofs \
879 target=../../../../kernel/fs/$(ARCH64)/autofs
880 hardlink path=kernel/sys/$(ARCH64)/rpcmod \
881 target=../../../../kernel/strmod/$(ARCH64)/rpcmod
882 $(i386_ONLY)hardlink path=kernel/sys/autofs target=../../../../kernel/fs/autofs
883 $(i386_ONLY)hardlink path=kernel/sys/rpcmod target=../../../../kernel/strmod/rpcmod
884 legacy pkg=SUNWckr \
885 desc="core kernel software for a specific instruction-set architecture" \
886 name="Core Solaris Kernel (Root)"
887 license cr_Sun license=cr_Sun
888 license lic_CDDL license=lic_CDDL
889 license usr/src/cmd/adb/common/libstand/THIRDPARTYLICENSE \
890 license=usr/src/cmd/adb/common/libstand/THIRDPARTYLICENSE
891 license usr/src/common/bzip2/LICENSE license=usr/src/common/bzip2/LICENSE
892 license usr/src/common/crypto/THIRDPARTYLICENSE.cryptogams \
893 license=usr/src/common/crypto/THIRDPARTYLICENSE.cryptogams
894 $(i386_ONLY)license usr/src/common/crypto/aes/amd64/THIRDPARTYLICENSE.gladman \
895 license=usr/src/common/crypto/aes/amd64/THIRDPARTYLICENSE.gladman
896 $(i386_ONLY)license usr/src/common/crypto/aes/amd64/THIRDPARTYLICENSE.openssl \
897 license=usr/src/common/crypto/aes/amd64/THIRDPARTYLICENSE.openssl
898 license usr/src/common/crypto/ecc/THIRDPARTYLICENSE \
899 license=usr/src/common/crypto/ecc/THIRDPARTYLICENSE
900 $(i386_ONLY)license usr/src/common/crypto/md5/amd64/THIRDPARTYLICENSE \
901 license=usr/src/common/crypto/md5/amd64/THIRDPARTYLICENSE
902 license usr/src/common/mpi/THIRDPARTYLICENSE \
903 license=usr/src/common/mpi/THIRDPARTYLICENSE
904 license usr/src/uts/common/inet/ip/THIRDPARTYLICENSE.rts \
905 license=usr/src/uts/common/inet/ip/THIRDPARTYLICENSE.rts
906 license usr/src/uts/common/inet/tcp/THIRDPARTYLICENSE \
907 license=usr/src/uts/common/inet/tcp/THIRDPARTYLICENSE
908 license usr/src/uts/common/io/THIRDPARTYLICENSE.etheraddr \
909 license=usr/src/uts/common/io/THIRDPARTYLICENSE.etheraddr
910 license usr/src/uts/common/sys/THIRDPARTYLICENSE.icu \
911 license=usr/src/uts/common/sys/THIRDPARTYLICENSE.icu
912 license usr/src/uts/common/sys/THIRDPARTYLICENSE.unicode \
913 license=usr/src/uts/common/sys/THIRDPARTYLICENSE.unicode
914 $(i386_ONLY)license usr/src/uts/intel/io/acpica/THIRDPARTYLICENSE \
915 license=usr/src/uts/intel/io/acpica/THIRDPARTYLICENSE
916 $(i386_ONLY)link path=boot/solaris/bin/root_archive \
917 target=../../../../usr/sbin/root_archive
918 link path=dev/dld target=../devices/pseudo/dld@0:ctl
919 link path=kernel/misc/$(ARCH64)/des \

```

**new/usr/src/pkg/manifests/system-kernel.mf**

15

```
920     target=../../../../kernel/crypto/${ARCH64}/des
921 $(i386_ONLY)link path=kernel/misc/des target=../../../../kernel/crypto/des
```

new/usr/src/pkg/manifests/system-test-ostest.mf

1

\*\*\*\*\*

1553 Sat Nov 22 13:16:58 2014

new/usr/src/pkg/manifests/system-test-ostest.mf

1403 Support deferring TCP accept()  
s

\*\*\*\*\*

1 #  
2 # This file and its contents are supplied under the terms of the  
3 # Common Development and Distribution License ("CDDL"), version 1.0.  
4 # You may only use this file in accordance with the terms of version  
5 # 1.0 of the CDDL.  
6 #  
7 # A full copy of the text of the CDDL should have accompanied this  
8 # source. A copy of the CDDL is also available via the Internet at  
9 # <http://www.illumos.org/license/CDDL>.  
10 #

12 #  
13 # Copyright (c) 2012 by Delphix. All rights reserved.  
14 # Copyright 2014, OmniTI Computer Consulting, Inc. All rights reserved.  
15 # **Copyright 2014, Ryan Zezeski. All rights reserved.**  
16 #

18 set name=pkg.fmri value=pkg:/system/test/ostest@\$(PKGVERS)  
19 set name=pkg.description value="Miscellaneous OS Unit Tests"  
20 set name=pkg.summary value="OS Unit Test Suite"  
21 set name=info.classification \  
22 value=org.opensolaris.category.2008:Development/System  
23 set name=variant.arch value=\$(ARCH)  
24 dir path=opt/os-tests  
25 dir path=opt/os-tests/bin  
26 dir path=opt/os-tests/runfiles  
27 dir path=opt/os-tests/tests  
28 dir path=opt/os-tests/tests/sigqueue  
29 file path=opt/os-tests/README mode=0444  
30 file path=opt/os-tests/bin/ostest mode=0555  
31 file path=opt/os-tests/runfiles/delphix.run mode=0444  
32 file path=opt/os-tests/runfiles/omnios.run mode=0444  
33 file path=opt/os-tests/runfiles/openindiana.run mode=0444  
34 **file path=opt/os-tests/tests/defer\_test mode=0555**  
35 file path=opt/os-tests/tests/poll\_test mode=0555  
36 file path=opt/os-tests/tests/sigqueue/sigqueue\_queue\_size mode=0555  
37 license cr\_Sun license=cr\_Sun  
38 license lic\_CDDL license=lic\_CDDL  
39 depend fmri=system/test/testrunner type=require

new/usr/src/test/os-tests/runfiles/delphix.run

1

\*\*\*\*\*

705 Sat Nov 22 13:16:58 2014

new/usr/src/test/os-tests/runfiles/delphix.run

1403 Support deferring TCP accept()  
\*\*\*\*\*

```
1 #
2 # This file and its contents are supplied under the terms of the
3 # Common Development and Distribution License ("CDDL"), version 1.0.
4 # You may only use this file in accordance with the terms of version
5 # 1.0 of the CDDL.
6 #
7 # A full copy of the text of the CDDL should have accompanied this
8 # source. A copy of the CDDL is also available via the Internet at
9 # http://www.illumos.org/license/CDDL.
10 #
```

```
12 #
13 # Copyright (c) 2012 by Delphix. All rights reserved.
14 #
```

```
16 [DEFAULT]
17 pre =
18 verbose = False
19 quiet = False
20 timeout = 60
21 post =
22 outputdir = /var/tmp/test_results
```

```
24 [/opt/os-tests/tests/defer_test]
```

```
26 [/opt/os-tests/tests/poll_test]
27 user = root
```

```
29 [/opt/os-tests/tests/sigqueue]
30 tests = ['sigqueue_queue_size']
```



new/usr/src/test/os-tests/runfiles/omnios.run

1

\*\*\*\*\*

705 Sat Nov 22 13:16:58 2014

new/usr/src/test/os-tests/runfiles/omnios.run

1403 Support deferring TCP accept()  
\*\*\*\*\*

```
1 #
2 # This file and its contents are supplied under the terms of the
3 # Common Development and Distribution License ("CDDL"), version 1.0.
4 # You may only use this file in accordance with the terms of version
5 # 1.0 of the CDDL.
6 #
7 # A full copy of the text of the CDDL should have accompanied this
8 # source. A copy of the CDDL is also available via the Internet at
9 # http://www.illumos.org/license/CDDL.
10 #
```

```
12 #
13 # Copyright (c) 2012 by Delphix. All rights reserved.
14 #
```

```
16 [DEFAULT]
17 pre =
18 verbose = False
19 quiet = False
20 timeout = 60
21 post =
22 outputdir = /var/tmp/test_results
```

```
24 [/opt/os-tests/tests/defer_test]
```

```
26 [/opt/os-tests/tests/poll_test]
27 user = root
```

```
29 [/opt/os-tests/tests/sigqueue]
30 tests = ['sigqueue_queue_size']
```

new/usr/src/test/os-tests/runfiles/openindiana.run

1

\*\*\*\*\*

705 Sat Nov 22 13:16:58 2014

new/usr/src/test/os-tests/runfiles/openindiana.run

1403 Support deferring TCP accept()  
\*\*\*\*\*

```
1 #
2 # This file and its contents are supplied under the terms of the
3 # Common Development and Distribution License ("CDDL"), version 1.0.
4 # You may only use this file in accordance with the terms of version
5 # 1.0 of the CDDL.
6 #
7 # A full copy of the text of the CDDL should have accompanied this
8 # source. A copy of the CDDL is also available via the Internet at
9 # http://www.illumos.org/license/CDDL.
10 #
```

```
12 #
13 # Copyright (c) 2012 by Delphix. All rights reserved.
14 #
```

```
16 [DEFAULT]
17 pre =
18 verbose = False
19 quiet = False
20 timeout = 60
21 post =
22 outputdir = /var/tmp/test_results
```

```
24 [/opt/os-tests/tests/defer_test]
```

```
26 [/opt/os-tests/tests/poll_test]
27 user = root
```

```
29 [/opt/os-tests/tests/sigqueue]
30 tests = ['sigqueue_queue_size']
```

new/usr/src/test/os-tests/tests/Makefile

1

\*\*\*\*\*

529 Sat Nov 22 13:16:58 2014

new/usr/src/test/os-tests/tests/Makefile

1403 Support deferring TCP accept()s

\*\*\*\*\*

```
1 #
2 # This file and its contents are supplied under the terms of the
3 # Common Development and Distribution License ("CDDL"), version 1.0.
4 # You may only use this file in accordance with the terms of version
5 # 1.0 of the CDDL.
6 #
7 # A full copy of the text of the CDDL should have accompanied this
8 # source. A copy of the CDDL is also available via the Internet at
9 # http://www.illumos.org/license/CDDL.
10 #
11 #
12 #
13 # Copyright (c) 2012 by Delphix. All rights reserved.
14 #
```

```
16 SUBDIRS = poll sigqueue tcpdefer
16 SUBDIRS = poll sigqueue
```

```
18 include $(SRC)/test/Makefile.com
```

new/usr/src/test/os-tests/tests/tcpdefer/Makefile

1

\*\*\*\*\*

1063 Sat Nov 22 13:16:58 2014

new/usr/src/test/os-tests/tests/tcpdefer/Makefile

1403 Support deferring TCP accept()  
s

\*\*\*\*\*

```
1 #
2 # This file and its contents are supplied under the terms of the
3 # Common Development and Distribution License ("CDDL"), version 1.0.
4 # You may only use this file in accordance with the terms of version
5 # 1.0 of the CDDL.
6 #
7 # A full copy of the text of the CDDL should have accompanied this
8 # source. A copy of the CDDL is also available via the Internet at
9 # http://www.illumos.org/license/CDDL.
10 #
11 #
12 #
13 # Copyright (c) 2014 Ryan Zezeski. All rights reserved.
14 #
15 #
16 include $(SRC)/cmd/Makefile.cmd
17 include $(SRC)/test/Makefile.com
18 #
19 PROG = defer_test
20 OBJS = $(PROG:%=%.o)
21 SRCS = $(OBJS:%.o=%.c)
22 #
23 LDLIBS += -lsocket -lnsl -lkstat
24 C99MODE = -xc99=%all
25 #
26 ROOTOPTPKG = $(ROOT)/opt/os-tests
27 TESTDIR = $(ROOTOPTPKG)/tests
28 #
29 CMDS = $(PROG:%=$(TESTDIR)/%)
30 $(CMDS) := FILEMODE = 0555
31 #
32 all: $(PROG)
33 #
34 $(PROG): $(OBJS)
35     $(LINK.c) $(OBJS) -o $@ $(LDLIBS)
36     $(POST_PROCESS)
37 #
38 %.o: ../%.c
39     $(COMPILE.c) $<
40 #
41 install: all $(CMDS)
42 #
43 lint: lint_SRCS
44 #
45 clobber: clean
46     -$(RM) $(PROG)
47 #
48 clean:
49     -$(RM) $(OBJS)
50 #
51 $(CMDS): $(TESTDIR) $(PROG)
52 #
53 $(TESTDIR):
54     $(INS.dir)
55 #
56 $(TESTDIR)/%: %
57     $(INS.file)
```

new/usr/src/test/os-tests/tests/tcpdefer/defer\_test.c

1

```
*****
12531 Sat Nov 22 13:16:58 2014
new/usr/src/test/os-tests/tests/tcpdefer/defer_test.c
1403 Support deferring TCP accept()s
*****
1 /*
2  * This file and its contents are supplied under the terms of the
3  * Common Development and Distribution License ("CDDL"), version 1.0.
4  * You may only use this file in accordance with the terms of version
5  * 1.0 of the CDDL.
6  *
7  * A full copy of the text of the CDDL should have accompanied this
8  * source. A copy of the CDDL is also available via the Internet at
9  * http://www.illumos.org/license/CDDL.
10 */

12 /*
13  * Copyright 2014 Ryan Zeseski
14  */

16 /*
17  * Test TCP deferred accept(). Specifically, test the data/httpfilt
18  * socket filter modules. Deferred accept() allows the kernel to defer
19  * the return of accept() on behalf of the application for the purpose
20  * of making sure data is ready so that the first read() call does not
21  * block or return EAGAIN. This test exercises the code path from the
22  * application level, i.e. entire stack is included.
23  *
24  * datafilt - This module is generic TCP deferment. It assures 1 byte
25  * is ready before accept() returns.
26  *
27  * httpfilt - This module is for HTTP/TCP deferment. It guesses if a
28  * request is valid HTTP and defers accept() until the entire request
29  * is ready, sans the body.
30  */
31 #include <arpa/inet.h>
32 #include <errno.h>
33 #include <kstat.h>
34 #include <netinet/in.h>
35 #include <signal.h>
36 #include <stdarg.h>
37 #include <stdio.h>
38 #include <stdlib.h>
39 #include <string.h>
40 #include <strings.h>
41 #include <sys/socket.h>
42 #include <sys/types.h>
43 #include <sys/wait.h>
44 #include <unistd.h>

46 #define BUFSIZE          1024
47 #define COMMON_HEAD      \
48     "Accept: */*\r\n"    \
49     "Host: test.com\r\n" \
50     "User-Agent: test\r\n" \
51     "\r\n"
52 #define DATAFILT        "datafilt"
53 #define HTTPFILT        "httpfilt"
54 #define NUM_TESTS        12
55 #define PORT             9876

57 /*
58  * Values used to indicate client/server state to each other.
59  */
60 enum State {
61     /* client states */
```

new/usr/src/test/os-tests/tests/tcpdefer/defer\_test.c

2

```
62     CLIENT_CONNECTED,
63     CLIENT_SENT_ALL,
64     CLIENT_SENT_SOME,

66     /* server states */
67     SERVER_LISTENING,
68     SERVER_NO_ACCEPT
69 };

71 /*
72  * Each test is defined as a tdata structure in the test_data array.
73  * Adding a new test is a matter of making a new tdata structure and
74  * bumping NUM_TESTS.
75  *
76  * td_type      The filter to use: HTTPFILT | DATAFILT.
77  *
78  * td_name      The name of the test, written to stdout.
79  *
80  * td_msgs      An array of data msgs that will be sent by the client()
81  *              process.
82  *
83  * td_num_msgs  The number of msgs in td_msgs.
84  */
85 struct tdata {
86     char *td_type;
87     char *td_name;
88     char *td_msgs[2];
89     int td_num_msgs;
90 };

92 static struct tdata test_data[NUM_TESTS] = {
93     {
94         HTTPFILT,
95         "httpfilt_GET",
96         {"GET /test HTTP/1.1\r\n", COMMON_HEAD},
97         2
98     },
99     {
100        HTTPFILT,
101        "httpfilt_GET_LF_only",
102        {"GET /test HTTP/1.1\n",
103         "Accept: */*\n"
104         "Host: test.com\n"
105         "User-Agent: test\n"
106         "\n"},
107        2
108    },
109    {
110        HTTPFILT,
111        "httpfilt_PUT",
112        {"PUT /test HTTP/1.1\r\n", COMMON_HEAD},
113        2
114    },
115    {
116        HTTPFILT,
117        "httpfilt_POST",
118        {"POST /test HTTP/1.1\r\n", COMMON_HEAD},
119        2
120    },
121    {
122        HTTPFILT,
123        "httpfilt_HEAD",
124        {"HEAD /test HTTP/1.1\r\n", COMMON_HEAD},
125        2
126    },
127    {
```

```

128     HTTPFILT,
129     "httpfilt_OPTIONS",
130     {"OPTIONS /test HTTP/1.1\r\n", COMMON_HEAD},
131     2
132 },
133 },
134     HTTPFILT,
135     "httpfilt_TRACE",
136     {"TRACE /test HTTP/1.1\r\n", COMMON_HEAD},
137     2
138 },
139 },
140     HTTPFILT,
141     "httpfilt_CONNECT",
142     {"CONNECT /test HTTP/1.1\r\n", COMMON_HEAD},
143     2
144 },
145 },
146     HTTPFILT,
147     "httpfilt_VERSION-CONTROL",
148     {"VERSION-CONTROL /test HTTP/1.1\r\n", COMMON_HEAD},
149     2
150 },
151 },
152 /*
153  * Verify that a starting space (non-HTTP) is not
154  * deferred.
155  */
156     HTTPFILT,
157     "httpfilt_space",
158     {" "some data..."},
159     1
160 },
161 },
162 /* Verify that a bad method is not deferred. */
163     HTTPFILT,
164     "httpfilt_bad_method",
165     {"badmethod /test HTTP/1.1\r\n"},
166     1
167 },
168 },
169     DATAFILT,
170     "datafilt_one_byte",
171     {"X"},
172     1
173 }
174 };
175
176 static int     debug = 0;
177
178 static int     client(int fd, char **msgs, int num_msgs);
179 static int     server(int fd, char *td_type, char **msgs, int num_msgs);
180
181 static void     dbg(const char *format, ...);
182 static int     msgscmp(char *data, char **msgs, int num_msgs);
183 static int     run(struct tdata tdata);
184 static int     sendstate(int fd, enum State state);
185 static int     waitforstate(int fd, enum State state);
186
187 int
188 main(int argc, char **argv)
189 {
190     int c, rc = 0;
191
192     while ((c = getopt(argc, argv, "d")) != -1) {
193         switch (c) {

```

```

194         case 'd':
195             debug = 1;
196             break;
197         default:
198             break;
199     }
200 }
201
202 for (int i = 0; i < NUM_TESTS; i++)
203     rc += run(test_data[i]);
204
205 return (rc);
206 }
207
208 /*
209  * This test is made up of two cooperating processes which share state
210  * updates over a pipe. The sequence diagram below shows the actions
211  * taken on each side of the socket as well as the state messages. The
212  * numbers represent the order, two steps with the same number happen
213  * concurrently. The state messages act as synchronization points; a
214  * process will block until it receives the expected message. Notice
215  * that a server may accept multiple times before the socket is
216  * created, this proves deferment. A non-deferred socket will accept
217  * on the first try.
218  */
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259

```

| CLI SOCK      | CLIENT                       | SERVER | SERVER SOCK                               |
|---------------|------------------------------|--------|---|
|               |                              |        | (1) listen()                              |
|               | <-----SERVER_LISTENING-----> |        |   |
| (2) connect() | +-----CLIENT_CONNECTED-----> |        | (3) accept()                              |
|               | <-----SERVER_NO_ACCEPT-----> |        |   |
| (4.x) send()  | +-----CLIENT_SENT_SOME-----> |        | (5.x) accept()                            |
|               | <-----SERVER_NO_ACCEPT-----> |        |   |
| (6) close()   | +-----CLIENT_SENT_ALL----->  |        | (6) accept()<br>(7) recv()<br>(8) close() |
|               | =                            | =      |   |

```

static int
run(struct tdata tdata)
{
    int fd[2], rc;
    pid_t child;

    (void) printf("TEST STARTING: %s\n", tdata.td_name);

    if (pipe(fd) != 0) {
        perror("failed to create pipe");
        return (1);
    }

    child = fork();
    if (child == 0) {
        exit(client(fd[1], tdata.td_msgs, tdata.td_num_msgs));
    } else if (child > 0) {
        rc = server(fd[0], tdata.td_type, tdata.td_msgs,
            tdata.td_num_msgs);
        if (rc != 0)

```

```

260         (void) printf("TEST FAILED: %s\n", tdata.td_name);
261     else
262         (void) printf("TEST PASSED: %s\n", tdata.td_name);

264     (void) close(fd[0]);
265     (void) close(fd[1]);
266     (void) Kill(child, SIGKILL);
267 } else {
268     perror("problem with fork()\n");
269     (void) printf("TEST FAILED: %s\n", tdata.td_name);
270     rc = 1;
271 }

273 return (rc);
274 }

276 static int
277 server(int fd, char *filt, char **msgs, int num_msgs)
278 {
279     int         csock, lsock, ndeferred, status;
280     char        buf[BUFSIZE];
281     kstat_ctl_t *kc;
282     kstat_named_t *kn;
283     kstat_t     *ks;
284     struct sockaddr_in addr, cliaddr;
285     socklen_t   clilen;

287     lsock = socket(PF_INET, SOCK_STREAM | SOCK_NONBLOCK, 0);

289     if (setsockopt(lsock, SOL_FILTER, FIL_ATTACH, filt, strlen(filt) + 1) <
290         0) {
291         perror("couldn't set filter");
292         return (1);
293     }

295     if (lsock == -1) {
296         perror("socket");
297         return (1);
298     }

300     bzero(&addr, sizeof (addr));
301     addr.sin_family = AF_INET;
302     addr.sin_addr.s_addr = htonl(INADDR_ANY);
303     addr.sin_port = htons(PORT);
304     if (bind(lsock, (struct sockaddr *)&addr, sizeof (addr)) < 0) {
305         perror("server failed to bind");
306         return (1);
307     }

309     if (listen(lsock, 2) < 0) {
310         perror("listen failed");
311         return (1);
312     }

314     dbg("(1) [server] listen()\n");

316     char filter_name[1024] = "filter_";
317     (void) strcat(filter_name, filt);
318     kc = kstat_open();
319     ks = kstat_lookup(kc, "sockfs", 0, filter_name);
320     (void) kstat_read(kc, ks, NULL);
321     kn = kstat_data_lookup(ks, "ndeferred");
322     ndeferred = kn->value.ui64;

324     if (ndeferred != 0) {
325         (void) fprintf(stderr, "expected 0 deferred conns but got %d\n",

```

```

326         ndeferred);
327     return (1);
328 }

330     if (sendstate(fd, SERVER_LISTENING) != 0)
331         return (1);

333     dbg("---SERVER_LISTENING---->\n");

335     if (waitforstate(fd, CLIENT_CONNECTED) != 0)
336         return (1);

338     clilen = sizeof (cliaddr);
339     csock = accept(lsock, (struct sockaddr *)&cliaddr, &clilen);
340     if ((csock == -1) && (errno != EAGAIN)) {
341         perror("problem accepting");
342         return (1);
343     } else if (csock != -1) {
344         (void) fprintf(stderr,
345             "server accepted before any data sent\n");
346         return (1);
347     }

349     dbg("(3) [server] accept()\n");

351     (void) kstat_chain_update(kc);
352     (void) kstat_read(kc, ks, NULL);
353     kn = kstat_data_lookup(ks, "ndeferred");
354     ndeferred = kn->value.ui64;
355     if (ndeferred != 1) {
356         (void) fprintf(stderr, "expected 1 deferred conns but got %d\n",
357             ndeferred);
358         return (1);
359     }

361     if (sendstate(fd, SERVER_NO_ACCEPT) != 0)
362         return (1);

364     dbg("---SERVER_NO_ACCEPT--->\n");

366     /* The last msg is the final msg. */
367     for (int i = 0; i < num_msgs - 1; i++) {
368         if (waitforstate(fd, CLIENT_SENT_SOME) != 0)
369             return (1);

371         csock = accept(lsock, (struct sockaddr *)&cliaddr, &clilen);
372         if ((csock == -1) && (errno != EAGAIN)) {
373             perror("problem accepting");
374             return (1);
375         } else if (csock != -1) {
376             (void) fprintf(stderr,
377                 "server accepted after some data sent\n");
378             return (1);
379         }

381         dbg("(5.%d) [server] accept()\n", i);

383         if (sendstate(fd, SERVER_NO_ACCEPT) != 0)
384             return (1);

386         dbg("---SERVER_NO_ACCEPT--->\n");

388     }

390     if (waitforstate(fd, CLIENT_SENT_ALL) != 0)
391         return (1);

```

```

393     csock = accept(lsock, (struct sockaddr *)&cliaddr, &clilen);
394     if (csock == -1) {
395         perror("problem accepting");
396         return (1);
397     }
399     dbg("(6) [server] accept()\n");
401     bzero(&buf, BUFSIZE);
402     if (recv(csock, buf, BUFSIZE, 0) == -1) {
403         perror("problem receiving data");
404         return (1);
405     }
407     dbg("(7) [server] recv()\n");
409     if (msgscmp(buf, msgs, num_msgs) != 0) {
410         (void) fprintf(stderr, "data modified\n");
411         return (1);
412     }
414     (void) close(csock);
415     (void) close(lsock);
417     dbg("(8) [server] close()\n");
419     (void) wait(&status);
420     return (status);
421 }
423 static int
424 client(int fd, char **msgs, int num_msgs)
425 {
426     int             sock;
427     size_t          sz;
428     struct sockaddr_in  addr;
430     if (waitforstate(fd, SERVER_LISTENING) != 0)
431         return (1);
433     bzero(&addr, sizeof (addr));
434     addr.sin_family = AF_INET;
435     addr.sin_addr.s_addr = inet_addr("127.0.0.1");
436     addr.sin_port = htons(PORT);
438     sock = socket(AF_INET, SOCK_STREAM, 0);
439     if (sock < 0) {
440         perror("cannot create client socket");
441         return (1);
442     }
444     if (connect(sock, (struct sockaddr *)&addr, sizeof (addr)) < 0) {
445         (void) printf("failed to connect: %d\n", errno);
446         perror("cannot connect to server");
447         return (1);
448     }
450     dbg("(2) [client] connect()\n");
452     if (sendstate(fd, CLIENT_CONNECTED) != 0)
453         return (1);
455     dbg("<---CLIENT_CONNECTED-----\n");
457     if (waitforstate(fd, SERVER_NO_ACCEPT) != 0)

```

```

458         return (1);
460     for (int i = 0; i < num_msgs; i++) {
461         size_t len = strlen(msgs[i]);
462         sz = send(sock, msgs[i], len, 0);
463         if (sz != len) {
464             (void) fprintf(stderr,
465                 "client sent %zu bytes, but should have sent %zu\n",
466                 sz, len);
467             return (1);
468         }
470         dbg("(4.%d) [client] send()\n", i);
472         if (i == num_msgs - 1)
473             break;
475         if (sendstate(fd, CLIENT_SENT_SOME) != 0)
476             return (1);
478         dbg("<---CLIENT_SENT_SOME-----\n");
480         if (waitforstate(fd, SERVER_NO_ACCEPT) != 0)
481             return (1);
482     }
484     if (sendstate(fd, CLIENT_SENT_ALL) != 0)
485         return (1);
487     dbg("<---CLIENT_SENT_ALL-----\n");
488     (void) close(sock);
489     dbg("(6) [client] close()\n");
491     return (0);
492 }
494 /* Send a debug msg to stdout. */
495 static void
496 dbg(const char *format, ...)
497 {
498     va_list args;
500     if (!debug) {
501         return;
502     }
504     va_start(args, format);
505     (void) vprintf(format, args);
506     va_end(args);
507     (void) fflush(stdout);
508 }
510 /*
511  * Test for byte-for-byte equality between data and msgs. Return 0 if
512  * equal, 1 otherwise.
513  */
514 static int
515 msgscmp(char *data, char **msgs, int num_msgs)
516 {
517     for (int i = 0, offset = 0; i < num_msgs; i++) {
518         int len = strlen(msgs[i]);
519         if (memcmp(&data[offset], msgs[i], len) != 0)
520             return (1);
522         offset += len;
523     }

```



```
525     return (0);
526 }

528 /* Send a state across the pipe. */
529 static int sendstate(int fd, enum State state)
530 {
531     size_t sz;

533     sz = write(fd, &state, sizeof (int));
534     return ((sz == sizeof (int)) ? 0 : 1);
535 }

537 /* Block on the given state. */
538 static int waitforstate(int fd, enum State state)
539 {
540     int i;
541     size_t sz;

543     sz = read(fd, &i, sizeof (int));
544     if (sz != sizeof (int)) {
545         (void) fprintf(stderr, "couldn't read int\n");
546         return (1);
547     }

549     if (i != state) {
550         (void) fprintf(stderr,
551             "saw state %d but expected %d\n", i, state);
552         return (1);
553     }

555     return (0);
556 }
```

```

*****
44236 Sat Nov 22 13:16:58 2014
new/usr/src/uts/common/Makefile.files
1403 Support deferring TCP accept()s
*****
1 #
2 # CDDL HEADER START
3 #
4 # The contents of this file are subject to the terms of the
5 # Common Development and Distribution License (the "License").
6 # You may not use this file except in compliance with the License.
7 #
8 # You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9 # or http://www.opensolaris.org/os/licensing.
10 # See the License for the specific language governing permissions
11 # and limitations under the License.
12 #
13 # When distributing Covered Code, include this CDDL HEADER in each
14 # file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 # If applicable, add the following below this CDDL HEADER, with the
16 # fields enclosed by brackets "[]" replaced with your own identifying
17 # information: Portions Copyright [yyyy] [name of copyright owner]
18 #
19 # CDDL HEADER END
20 #
21 #
22 #
23 # Copyright (c) 1991, 2010, Oracle and/or its affiliates. All rights reserved.
24 # Copyright (c) 2013 by Delphix. All rights reserved.
25 # Copyright (c) 2013 by Saso Kiselkov. All rights reserved.
26 # Copyright 2014 Nexenta Systems, Inc. All rights reserved.
27 #
28 #
29 #
30 # This Makefile defines all file modules for the directory uts/common
31 # and its children. These are the source files which may be considered
32 # common to all SunOS systems.
33 #
34 i386_CORE_OBJS += \
35     atomic.o      \
36     avintr.o     \
37     pic.o
38 #
39 sparc_CORE_OBJS +=
40 #
41 COMMON_CORE_OBJS += \
42     beep.o       \
43     bitset.o    \
44     bp_map.o    \
45     brand.o     \
46     cpucaps.o   \
47     cmt.o       \
48     cmt_policy.o \
49     cpu.o       \
50     cpu_event.o \
51     cpu_intr.o  \
52     cpu_pm.o    \
53     cpupart.o   \
54     cap_util.o  \
55     disp.o      \
56     group.o     \
57     kstat_fr.o  \
58     iscsiboot_prop.o \
59     lgrp.o      \
60     lgrp_topo.o \
61     mmapobj.o  \

```

```

62     mutex.o     \
63     page_lock.o \
64     page_retire.o \
65     panic.o    \
66     param.o    \
67     pg.o       \
68     pghw.o     \
69     putnext.o  \
70     rctl_proc.o \
71     rwlock.o   \
72     seg_kmem.o \
73     softint.o  \
74     string.o   \
75     strtol.o   \
76     strtoul.o  \
77     strtoll.o  \
78     strtoull.o \
79     thread_intr.o \
80     vm_page.o  \
81     vm_pagelist.o \
82     zlib_obj.o \
83     clock_tick.o
84 #
85 CORE_OBJS += $(COMMON_CORE_OBJS) $( $(MACH)_CORE_OBJS)
86 #
87 ZLIB_OBJS = zutil.o zmod.o zmod_subr.o \
88     adler32.o crc32.o deflate.o inffast.o \
89     inflate.o inftrees.o trees.o
90 #
91 GENUNIX_OBJS += \
92     access.o     \
93     acl.o        \
94     acl_common.o \
95     adjtime.o    \
96     alarm.o      \
97     aio_subr.o   \
98     auditsys.o   \
99     audit_core.o \
100    audit_zone.o \
101    audit_memory.o \
102    autoconf.o   \
103    avl.o         \
104    bdev_dsort.o \
105    bio.o         \
106    bitmap.o     \
107    blabel.o     \
108    brandsys.o   \
109    bz2blocksort.o \
110    bz2compress.o \
111    bz2decompress.o \
112    bz2randtable.o \
113    bz2bzlib.o   \
114    bz2crctable.o \
115    bz2huffman.o \
116    callb.o      \
117    callout.o    \
118    chdir.o      \
119    chmod.o      \
120    chown.o      \
121    cladm.o      \
122    class.o      \
123    clock.o      \
124    clock_highres.o \
125    clock_realtime.o \
126    close.o      \
127    compress.o   \

```

## new/usr/src/uts/common/Makefile.files

```

128 condvar.o \
129 conf.o \
130 console.o \
131 contract.o \
132 copyops.o \
133 core.o \
134 corectl.o \
135 cred.o \
136 cs_stubs.o \
137 dacf.o \
138 dacf_clnt.o \
139 damap.o \
140 cyclic.o \
141 ddi.o \
142 ddifm.o \
143 ddi_hp_impl.o \
144 ddi_hp_ndi.o \
145 ddi_intr.o \
146 ddi_intr_impl.o \
147 ddi_intr_irm.o \
148 ddi_nodeid.o \
149 ddi_periodic.o \
150 devcfg.o \
151 devcache.o \
152 device.o \
153 devid.o \
154 devid_cache.o \
155 devid_scsi.o \
156 devid_smp.o \
157 devpolicy.o \
158 disp_lock.o \
159 dnlc.o \
160 driver.o \
161 dumpsubr.o \
162 driver_lyr.o \
163 dtrace_subr.o \
164 errorq.o \
165 etheraddr.o \
166 evchannels.o \
167 exacct.o \
168 exacct_core.o \
169 exec.o \
170 exit.o \
171 fbio.o \
172 fcntl.o \
173 fdbuffer.o \
174 fdsync.o \
175 fem.o \
176 ffs.o \
177 fio.o \
178 flock.o \
179 fm.o \
180 fork.o \
181 vpm.o \
182 fs_reparse.o \
183 fs_subr.o \
184 fsflush.o \
185 ftrace.o \
186 getcwd.o \
187 getdents.o \
188 getloadavg.o \
189 getpagesizes.o \
190 getpid.o \
191 gfs.o \
192 rusagesys.o \
193 gid.o \

```

3

## new/usr/src/uts/common/Makefile.files

```

194 groups.o \
195 grow.o \
196 hat_refmod.o \
197 id32.o \
198 id_space.o \
199 inet_ntop.o \
200 instance.o \
201 ioctl.o \
202 ip_cksum.o \
203 issetugid.o \
204 ippconf.o \
205 kpcp.o \
206 kdi.o \
207 kiconv.o \
208 klpd.o \
209 kmem.o \
210 ksyms_snapshot.o \
211 l_strplumb.o \
212 labelsys.o \
213 link.o \
214 list.o \
215 lockstat_subr.o \
216 log_sysevent.o \
217 logsubr.o \
218 lookup.o \
219 lseek.o \
220 ltos.o \
221 lwp.o \
222 lwp_create.o \
223 lwp_info.o \
224 lwp_self.o \
225 lwp_sobj.o \
226 lwp_timer.o \
227 lwpsys.o \
228 main.o \
229 mmapobjs.o \
230 memcntl.o \
231 memstr.o \
232 lgrpsys.o \
233 mknod.o \
234 mkndir.o \
235 mount.o \
236 move.o \
237 msacct.o \
238 multidata.o \
239 nbmlock.o \
240 ndifm.o \
241 nice.o \
242 netstack.o \
243 ntptime.o \
244 nvpair.o \
245 nvpair_alloc_system.o \
246 nvpair_alloc_fixed.o \
247 fnvpair.o \
248 octet.o \
249 open.o \
250 p_online.o \
251 pathconf.o \
252 pathname.o \
253 pause.o \
254 serializer.o \
255 pci_intr_lib.o \
256 pci_cap.o \
257 pcifm.o \
258 pgrp.o \
259 pgrpsys.o \

```

4

## new/usr/src/uts/common/Makefile.files

```

260 pid.o \
261 pkp_hash.o \
262 policy.o \
263 poll.o \
264 pool.o \
265 pool_pset.o \
266 port_subr.o \
267 ppriv.o \
268 printf.o \
269 priocntl.o \
270 priv.o \
271 priv_const.o \
272 proc.o \
273 procset.o \
274 processor_bind.o \
275 processor_info.o \
276 profil.o \
277 project.o \
278 qsort.o \
279 rctl.o \
280 rctlsys.o \
281 readlink.o \
282 refstr.o \
283 rename.o \
284 resolvepath.o \
285 retire_store.o \
286 process.o \
287 rlimit.o \
288 rmap.o \
289 rw.o \
290 rwstlock.o \
291 sad_conf.o \
292 sid.o \
293 sidsys.o \
294 sched.o \
295 schedctl.o \
296 sctp_crc32.o \
297 seg_dev.o \
298 seg_kp.o \
299 seg_kpm.o \
300 seg_map.o \
301 seg_vn.o \
302 seg_spt.o \
303 semaphore.o \
304 sendfile.o \
305 session.o \
306 share.o \
307 shuttle.o \
308 sig.o \
309 sigaction.o \
310 sigaltstack.o \
311 signotify.o \
312 sigpending.o \
313 sigprocmask.o \
314 sigqueue.o \
315 sigsendset.o \
316 sigsuspend.o \
317 sigtimedwait.o \
318 sleepq.o \
319 sock_conf.o \
320 space.o \
321 sscanf.o \
322 stat.o \
323 statfs.o \
324 statvfs.o \
325 stol.o \

```

5

## new/usr/src/uts/common/Makefile.files

```

326 str_conf.o \
327 strcalls.o \
328 stream.o \
329 streamio.o \
330 strext.o \
331 strsubr.o \
332 strsun.o \
333 subr.o \
334 sunddi.o \
335 sunmdi.o \
336 sunndi.o \
337 sunpci.o \
338 sunpm.o \
339 sundlpi.o \
340 suntpi.o \
341 swap_subr.o \
342 swap_vnops.o \
343 symlink.o \
344 sync.o \
345 sysclass.o \
346 sysconfig.o \
347 sysent.o \
348 sysfs.o \
349 systeminfo.o \
350 task.o \
351 taskq.o \
352 tasksys.o \
353 time.o \
354 timer.o \
355 times.o \
356 timers.o \
357 thread.o \
358 tlabel.o \
359 tn timer.o \
360 turnstile.o \
361 tty_common.o \
362 u8_textprep.o \
363 uadmin.o \
364 uconv.o \
365 ucredsys.o \
366 uid.o \
367 umask.o \
368 umount.o \
369 uname.o \
370 unix_bb.o \
371 unlink.o \
372 urw.o \
373 utime.o \
374 utssys.o \
375 uucopy.o \
376 vfs.o \
377 vfs_conf.o \
378 vmem.o \
379 vm_anon.o \
380 vm_as.o \
381 vm_meter.o \
382 vm_pageout.o \
383 vm_pvn.o \
384 vm_rm.o \
385 vm_seg.o \
386 vm_subr.o \
387 vm_swap.o \
388 vm_usage.o \
389 vnode.o \
390 vuid_queue.o \
391 vuid_store.o \

```

6

## new/usr/src/uts/common/Makefile.files

7

```

392          waitq.o \
393          watchpoint.o \
394          yield.o \
395          scsi_confdata.o \
396          xattr.o \
397          xattr_common.o \
398          xdr_mblk.o \
399          xdr_mem.o \
400          xdr.o \
401          xdr_array.o \
402          xdr_refer.o \
403          xhat.o \
404          zone.o

406 #
407 #       Stubs for the stand-alone linker/loader
408 #
409 sparc_GENSTUBS_OBJS = \
410         kobj_stubs.o

412 i386_GENSTUBS_OBJS =

414 COMMON_GENSTUBS_OBJS =

416 GENSTUBS_OBJS += $(COMMON_GENSTUBS_OBJS) $($ (MACH)_GENSTUBS_OBJS)

418 #
419 #       DTrace and DTrace Providers
420 #
421 DTRACE_OBJS += dtrace.o dtrace_isa.o dtrace_asm.o

423 SDT_OBJS += sdt_subr.o

425 PROFILE_OBJS += profile.o

427 SYSTRACE_OBJS += systrace.o

429 LOCKSTAT_OBJS += lockstat.o

431 FASTTRAP_OBJS += fasttrap.o fasttrap_isa.o

433 DCPC_OBJS += dcpc.o

435 #
436 #       Driver (pseudo-driver) Modules
437 #
438 IPP_OBJS += ippctl.o

440 AUDIO_OBJS += audio_client.o audio_ddi.o audio_engine.o \
441         audio_fltdata.o audio_format.o audio_ctrl.o \
442         audio_grc3.o audio_output.o audio_input.o \
443         audio_oss.o audio_sun.o

445 AUDIOEMU10K_OBJS += audioemu10k.o

447 AUDIOENS_OBJS += audioens.o

449 AUDIOVIA823X_OBJS += audiovia823x.o

451 AUDIOVIA97_OBJS += audiovia97.o

453 AUDIO1575_OBJS += audio1575.o

455 AUDIO810_OBJS += audio810.o

457 AUDIOCMI_OBJS += audiocmi.o

```

## new/usr/src/uts/common/Makefile.files

8

```

459 AUDIOCMIHD_OBJS += audiocmihd.o

461 AUDIOHD_OBJS += audiohd.o

463 AUDIOIXP_OBJS += audioixp.o

465 AUDIOLS_OBJS += audiols.o

467 AUDIOP16X_OBJS += audiop16x.o

469 AUDIOPCI_OBJS += audiopci.o

471 AUDIOSOLO_OBJS += audiosolo.o

473 AUDIOTS_OBJS += audiots.o

475 AC97_OBJS += ac97.o ac97_ad.o ac97_alc.o ac97_cmi.o

477 BLKDEV_OBJS += blkdev.o

479 CARDBUS_OBJS += cardbus.o cardbus_hp.o cardbus_cfg.o

481 CONSKBD_OBJS += conskbd.o

483 CONSMS_OBJS += consms.o

485 OLDPTY_OBJS += tty_ptyconf.o

487 PTC_OBJS += tty_pty.o

489 PTSL_OBJS += tty_pts.o

491 PTM_OBJS += ptm.o

493 MII_OBJS += mii.o mii_cicada.o mii_natsemi.o mii_intel.o mii_qualsemi.o \
494         mii_marvell.o mii_realtek.o mii_other.o

496 PTS_OBJS += pts.o

498 PTY_OBJS += ptms_conf.o

500 SAD_OBJS += sad.o

502 MD4_OBJS += md4.o md4_mod.o

504 MD5_OBJS += md5.o md5_mod.o

506 SHA1_OBJS += sha1.o sha1_mod.o

508 SHA2_OBJS += sha2.o sha2_mod.o

510 IPGPC_OBJS += classifierddi.o classifier.o filters.o trie.o table.o \
511         ba_table.o

513 DSCPMK_OBJS += dscpmk.o dscpmkddi.o

515 DLCOSMK_OBJS += dlcosmk.o dlcosmkddi.o

517 FLOWACCT_OBJS += flowacctddi.o flowacct.o

519 TOKENMT_OBJS += tokenmt.o tokenmtddi.o

521 TSWTCL_OBJS += tswtcl.o tswtclddi.o

523 ARP_OBJS += arpddi.o

```

```

525 ICMP_OBJS +=      icmpddi.o
527 ICMP6_OBJS +=    icmp6ddi.o
529 RTS_OBJS +=      rtsddi.o

531 IP_ICMP_OBJS =    icmp.o icmp_opt_data.o
532 IP_RTS_OBJS =     rts.o rts_opt_data.o
533 IP_TCP_OBJS =     tcp.o tcp_fusion.o tcp_opt_data.o tcp_sack.o tcp_stats.o \
534 tcp_misc.o tcp_timers.o tcp_time_wait.o tcp_tpi.o tcp_output.o \
535 tcp_input.o tcp_socket.o tcp_bind.o tcp_cluster.o tcp_tunables.o
536 IP_UDP_OBJS =     udp.o udp_opt_data.o udp_tunables.o udp_stats.o
537 IP_SCTP_OBJS =    sctp.o sctp_opt_data.o sctp_output.o \
538 sctp_init.o sctp_input.o sctp_cookie.o \
539 sctp_conn.o sctp_error.o sctp_snmp.o \
540 sctp_tunables.o sctp_shutdown.o sctp_common.o \
541 sctp_timer.o sctp_heartbeat.o sctp_hash.o \
542 sctp_bind.o sctp_notify.o sctp_asconf.o \
543 sctp_addr.o tn_ipopt.o tnet.o ip_netinfo.o \
544 sctp_misc.o
545 IP_ILB_OBJS =     ilb.o ilb_nat.o ilb_conn.o ilb_alg_hash.o ilb_alg_rr.o

547 IP_OBJS +=        igmp.o ipmp.o ip.o ip6.o ip6_asp.o ip6_if.o ip6_ire.o \
548 ip6_rts.o ip_if.o ip_ire.o ip_listutils.o ip_mrout.o \
549 ip_multi.o ip2mac.o ip_ndp.o ip_rts.o ip_srcid.o \
550 ipddi.o ipdrop.o mi.o nd.o tunables.o optcom.o snmpcom.o \
551 ipsec_loader.o spd.o ipclassifier.o inet_common.o ip_queue.o \
552 squeue.o ip_sadb.o ip_fhtable.o proto_set.o radix.o ip_dummy.o \
553 ip_helper_stream.o ip_tunables.o \
554 ip_output.o ip_input.o ip6_input.o ip6_output.o ip_arp.o \
555 conn_opt.o ip_attr.o ip_dce.o \
556 $(IP_ICMP_OBJS) \
557 $(IP_RTS_OBJS) \
558 $(IP_TCP_OBJS) \
559 $(IP_UDP_OBJS) \
560 $(IP_SCTP_OBJS) \
561 $(IP_ILB_OBJS)

563 IP6_OBJS +=        ip6ddi.o
565 HOOK_OBJS +=       hook.o
567 NETI_OBJS +=       neti_impl.o neti_mod.o neti_stack.o
569 KEYSOCK_OBJS +=    keysockddi.o keysock.o keysock_opt_data.o
571 IPNET_OBJS +=      ipnet.o ipnet_bpf.o
573 SPDSOCK_OBJS +=    spdsockddi.o spdsock.o spdsock_opt_data.o
575 IPSECESP_OBJS +=   ipsecespddi.o ipsecesp.o
577 IPSECAH_OBJS +=    ipsecahddi.o ipsecah.o sadb.o
579 DATAFILT_OBJS +=  datafilt.o
581 HTTPFILT_OBJS +=   httpfilt.o
583 SPPP_OBJS +=        sPPP.o sPPP_dlpi.o sPPP_mod.o s_common.o
585 SPPPTUN_OBJS +=    sppptun.o sppptun_mod.o
587 SPPPASYN_OBJS +=   spppasyn.o spppasyn_mod.o
589 SPPPCOMP_OBJS +=   spppcomp.o spppcomp_mod.o deflate.o bsd-comp.o vjcompress.o \

```

```

590                zlib.o
592 TCP_OBJS +=        tcpddi.o
594 TCP6_OBJS +=       tcp6ddi.o
596 NCA_OBJS +=        ncaddi.o
598 SDP SOCK_MOD_OBJS += sockmod_sdp.o socksdp.o socksdpsubr.o
600 SCTP SOCK_MOD_OBJS += sockmod_sctp.o socksctp.o socksctpsubr.o
602 PFP SOCK_MOD_OBJS += sockmod_pfp.o
604 RDS SOCK_MOD_OBJS += sockmod_rds.o
606 RDS_OBJS +=        rdsddi.o rdssubr.o rds_opt.o rds_ioctl.o
608 RDSIB_OBJS +=       rdsib.o rdsib_ib.o rdsib_cm.o rdsib_ep.o rdsib_buf.o \
609 rdsib_debug.o rdsib_sc.o
611 RDSV3_OBJS +=       af_rds.o rdsrv3_ddi.o bind.o loop.o threads.o connection.o \
612 transport.o cong.o sysctl.o message.o rds_recv.o send.o \
613 stats.o info.o page.o rdma_transport.o ib_ring.o ib_rdma.o \
614 ib_recv.o ib.o ib_send.o ib_sysctl.o ib_stats.o ib_cm.o \
615 rdsrv3_sc.o rdsrv3_debug.o rdsrv3_impl.o rdma.o rdsrv3_af_thr.o
617 ISER_OBJS +=        iser.o iser_cm.o iser_cg.o iser_ib.o iser_idm.o \
618 iser_resource.o iser_xfer.o
620 UDP_OBJS +=         udpddi.o
622 UDP6_OBJS +=        udp6ddi.o
624 SY_OBJS +=          genty.o
626 TCO_OBJS +=         ticots.o
628 TCOO_OBJS +=        ticotsord.o
630 TCL_OBJS +=         ticlts.o
632 TL_OBJS +=          tl.o
634 DUMP_OBJS +=        dump.o
636 BPF_OBJS +=         bpf.o bpf_filter.o bpf_mod.o bpf_dlt.o bpf_mac.o
638 CLONE_OBJS +=       clone.o
640 CN_OBJS +=           cons.o
642 DLD_OBJS +=         dld_drv.o dld_proto.o dld_str.o dld_flow.o
644 DLS_OBJS +=         dls.o dls_link.o dls_mod.o dls_stat.o dls_mgmt.o
646 GLD_OBJS +=         gld.o gldutil.o
648 MAC_OBJS +=         mac.o mac_bcast.o mac_client.o mac_datapath_setup.o mac_flow.o
649 mac_hio.o mac_mod.o mac_ndd.o mac_provider.o mac_sched.o \
650 mac_protect.o mac_soft_ring.o mac_stat.o mac_util.o
652 MAC_6TO4_OBJS +=    mac_6to4.o
654 MAC_ETHER_OBJS +=   mac_ether.o

```

## new/usr/src/uts/common/Makefile.files

11

```

656 MAC_IPV4_OBJS +=      mac_ipv4.o
658 MAC_IPV6_OBJS +=      mac_ipv6.o
660 MAC_WIFI_OBJS +=      mac_wifi.o
662 MAC_IB_OBJS +=        mac_ib.o
664 IPTUN_OBJS +=         iptun_dev.o iptun_ctl.o iptun.o
666 AGGR_OBJS +=          aggr_dev.o aggr_ctl.o aggr_grp.o aggr_port.o \
667                        aggr_send.o aggr_recv.o aggr_lacp.o
669 SOFTMAC_OBJS +=        softmac_main.o softmac_ctl.o softmac_capab.o \
670                        softmac_dev.o softmac_stat.o softmac_pkt.o softmac_fp.o
672 NET80211_OBJS +=        net80211.o net80211_proto.o net80211_input.o \
673                        net80211_output.o net80211_node.o net80211_crypto.o \
674                        net80211_crypto_none.o net80211_crypto_wep.o net80211_ioctl.o \
675                        net80211_crypto_tkip.o net80211_crypto_ccmp.o \
676                        net80211_ht.o
678 VNIC_OBJS +=           vnic_ctl.o vnic_dev.o
680 SIMNET_OBJS +=         simnet.o
682 IB_OBJS +=             ibnex.o ibnex_ioctl.o ibnex_hca.o
684 IBCM_OBJS +=           ibcm_impl.o ibcm_sm.o ibcm_ti.o ibcm_utils.o ibcm_path.o \
685                        ibcm_arp.o ibcm_arp_link.o
687 IBDM_OBJS +=           ibdm.o
689 IBDMA_OBJS +=          ibdma.o
691 IBMF_OBJS +=           ibmf.o ibmf_impl.o ibmf_dr.o ibmf_wqe.o ibmf_ud_dest.o ibmf_mod.o
692                        ibmf_send.o ibmf_recv.o ibmf_handlers.o ibmf_trans.o \
693                        ibmf_timers.o ibmf_msg.o ibmf_utils.o ibmf_rmpp.o \
694                        ibmf_saa.o ibmf_saa_impl.o ibmf_saa_utils.o ibmf_saa_events.o
696 IBTL_OBJS +=           ibtl_impl.o ibtl_util.o ibtl_mem.o ibtl_handlers.o ibtl_qp.o \
697                        ibtl_cq.o ibtl_wr.o ibtl_hca.o ibtl_chan.o ibtl_cm.o \
698                        ibtl_mcg.o ibtl_ibnex.o ibtl_srq.o ibtl_part.o
700 TAVOR_OBJS +=          tavor.o tavor_agents.o tavor_cfg.o tavor_ci.o tavor_cmd.o \
701                        tavor_cq.o tavor_event.o tavor_ioctl.o tavor_misc.o \
702                        tavor_mr.o tavor_qp.o tavor_qpmod.o tavor_rsrc.o \
703                        tavor_srq.o tavor_stats.o tavor_umap.o tavor_wr.o
705 HERMON_OBJS +=         hermon.o hermon_agents.o hermon_cfg.o hermon_ci.o hermon_cmd.o \
706                        hermon_cq.o hermon_event.o hermon_ioctl.o hermon_misc.o \
707                        hermon_mr.o hermon_qp.o hermon_qpmod.o hermon_rsrc.o \
708                        hermon_srq.o hermon_stats.o hermon_umap.o hermon_wr.o \
709                        hermon_fcoib.o hermon_fm.o
711 DAPLT_OBJS +=          daplt.o
713 SOL_OFS_OBJS +=         sol_cma.o sol_ib_cma.o sol_uobj.o \
714                        sol_ofs_debug_util.o sol_ofs_gen_util.o \
715                        sol_kverbs.o
717 SOL_UCMA_OBJS +=        sol_ucma.o
719 SOL_UVERBS_OBJS +=      sol_uverbs.o sol_uverbs_comp.o sol_uverbs_event.o \
720                        sol_uverbs_hca.o sol_uverbs_qp.o

```

## new/usr/src/uts/common/Makefile.files

12

```

722 SOL_UMAD_OBJS +=        sol_umad.o
724 KSTAT_OBJS +=          kstat.o
726 KSYMS_OBJS +=          ksyms.o
728 INSTANCE_OBJS +=       inst_sync.o
730 IWSCN_OBJS +=          iwscons.o
732 LOFI_OBJS +=           lofi.o LzmaDec.o
734 FSSNAP_OBJS +=         fssnap.o
736 FSSNAPIF_OBJS +=       fssnap_if.o
738 MM_OBJS +=             mem.o
740 PHYSMEM_OBJS +=        physmem.o
742 OPTIONS_OBJS +=        options.o
744 WINLOCK_OBJS +=        winlockio.o
746 PM_OBJS +=             pm.o
747 SRN_OBJS +=            srn.o
749 PSEUDO_OBJS +=         pseudonex.o
751 RAMDISK_OBJS +=        ramdisk.o
753 LLC1_OBJS +=           llc1.o
755 USBKBM_OBJS +=         usbkbm.o
757 USBWCM_OBJS +=         usbwcm.o
759 BOFI_OBJS +=           bofi.o
761 HID_OBJS +=            hid.o
763 HWA_RC_OBJS +=         hwarc.o
765 USBSKEL_OBJS +=        usbskel.o
767 USBVC_OBJS +=          usbvc.o usbvc_v412.o
769 HIDPARSER_OBJS +=     hidparser.o
771 USB_AC_OBJS +=         usb_ac.o
773 USB_AS_OBJS +=         usb_as.o
775 USB_AH_OBJS +=         usb_ah.o
777 USBMS_OBJS +=          usbms.o
779 USBPRN_OBJS +=         usbprn.o
781 UGEN_OBJS +=           ugen.o
783 USBSER_OBJS +=         usbser.o usbser_rseq.o
785 USBSACM_OBJS +=        usb sacm.o
787 USBSER_KEYSPAN_OBJS += usbser_keyspan.o keyspan_dsd.o keyspan_pipe.o

```

```

789 USBS49_FW_OBJS += keyspan_49fw.o
791 USBSPRL_OBJS += usbser_pl2303.o pl2303_dsd.o
793 WUSB_CA_OBJS += wusb_ca.o
795 USBFTDI_OBJS += usbser_uftdi.o uftdi_dsd.o
797 USBECM_OBJS += usbecm.o
799 WC_OBJS += wscons.o vcons.o
801 VCONS_CONF_OBJS += vcons_conf.o
803 SCSI_OBJS +=      scsi_capabilities.o scsi_confsubr.o scsi_control.o \
804                scsi_data.o scsi_fm.o scsi_hba.o scsi_reset_notify.o \
805                scsi_resource.o scsi_subr.o scsi_transport.o scsi_watch.o \
806                smp_transport.o
808 SCSI_VHCI_OBJS +=      scsi_vhci.o mpapi_impl.o scsi_vhci_tpgs.o
810 SCSI_VHCI_F_SYM_OBJS +=      sym.o
812 SCSI_VHCI_F_TPGS_OBJS +=      tpgs.o
814 SCSI_VHCI_F_ASYM_SUN_OBJS +=  asym_sun.o
816 SCSI_VHCI_F_SYM_HDS_OBJS +=  sym_hds.o
818 SCSI_VHCI_F_TAPE_OBJS +=      tape.o
820 SCSI_VHCI_F_TPGS_TAPE_OBJS +=  tpgs_tape.o
822 SGEN_OBJS +=      sgen.o
824 SMP_OBJS +=      smp.o
826 SATA_OBJS +=      sata.o
828 USBA_OBJS +=      hcidi.o usba.o usbai.o hubdi.o parser.o genconsole.o \
829                usbai_pipe_mgmt.o usbai_req.o usbai_util.o usbai_register.o \
830                usba_devdb.o usbal0_calls.o usba_uugen.o whcdi.o wa.o
831 USBA_WITHOUT_WUSB_OBJS +=      hcidi.o usba.o usbai.o hubdi.o parser.o gencons
832                usbai_pipe_mgmt.o usbai_req.o usbai_util.o usbai_register.o \
833                usba_devdb.o usbal0_calls.o usba_uugen.o
835 USBAL0_OBJS +=      usbal0.o
837 RSM_OBJS +=      rsm.o rsmka_pathmanager.o rsmka_util.o
839 RSMOPS_OBJS +=      rsmops.o
841 S1394_OBJS +=      t1394.o t1394_errmsg.o s1394.o s1394_addr.o s1394_async.o \
842                s1394_bus_reset.o s1394_cmp.o s1394_csr.o s1394_dev_disc.o \
843                s1394_fa.o s1394_fcp.o \
844                s1394_hotplug.o s1394_isoch.o s1394_misc.o h1394.o nx1394.o
846 HCI1394_OBJS +=      hcil1394.o hcil1394_async.o hcil1394_attach.o hcil1394_buf.o \
847                hcil1394_csr.o hcil1394_detach.o hcil1394_extern.o \
848                hcil1394_ioctl.o hcil1394_isoch.o hcil1394_isr.o \
849                hcil1394_ixl_comp.o hcil1394_ixl_isr.o hcil1394_ixl_misc.o \
850                hcil1394_ixl_update.o hcil1394_misc.o hcil1394_ohci.o \
851                hcil1394_q.o hcil1394_s1394if.o hcil1394_tlabel.o \
852                hcil1394_tlist.o hcil1394_vendor.o

```

```

854 AV1394_OBJS +=      avl1394.o avl1394_as.o avl1394_async.o avl1394_cfgrom.o \
855                avl1394_cmp.o avl1394_fcp.o avl1394_isoch.o avl1394_isoch_chan.o \
856                avl1394_isoch_recv.o avl1394_isoch_xmit.o avl1394_list.o \
857                avl1394_queue.o
859 DCAM1394_OBJS +=      dcam.o dcam_frame.o dcam_param.o dcam_reg.o \
860                dcam_ring_buff.o
862 SCSA1394_OBJS +=      hba.o sbp2_driver.o sbp2_bus.o
864 SBP2_OBJS +=      cfgrom.o sbp2.o
866 PMODEM_OBJS +=      pmodem.o pmodem_cis.o cis.o cis_callout.o cis_handlers.o cis_para
868 DSW_OBJS +=      dsw.o dsw_dev.o ii_tree.o
870 NCALL_OBJS +=      ncall.o \
871                ncall_stub.o
873 RDC_OBJS +=      rdc.o \
874                rdc_dev.o \
875                rdc_io.o \
876                rdc_clnt.o \
877                rdc_prot_xdr.o \
878                rdc_svc.o \
879                rdc_bitmap.o \
880                rdc_health.o \
881                rdc_subr.o \
882                rdc_diskq.o
884 RDCSRV_OBJS +=      rdcsrv.o
886 RDCSTUB_OBJS +=      rdc_stub.o
888 SDBC_OBJS +=      sd_bcache.o \
889                sd_bio.o \
890                sd_conf.o \
891                sd_ft.o \
892                sd_hash.o \
893                sd_io.o \
894                sd_misc.o \
895                sd_pcu.o \
896                sd_tdaemon.o \
897                sd_trace.o \
898                sd_iob_impl0.o \
899                sd_iob_impl1.o \
900                sd_iob_impl2.o \
901                sd_iob_impl3.o \
902                sd_iob_impl4.o \
903                sd_iob_impl5.o \
904                sd_iob_impl6.o \
905                sd_iob_impl7.o \
906                safestore.o \
907                safestore_ram.o
909 NSCTL_OBJS +=      nsctl.o \
910                nsc_cache.o \
911                nsc_disk.o \
912                nsc_dev.o \
913                nsc_freeze.o \
914                nsc_gen.o \
915                nsc_mem.o \
916                nsc_ncallio.o \
917                nsc_power.o \
918                nsc_resv.o \
919                nsc_rmspin.o \

```



## new/usr/src/uts/common/Makefile.files

15

```

920          nsc_solaris.o \
921          nsc_trap.o \
922          nsc_list.o
923 UNISTAT_OBJS += spuni.o \
924          spcs_s_k.o

926 NSKERN_OBJS += nsc_ddi.o \
927          nsc_proc.o \
928          nsc_raw.o \
929          nsc_thread.o \
930          nskernd.o

932 SV_OBJS += sv.o

934 PMCS_OBJS += pmcs_attach.o pmcs_ds.o pmcs_intr.o pmcs_nvram.o pmcs_sata.o \
935          pmcs_scsa.o pmcs_smhba.o pmcs_subr.o pmcs_fwlog.o

937 PMCS8001FW_C_OBJS += pmcs_fw_hdr.o
938 PMCS8001FW_OBJS += $(PMCS8001FW_C_OBJS) SPCBoot.o ila.o firmware.o

940 #
941 #      Build up defines and paths.

943 ST_OBJS += st.o st_conf.o

945 EMLXS_OBJS += emlxs_clock.o emlxs_dfc.o emlxs_dhchap.o emlxs_diag.o \
946          emlxs_download.o emlxs_dump.o emlxs_els.o emlxs_event.o \
947          emlxs_fcf.o emlxs_fcp.o emlxs_fct.o emlxs_hba.o emlxs_ip.o \
948          emlxs_mbox.o emlxs_mem.o emlxs_msg.o emlxs_node.o \
949          emlxs_pkt.o emlxs_sli3.o emlxs_sli4.o emlxs_solaris.o \
950          emlxs_thread.o

952 EMLXS_FW_OBJS += emlxs_fw.o

954 OCE_OBJS += oce_buf.o oce_fm.o oce_gld.o oce_hw.o oce_intr.o oce_main.o \
955          oce_mbx.o oce_mq.o oce_queue.o oce_rx.o oce_stat.o oce_tx.o \
956          oce_utils.o

958 FCT_OBJS += discovery.o fct.o

960 QLT_OBJS += 2400.o 2500.o 8100.o qlt.o qlt_dma.o

962 SRPT_OBJS += srpt_mod.o srpt_ch.o srpt_cm.o srpt_ioc.o srpt_stp.o

964 FCOE_OBJS += fcoe.o fcoe_eth.o fcoe_fc.o

966 FCOET_OBJS += fcoet.o fcoet_eth.o fcoet_fc.o

968 FCOEI_OBJS += fcoei.o fcoei_eth.o fcoei_lv.o

970 ISCSIT_SHARED_OBJS += \
971          iscsit_common.o

973 ISCSIT_OBJS += $(ISCSIT_SHARED_OBJS) \
974          iscsit.o iscsit_tgt.o iscsit_sess.o iscsit_login.o \
975          iscsit_text.o iscsit_isns.o iscsit_radiusauth.o \
976          iscsit_radiuspacket.o iscsit_auth.o iscsit_authclient.o

978 PPPT_OBJS += alua_ic_if.o pppt.o pppt_msg.o pppt_tgt.o

980 STMF_OBJS += lun_map.o stmf.o

982 STMF_SBD_OBJS += sbd.o sbd_scsi.o sbd_pgr.o sbd_zvol.o

984 SYSMMSG_OBJS += sysmsg.o

```

## new/usr/src/uts/common/Makefile.files

16

```

986 SES_OBJS += ses.o ses_sen.o ses_safte.o ses_ses.o

988 TNF_OBJS += tnf_buf.o tnf_trace.o tnf_writer.o trace_init.o \
989          trace_funcs.o tnf_probe.o tnf.o

991 LOGINDMUX_OBJS += logindmux.o

993 DEVINFO_OBJS += devinfo.o

995 DEVPOLL_OBJS += devpoll.o

997 DEVPOOL_OBJS += devpool.o

999 I8042_OBJS += i8042.o

1001 KB8042_OBJS += \
1002          at_keyprocess.o \
1003          kb8042.o \
1004          kb8042_keytables.o

1006 MOUSE8042_OBJS += mouse8042.o

1008 FDC_OBJS += fdc.o

1010 ASY_OBJS += asy.o

1012 ECPP_OBJS += ecpp.o

1014 VUIDM3P_OBJS += vuidmice.o vuidm3p.o

1016 VUIDM4P_OBJS += vuidmice.o vuidm4p.o

1018 VUIDM5P_OBJS += vuidmice.o vuidm5p.o

1020 VUIDPS2_OBJS += vuidmice.o vuidps2.o

1022 HPCSV_C_OBJS += hpcsvc.o

1024 PCIE_MISC_OBJS += pcie.o pcie_fault.o pcie_hp.o pciehpc.o pcishpc.o pcie_pwr.o p

1026 PCIHPNEXUS_OBJS += pcihp.o

1028 OPENEPR_OBJS += openprom.o

1030 RANDOM_OBJS += random.o

1032 PSHOT_OBJS += pshot.o

1034 GEN_DRV_OBJS += gen_drv.o

1036 TCLIENT_OBJS += tclient.o

1038 TPHCI_OBJS += tphci.o

1040 TVHCI_OBJS += tvhci.o

1042 EMUL64_OBJS += emul64.o emul64_bsd.o

1044 FCP_OBJS += fcp.o

1046 FCIP_OBJS += fcip.o

1048 FCSM_OBJS += fcsm.o

1050 FCTL_OBJS += fctl.o

```

```

1052 FP_OBJS += fp.o

1054 QLC_OBJS += ql_api.o ql_debug.o ql_hba_fru.o ql_init.o ql_iocb.o ql_ioctl.o \
1055     ql_isr.o ql_mbx.o ql_nx.o ql_xioctl.o ql_fw_table.o

1057 QLC_FW_2200_OBJS += ql_fw_2200.o

1059 QLC_FW_2300_OBJS += ql_fw_2300.o

1061 QLC_FW_2400_OBJS += ql_fw_2400.o

1063 QLC_FW_2500_OBJS += ql_fw_2500.o

1065 QLC_FW_6322_OBJS += ql_fw_6322.o

1067 QLC_FW_8100_OBJS += ql_fw_8100.o

1069 QLGE_OBJS += qlge.o qlge_dbg.o qlge_flash.o qlge_fm.o qlge_gld.o qlge_mpi.o

1071 ZCONS_OBJS += zcons.o

1073 NV_SATA_OBJS += nv_sata.o

1075 SI3124_OBJS += si3124.o

1077 AHCI_OBJS += ahci.o

1079 PCIIDE_OBJS += pci-ide.o

1081 PCEPP_OBJS += pcepp.o

1083 CPC_OBJS += cpc.o

1085 CPUID_OBJS += cpuid_drv.o

1087 SYSEVENT_OBJS += sysevent.o

1089 BL_OBJS += bl.o

1091 DRM_OBJS += drm_sunmod.o drm_kstat.o drm_agpsupport.o \
1092     drm_auth.o drm_bufs.o drm_context.o drm_dma.o \
1093     drm_drawable.o drm_drv.o drm_fops.o drm_ioctl.o drm_irq.o \
1094     drm_lock.o drm_memory.o drm_msg.o drm_pci.o drm_scatter.o \
1095     drm_cache.o drm_gem.o drm_mm.o ati_pcgart.o

1097 FM_OBJS += devfm.o devfm_machdep.o

1099 RTLS_OBJS +=     rtls.o

1101 #
1102 #             exec modules
1103 #
1104 AOUTEXEC_OBJS +=aout.o

1106 ELFEXEC_OBJS += elf.o elf_notes.o old_notes.o

1108 INTPEXEC_OBJS +=intp.o

1110 SHBINEXEC_OBJS +=shbin.o

1112 JAVAEXEC_OBJS +=java.o

1114 #
1115 #             file system modules
1116 #
1117 AUTOFNS_OBJS += auto_vfsops.o auto_vnops.o auto_subr.o auto_xdr.o auto_sys.o

```

```

1119 CACHEFS_OBJS += cachefs_cnode.o      cachefs_cod.o \
1120     cachefs_dir.o      cachefs_dlog.o  cachefs_filegrp.o \
1121     cachefs_fscache.o  cachefs_ioctl.o cachefs_log.o \
1122     cachefs_module.o \
1123     cachefs_noopc.o    cachefs_resource.o \
1124     cachefs_strict.o \
1125     cachefs_subr.o     cachefs_vfsops.o \
1126     cachefs_vnops.o

1128 DCFS_OBJS +=     dc_vnops.o

1130 DEVFS_OBJS +=     devfs_subr.o      devfs_vfsops.o  devfs_vnops.o

1132 DEV_OBJS  +=     sdev_subr.o      sdev_vfsops.o  sdev_vnops.o \
1133     sdev_ptsops.o  sdev_zvolops.o  sdev_comm.o   \
1134     sdev_profile.o sdev_ncache.o  sdev_netops.o \
1135     sdev_ipnetops.o \
1136     sdev_vtops.o

1138 CTFS_OBJS +=     ctfs_all.o  ctfs_cdir.o  ctfs_ctl.o  ctfs_event.o \
1139     ctfs_latest.o ctfs_root.o  ctfs_sym.o  ctfs_tdir.o  ctfs_tmpl.o

1141 OBJFS_OBJS +=     objfs_vfs.o    objfs_root.o   objfs_common.o \
1142     objfs_odir.o   objfs_data.o

1144 FDFS_OBJS +=     fdops.o

1146 FIFO_OBJS +=     fifosubr.o     fifovnops.o

1148 PIPE_OBJS +=     pipe.o

1150 HSFS_OBJS +=     hsfs_node.o    hsfs_subr.o    hsfs_vfsops.o  hsfs_vnops.o \
1151     hsfs_susp.o    hsfs_rrip.o    hsfs_susp_subr.o

1153 LOFS_OBJS +=     lofs_subr.o    lofs_vfsops.o  lofs_vnops.o

1155 NAMEFS_OBJS +=   namevfs.o      namevno.o

1157 NFS_OBJS  +=     nfs_client.o    nfs_common.o   nfs_dump.o \
1158     nfs_subr.o     nfs_vfsops.o   nfs_vnops.o \
1159     nfs_xdr.o      nfs_sys.o      nfs_strerror.o \
1160     nfs3_vfsops.o  nfs3_vnops.o   nfs3_xdr.o \
1161     nfs_acl_vnops.o nfs_acl_xdr.o   nfs4_vfsops.o \
1162     nfs4_vnops.o   nfs4_xdr.o     nfs4_idmap.o \
1163     nfs4_shadow.o  nfs4_subr.o \
1164     nfs4_attr.o    nfs4_rnode.o   nfs4_client.o \
1165     nfs4_acache.o  nfs4_common.o  nfs4_client_state.o \
1166     nfs4_callback.o nfs4_recovery.o nfs4_client_secinfo.o \
1167     nfs4_client_debug.o nfs_stats.o \
1168     nfs4_acl.o     nfs4_stub_vnops.o  nfs_cmd.o

1170 NFSSRV_OBJS +=   nfs_server.o    nfs_srv.o      nfs3_srv.o \
1171     nfs_acl_srv.o  nfs_auth.o     nfs_auth_xdr.o \
1172     nfs_export.o   nfs_log.o      nfs_log_xdr.o \
1173     nfs4_srv.o     nfs4_state.o   nfs4_srv_attr.o \
1174     nfs4_srv_ns.o  nfs4_db.o      nfs4_srv_deleg.o \
1175     nfs4_deleg_ops.o nfs4_srv_readdir.o nfs4_dispatch.o

1177 SMBSRV_SHARED_OBJS += \
1178     smb_inet.o \
1179     smb_match.o \
1180     smb_msghbuf.o \
1181     smb_oem.o \
1182     smb_string.o \
1183     smb_utf8.o \

```

## new/usr/src/uts/common/Makefile.files

19

```

1184         smb_door_legacy.o \
1185         smb_xdr.o \
1186         smb_token.o \
1187         smb_token_xdr.o \
1188         smb_sid.o \
1189         smb_native.o \
1190         smb_netbios_util.o

1192 SMBSRV_OBJS += $(SMBSRV_SHARED_OBJS) \
1193         smb_acl.o \
1194         smb_alloc.o \
1195         smb_close.o \
1196         smb_common_open.o \
1197         smb_common_transact.o \
1198         smb_create.o \
1199         smb_delete.o \
1200         smb_directory.o \
1201         smb_dispatch.o \
1202         smb_echo.o \
1203         smb_fem.o \
1204         smb_find.o \
1205         smb_flush.o \
1206         smb_fsinfo.o \
1207         smb_fsops.o \
1208         smb_init.o \
1209         smb_kdoor.o \
1210         smb_kshare.o \
1211         smb_kutil.o \
1212         smb_lock.o \
1213         smb_lock_byte_range.o \
1214         smb_locking_andx.o \
1215         smb_logoff_andx.o \
1216         smb_mangle_name.o \
1217         smb_mbuf_marshall.o \
1218         smb_mbuf_util.o \
1219         smb_negotiate.o \
1220         smb_net.o \
1221         smb_node.o \
1222         smb_nt_cancel.o \
1223         smb_nt_create_andx.o \
1224         smb_nt_transact_create.o \
1225         smb_nt_transact_ioctl.o \
1226         smb_nt_transact_notify_change.o \
1227         smb_nt_transact_quota.o \
1228         smb_nt_transact_security.o \
1229         smb_odor.o \
1230         smb_ofile.o \
1231         smb_open_andx.o \
1232         smb_opipe.o \
1233         smb_oplock.o \
1234         smb_pathname.o \
1235         smb_print.o \
1236         smb_process_exit.o \
1237         smb_query_fileinfo.o \
1238         smb_read.o \
1239         smb_rename.o \
1240         smb_sd.o \
1241         smb_seek.o \
1242         smb_server.o \
1243         smb_session.o \
1244         smb_session_setup_andx.o \
1245         smb_set_fileinfo.o \
1246         smb_signing.o \
1247         smb_tree.o \
1248         smb_trans2_create_directory.o \
1249         smb_trans2_dfs.o

```

## new/usr/src/uts/common/Makefile.files

20

```

1250         smb_trans2_find.o \
1251         smb_tree_connect.o \
1252         smb_unlock_byte_range.o \
1253         smb_user.o \
1254         smb_vfs.o \
1255         smb_vops.o \
1256         smb_vss.o \
1257         smb_write.o \
1258         smb_write_raw.o

1260 PCFS_OBJS += pc_alloc.o pc_dir.o pc_node.o pc_subr.o \
1261         pc_vfsops.o pc_vnops.o

1263 PROC_OBJS += prcontrol.o prioctl.o prsubr.o prusr.o \
1264         prvfops.o prvnops.o

1266 MNTFS_OBJS += mntvfops.o mntvnops.o

1268 SHAREFS_OBJS += sharetab.o sharefs_vfsops.o sharefs_vnops.o

1270 SPEC_OBJS += specsubr.o specvfops.o specvnops.o

1272 SOCK_OBJS += socksubr.o sockvfops.o sockparams.o \
1273         socksyscalls.o socktpi.o sockstr.o \
1274         sockcommon_vnops.o sockcommon_subr.o \
1275         sockcommon_sops.o sockcommon.o \
1276         sock_notsupp.o socknotify.o \
1277         nl7c.o nl7curio.o nl7chttp.o nl7clogd.o \
1278         nl7cnca.o sodirect.o sockfilter.o

1280 TMPFS_OBJS += tmp_dir.o tmp_subr.o tmp_tnode.o tmp_vfsops.o \
1281         tmp_vnops.o

1283 UDFS_OBJS += udf_alloc.o udf_bmap.o udf_dir.o \
1284         udf_inode.o udf_subr.o udf_vfsops.o \
1285         udf_vnops.o

1287 UFS_OBJS += ufs_alloc.o ufs_bmap.o ufs_dir.o ufs_xattr.o \
1288         ufs_inode.o ufs_subr.o ufs_tables.o ufs_vfsops.o \
1289         ufs_vnops.o quota.o quotacalls.o quota_ufs.o \
1290         ufs_filio.o ufs_lockfs.o ufs_thread.o ufs_trans.o \
1291         ufs_acl.o ufs_panic.o ufs_directio.o ufs_log.o \
1292         ufs_extvnops.o ufs_snap.o lufs.o lufs_thread.o \
1293         lufs_log.o lufs_map.o lufs_top.o lufs_debug.o \
1294         vscan_drv.o vscan_svc.o vscan_door.o

1296 NSMB_OBJS += smb_conn.o smb_dev.o smb_iod.o smb_pass.o \
1297         smb_rq.o smb_sign.o smb_smb.o smb_subrs.o \
1298         smb_time.o smb_tran.o smb_trantcp.o smb_usr.o \
1299         subr_mchain.o

1301 SMBFS_COMMON_OBJS += smbfs_ntacl.o
1302 SMBFS_OBJS += smbfs_vfsops.o smbfs_vnops.o smbfs_node.o \
1303         smbfs_acl.o smbfs_client.o smbfs_smb.o \
1304         smbfs_subr.o smbfs_subr2.o \
1305         smbfs_rwlock.o smbfs_xattr.o \
1306         $(SMBFS_COMMON_OBJS)

1309 #
1310 #             LVM modules
1311 #
1312 MD_OBJS += md.o md_error.o md_ioctl.o md_mddb.o md_names.o \
1313         md_med.o md_rename.o md_subr.o

1315 MD_COMMON_OBJS = md_convert.o md_crc.o md_revchk.o

```

```

1317 MD_DERIVED_OBJS = metamed_xdr.o meta_basic_xdr.o
1319 SOFTPART_OBJS += sp.o sp_ioctl.o
1321 STRIPE_OBJS += stripe.o stripe_ioctl.o
1323 HOTSPARES_OBJS += hotspares.o
1325 RAID_OBJS += raid.o raid_ioctl.o raid_replay.o raid_resync.o raid_hotspare.o
1327 MIRROR_OBJS += mirror.o mirror_ioctl.o mirror_resync.o
1329 NOTIFY_OBJS += md_notify.o
1331 TRANS_OBJS += mdtrans.o trans_ioctl.o trans_log.o
1333 ZFS_COMMON_OBJS += \
1334     arc.o \
1335     blkptr.o \
1336     bplist.o \
1337     bpobj.o \
1338     bptree.o \
1339     dbuf.o \
1340     ddt.o \
1341     ddt_zap.o \
1342     dmuf.o \
1343     dmuf_diff.o \
1344     dmuf_send.o \
1345     dmuf_object.o \
1346     dmuf_objset.o \
1347     dmuf_traverse.o \
1348     dmuf_tx.o \
1349     dnode.o \
1350     dnode_sync.o \
1351     dsl_bookmark.o \
1352     dsl_dir.o \
1353     dsl_dataset.o \
1354     dsl_deadlist.o \
1355     dsl_destroy.o \
1356     dsl_pool.o \
1357     dsl_synctask.o \
1358     dsl_userhold.o \
1359     dmuf_zfetch.o \
1360     dsl_deleg.o \
1361     dsl_prop.o \
1362     dsl_scan.o \
1363     zfeature.o \
1364     gzip.o \
1365     lz4.o \
1366     lzjb.o \
1367     metaslab.o \
1368     range_tree.o \
1369     refcount.o \
1370     rrwlock.o \
1371     sa.o \
1372     sha256.o \
1373     spa.o \
1374     spa_config.o \
1375     spa_errlog.o \
1376     spa_history.o \
1377     spa_misc.o \
1378     space_map.o \
1379     space_reftree.o \
1380     txg.o \
1381     uberblock.o \

```

```

1382     unique.o \
1383     vdev.o \
1384     vdev_cache.o \
1385     vdev_file.o \
1386     vdev_label.o \
1387     vdev_mirror.o \
1388     vdev_missing.o \
1389     vdev_queue.o \
1390     vdev_raidz.o \
1391     vdev_root.o \
1392     zap.o \
1393     zap_leaf.o \
1394     zap_micro.o \
1395     zfs_byteswap.o \
1396     zfs_debug.o \
1397     zfs_fm.o \
1398     zfs_fuid.o \
1399     zfs_sa.o \
1400     zfs_znode.o \
1401     zil.o \
1402     zio.o \
1403     zio_checksum.o \
1404     zio_compress.o \
1405     zio_inject.o \
1406     zle.o \
1407     zlock.o
1409 ZFS_SHARED_OBJS += \
1410     zfeature_common.o \
1411     zfs_comutil.o \
1412     zfs_deleg.o \
1413     zfs_fletcher.o \
1414     zfs_namecheck.o \
1415     zfs_prop.o \
1416     zpool_prop.o \
1417     zprop_common.o
1419 ZFS_OBJS += \
1420     $(ZFS_COMMON_OBJS) \
1421     $(ZFS_SHARED_OBJS) \
1422     vdev_disk.o \
1423     zfs_acl.o \
1424     zfs_ctldir.o \
1425     zfs_dir.o \
1426     zfs_ioctl.o \
1427     zfs_log.o \
1428     zfs_onexit.o \
1429     zfs_replay.o \
1430     zfs_rlock.o \
1431     zfs_vfsops.o \
1432     zfs_vnops.o \
1433     zvol.o
1435 ZUT_OBJS += \
1436     zut.o
1438 #
1439 #           streams modules
1440 #
1441 BUFMOD_OBJS += bufmod.o
1443 CONNLD_OBJS += connld.o
1445 DEDUMP_OBJS += dedump.o
1447 DRCOMPAT_OBJS += drcompat.o

```

```

1449 LDLINUX_OBJS += ldlinux.o
1451 LDTERM_OBJS += ldterm.o uwidth.o
1453 PCKT_OBJS += pkt.o
1455 PFMOD_OBJS += pfmod.o
1457 PTEM_OBJS += ptem.o
1459 REDIRMOD_OBJS += strredirm.o
1461 TIMOD_OBJS += timod.o
1463 TIRDWR_OBJS += tirdwr.o
1465 TTCOMPAT_OBJS +=ttcompat.o
1467 LOG_OBJS += log.o
1469 PIPEMOD_OBJS += pipemod.o

1471 RPCMOD_OBJS += rpcmod.o      clnt_cots.o      clnt_clts.o \
1472                  clnt_gen.o      clnt_perr.o      mt_rpcinit.o  rpc_calmsg.o \
1473                  rpc_prot.o      rpc_sztypes.o   rpc_subr.o    rpcb_prot.o \
1474                  svc.o           svc_clts.o      svc_gen.o     svc_cots.o \
1475                  rpcsys.o        xdr_sizeof.o   clnt_rdma.o  svc_rdma.o \
1476                  xdr_rdma.o      rdma_subr.o    xdrdma_sizeof.o

1478 KLMMOD_OBJS += klmmod.o \
1479                nlm_impl.o \
1480                nlm_rpc_handle.o \
1481                nlm_dispatch.o \
1482                nlm_rpc_svc.o \
1483                nlm_client.o \
1484                nlm_service.o \
1485                nlm_prot_clnt.o \
1486                nlm_prot_xdr.o \
1487                nlm_rpc_clnt.o \
1488                nsm_addr_clnt.o \
1489                nsm_addr_xdr.o \
1490                sm_inter_clnt.o \
1491                sm_inter_xdr.o

1493 KLMOPS_OBJS += klmops.o

1495 TLIMOD_OBJS += tlimod.o      t_kalloc.o      t_kbind.o      t_kclose.o \
1496                t_kconnect.o  t_kfree.o      t_kgtstate.o   t_kopen.o \
1497                t_krcvdat.o    t_ksndudat.o  t_kspoll.o     t_kunbind.o \
1498                t_kutil.o

1500 RLMOD_OBJS += rlmod.o
1502 TELMOD_OBJS += telmod.o
1504 CRYPTMOD_OBJS += cryptmod.o

1506 KB_OBJS += kbd.o          keytables.o

1508 #
1509 #           ID mapping module
1510 #
1511 IDMAP_OBJS += idmap_mod.o  idmap_kapi.o   idmap_xdr.o    idmap_cache.o

1513 #

```

```

1514 #           scheduling class modules
1515 #
1516 SDC_OBJS += sysdc.o

1518 RT_OBJS += rt.o
1519 RT_DPTBL_OBJS += rt_dptbl.o

1521 TS_OBJS += ts.o
1522 TS_DPTBL_OBJS += ts_dptbl.o

1524 IA_OBJS += ia.o

1526 FSS_OBJS += fss.o

1528 FX_OBJS += fx.o
1529 FX_DPTBL_OBJS += fx_dptbl.o

1531 #
1532 #           Inter-Process Communication (IPC) modules
1533 #
1534 IPC_OBJS += ipc.o

1536 IPCMSG_OBJS += msg.o

1538 IPCSEM_OBJS += sem.o

1540 IPCSHM_OBJS += shm.o

1542 #
1543 #           bignum module
1544 #
1545 COMMON_BIGNUM_OBJS += bignum_mod.o bignumimpl.o

1547 BIGNUM_OBJS += $(COMMON_BIGNUM_OBJS) $(BIGNUM_PSR_OBJS)

1549 #
1550 #           kernel cryptographic framework
1551 #
1552 KCF_OBJS += kcf.o kcf_callprov.o kcf_cbufcall.o kcf_cipher.o kcf_crypto.o \
1553             kcf_cryptoadm.o kcf_ctxops.o kcf_digest.o kcf_dual.o \
1554             kcf_keys.o kcf_mac.o kcf_mech_tabs.o kcf_miscapi.o \
1555             kcf_object.o kcf_policy.o kcf_prov_lib.o kcf_prov_tabs.o \
1556             kcf_sched.o kcf_session.o kcf_sign.o kcf_spi.o kcf_verify.o \
1557             kcf_random.o modes.o ecb.o cbc.o ctr.o ccm.o gcm.o \
1558             fips_random.o

1560 CRYPTOADM_OBJS += cryptoadm.o

1562 CRYPTO_OBJS += crypto.o

1564 DPROV_OBJS += dprov.o

1566 DCA_OBJS += dca.o dca_3des.o dca_debug.o dca_dsa.o dca_kstat.o dca_rng.o \
1567             dca_rsa.o

1569 AESPROV_OBJS += aes.o aes_impl.o aes_modes.o

1571 ARCFOURPROV_OBJS += arcfour.o arcfour_crypt.o

1573 BLOWFISHPROV_OBJS += blowfish.o blowfish_impl.o

1575 ECCPROV_OBJS += ecc.o ec.o ec2_163.o ec2_mont.o ecdecode.o ecl_mult.o \
1576                ecp_384.o ecp_jac.o ec2_193.o ecl.o ecp_192.o ecp_521.o \
1577                ecp_jm.o ec2_233.o ecl_curve.o ecp_224.o ecp_aff.o \
1578                ecp_mont.o ec2_aff.o ec_naf.o ecl_gf.o ecp_256.o mp_gf2m.o \
1579                mpi.o mplogic.o mpmontg.o mpprime.o oid.o \

```

```

1580          secitem.o ec2_test.o ecp_test.o
1582 RSAPROV_OBJS += rsa.o rsa_impl.o pkcs1.o
1584 SWRANDPROV_OBJS += swrand.o
1586 #
1587 #             kernel SSL
1588 #
1589 KSSL_OBJS += kssl.o ksslioct1.o
1591 KSSL_SOCKETFIL_MOD_OBJS += ksslfilter.o ksslapi.o ksslrec.o
1593 #
1594 #             misc. modules
1595 #
1597 C2AUDIT_OBJS += adr.o audit.o audit_event.o audit_io.o \
1598                audit_path.o audit_start.o audit_syscalls.o audit_token.o \
1599                audit_mem.o
1601 PCIC_OBJS += pcic.o
1603 RPCSEC_OBJS += secmod.o          sec_clnt.o          sec_svc.o          sec_gen.o \
1604                auth_des.o         auth_kern.o         auth_none.o        auth_loopb.o \
1605                authdesprt.o       authdesubr.o       authu_prot.o \
1606                key_call.o         key_prot.o         svc_authu.o        svcauthdes.o
1608 RPCSEC_GSS_OBJS += rpcsec_gssmod.o rpcsec_gss.o rpcsec_gss_misc.o \
1609                rpcsec_gss_utils.o svc_rpcsec_gss.o
1611 CONSCONFIG_OBJS += consconfig.o
1613 CONSCONFIG_DACF_OBJS += consconfig_dacf.o consplat.o
1615 TEM_OBJS += tem.o tem_safe.o 6x10.o 7x14.o 12x22.o
1617 KBTRANS_OBJS += \
1618                kbtrans.o \
1619                kbtrans_keytables.o \
1620                kbtrans_polled.o \
1621                kbtrans_streams.o \
1622                usb_keytables.o
1624 KGSSD_OBJS += gssd_clnt_stubs.o gssd_handle.o gssd_prot.o \
1625                gss_display_name.o gss_release_name.o gss_import_name.o \
1626                gss_release_buffer.o gss_release_oid_set.o gen_oids.o gssdmod.o
1628 KGSSD_DERIVED_OBJS = gssd_xdr.o
1630 KGSS_DUMMY_OBJS += dmech.o
1632 KSOCKET_OBJS += ksocket.o ksocket_mod.o
1634 CRYPTO= cksumtypes.o decrypt.o encrypt.o encrypt_length.o etypes.o \
1635          nfold.o verify_checksum.o prng.o block_size.o make_checksum.o \
1636          checksum_length.o hmac.o default_state.o mandatory_sumtype.o
1638 # crypto/des
1639 CRYPTO_DES= f CBC.o f_cksum.o f_parity.o weak_key.o d3 CBC.o ef_crypto.o
1641 CRYPTO_DK= checksum.o derive.o dk_decrypt.o dk_encrypt.o
1643 CRYPTO_ARCFOUR= k5_arcfour.o
1645 # crypto/enc_provider

```

```

1646 CRYPTO_ENC= des.o des3.o arcfour_provider.o aes_provider.o
1648 # crypto/hash_provider
1649 CRYPTO_HASH= hash_kef_generic.o hash_kmd5.o hash_crc32.o hash_kshal.o
1651 # crypto/keyhash_provider
1652 CRYPTO_KEYHASH= descbc.o k5_kmd5des.o k_hmac_md5.o
1654 # crypto/crc32
1655 CRYPTO_CRC32= crc32.o
1657 # crypto/old
1658 CRYPTO_OLD= old_decrypt.o old_encrypt.o
1660 # crypto/raw
1661 CRYPTO_RAW= raw_decrypt.o raw_encrypt.o
1663 K5_KRB= kfree.o copy_key.o \
1664         parse.o init_ctx.o \
1665         ser_adata.o ser_addr.o \
1666         ser_auth.o ser_cksum.o \
1667         ser_key.o ser Princ.o \
1668         serialize.o unparse.o \
1669         ser_actx.o
1671 K5_OS= timeofday.o toffset.o \
1672        init_os_ctx.o c_ustime.o
1674 SEAL= seal.o unseal.o
1676 MECH= delete_sec_context.o \
1677        import_sec_context.o \
1678        gssapi_krb5.o \
1679        k5seal.o k5unseal.o k5sealv3.o \
1680        ser_sctx.o \
1681        sign.o \
1682        util_crypt.o \
1683        util_validate.o util_ordering.o \
1684        util_seqnum.o util_set.o util_seed.o \
1685        wrap_size_limit.o verify.o
1689 MECH_GEN= util_token.o
1692 KGSS_KRB5_OBJS += krb5mech.o \
1693                 $(MECH) $(SEAL) $(MECH_GEN) \
1694                 $(CRYPTO) $(CRYPTO_DES) $(CRYPTO_DK) $(CRYPTO_ARCFOUR) \
1695                 $(CRYPTO_ENC) $(CRYPTO_HASH) \
1696                 $(CRYPTO_KEYHASH) $(CRYPTO_CRC32) \
1697                 $(CRYPTO_OLD) \
1698                 $(CRYPTO_RAW) $(K5_KRB) $(K5_OS)
1700 DES_OBJS += des_crypt.o des_impl.o des_ks.o des_soft.o
1702 DLBOOT_OBJS += bootparam_xdr.o nfs_dlinet.o scan.o
1704 KRTLD_OBJS += kobj_bootflags.o getoptstr.o \
1705              kobj.o kobj_kdi.o kobj_lm.o kobj_subr.o
1707 MOD_OBJS += modctl.o modsubr.o modsysfile.o modconf.o modhash.o
1709 STRPLUMB_OBJS += strplumb.o
1711 CPR_OBJS += cpr_driver.o cpr_dump.o \

```

## new/usr/src/uts/common/Makefile.files

27

```

1712          cpr_main.o cpr_misc.o cpr_mod.o cpr_stat.o \
1713          cpr_uthread.o
1715 PROF_OBJS += prf.o
1717 SE_OBJS += se_driver.o
1719 SYSACCT_OBJS += acct.o
1721 ACCTCTL_OBJS += acctctl.o
1723 EXACCTSYS_OBJS += exacctsys.o
1725 KATIO_OBJS += aio.o
1727 PCMCIA_OBJS += pcmcia.o cs.o cis.o cis_callout.o cis_handlers.o cis_params.o
1729 BUSRA_OBJS += busra.o
1731 PCS_OBJS += pcs.o
1733 PSET_OBJS += pset.o
1735 OHCI_OBJS += ohci.o ohci_hub.o ohci_polled.o
1737 UHCI_OBJS += uhci.o uhciutil.o uhcitygt.o uhcihub.o uhcipolled.o
1739 EHCI_OBJS += ehci.o ehci_hub.o ehci_xfer.o ehci_intr.o ehci_util.o ehci_polled.o
1741 HUBD_OBJS += hubd.o
1743 USB_MID_OBJS += usb_mid.o
1745 USB_IA_OBJS += usb_ia.o
1747 UWBA_OBJS += uwba.o uwbai.o
1749 SCSA2USB_OBJS += scsa2usb.o usb_ms_bulkonly.o usb_ms_cbi.o
1751 HWAHC_OBJS += hwahc.o hwahc_util.o
1753 WUSB_DF_OBJS += wusb_df.o
1754 WUSB_FWMOD_OBJS += wusb_fwmod.o
1756 IPF_OBJS += ip_fil_solaris.o fil.o solaris.o ip_state.o ip_frag.o ip_nat.o \
1757 ip_proxy.o ip_auth.o ip_pool.o ip_hstable.o ip_lookup.o \
1758 ip_log.o misc.o ip_compat.o ip_nat6.o drand48.o
1760 IPD_OBJS += ipd.o
1762 IBD_OBJS += ibd.o ibd_cm.o
1764 EIBNX_OBJS += enx_main.o enx_hdlrs.o enx_ibt.o enx_log.o enx_fip.o \
1765 enx_misc.o enx_q.o enx_ctl.o
1767 EOIB_OBJS += eib_adm.o eib_chan.o eib_cmn.o eib_ctl.o eib_data.o \
1768 eib_fip.o eib_ibt.o eib_log.o eib_mac.o eib_main.o \
1769 eib_rsrc.o eib_svc.o eib_vnic.o
1771 DLPSTUB_OBJS += dlpistub.o
1773 SDP_OBJS += sdpddi.o
1775 TRILL_OBJS += trill.o
1777 CTF_OBJS += ctf_create.o ctf_decl.o ctf_error.o ctf_hash.o ctf_labels.o \

```

## new/usr/src/uts/common/Makefile.files

28

```

1778          ctf_lookup.o ctf_open.o ctf_types.o ctf_util.o ctf_subr.o ctf_mod.o
1780 SMBIOS_OBJS += smb_error.o smb_info.o smb_open.o smb_subr.o smb_dev.o
1782 RPCIB_OBJS += rpcib.o
1784 KMDB_OBJS += kdrv.o
1786 AFE_OBJS += afe.o
1788 BGE_OBJS += bge_main2.o bge_chip2.o bge_kstats.o bge_log.o bge_ndd.o \
1789 bge_atomic.o bge_mii.o bge_send.o bge_rcv2.o bge_mii_5906.o
1791 DMFE_OBJS += dmfe_log.o dmfe_main.o dmfe_mii.o
1793 EFE_OBJS += efe.o
1795 ELXL_OBJS += elxl.o
1797 HME_OBJS += hme.o
1799 IXGB_OBJS += ixgb.o ixgb_atomic.o ixgb_chip.o ixgb_gld.o ixgb_kstats.o \
1800 ixgb_log.o ixgb_ndd.o ixgb_rx.o ixgb_tx.o ixgb_xmii.o
1802 NGE_OBJS += nge_main.o nge_atomic.o nge_chip.o nge_ndd.o nge_kstats.o \
1803 nge_log.o nge_rx.o nge_tx.o nge_xmii.o
1805 PCN_OBJS += pcn.o
1807 RGE_OBJS += rge_main.o rge_chip.o rge_ndd.o rge_kstats.o rge_log.o rge_rxtx.o
1809 URTW_OBJS += urtw.o
1811 ARN_OBJS += arn_hw.o arn_eeeprom.o arn_mac.o arn_calib.o arn_ani.o arn_phy.o arn_
1812 arn_main.o arn_rcv.o arn_xmit.o arn_rc.o
1814 ATH_OBJS += ath_aux.o ath_main.o ath_osdep.o ath_rate.o
1816 ATU_OBJS += atu.o
1818 IPW_OBJS += ipw2100_hw.o ipw2100.o
1820 IWI_OBJS += ipw2200_hw.o ipw2200.o
1822 IWH_OBJS += iwh.o
1824 IWK_OBJS += iwk2.o
1826 IWP_OBJS += iwp.o
1828 MWL_OBJS += mwl.o
1830 MWLFW_OBJS += mwlfw_mode.o
1832 WPI_OBJS += wpi.o
1834 RAL_OBJS += rt2560.o ral_rate.o
1836 RUM_OBJS += rum.o
1838 RWD_OBJS += rt2661.o
1840 RWN_OBJS += rt2860.o
1842 UATH_OBJS += uath.o

```

## new/usr/src/uts/common/Makefile.files

29

```

1844 UATHFW_OBJS += uathfw_mod.o
1846 URAL_OBJS += ural.o
1848 RTW_OBJS += rtw.o smc93cx6.o rtwphy.o rtwphyio.o
1850 ZYD_OBJS += zyd.o zyd_usb.o zyd_hw.o zyd_fw.o
1852 MXFE_OBJS += mxfe.o
1854 MPTSAS_OBJS += mptsas.o mptsas_hash.o mptsas_impl.o mptsas_init.o \
1855     mptsas_raid.o mptsas_smhba.o
1857 SFE_OBJS += sfe.o sfe_util.o
1859 BFE_OBJS += bfe.o
1861 BRIDGE_OBJS += bridge.o
1863 IDM_SHARED_OBJS += base64.o
1865 IDM_OBJS += $(IDM_SHARED_OBJS) \
1866     idm.o idm_impl.o idm_text.o idm_conn_sm.o idm_so.o
1868 VR_OBJS += vr.o
1870 ATGE_OBJS += atge_main.o atge_lle.o atge_mii.o atge_ll.o atge_llc.o
1872 YGE_OBJS = yge.o
1874 SKD_OBJS = skd.o
1876 #
1877 #     Build up defines and paths.
1878 #
1879 LINT_DEFS     += -Dunix
1881 #
1882 #     This duality can be removed when the native and target compilers
1883 #     are the same (or at least recognize the same command line syntax!)
1884 #     It is a bug in the current compilation system that the assembler
1885 #     can't process the -Y I, flag.
1886 #
1887 NATIVE_INC_PATH += $(INC_PATH) $(CCYFLAG)$(UTSBASE)/common
1888 AS_INC_PATH     += $(INC_PATH) -I$(UTSBASE)/common
1889 INCLUDE_PATH    += $(INC_PATH) $(CCYFLAG)$(UTSBASE)/common
1891 PCIEB_OBJS += pcieb.o
1893 #     Chelsio N110 10G NIC driver module
1894 #
1895 CH_OBJS = ch.o glue.o pe.o sge.o
1897 CH_COM_OBJS = ch_mac.o ch_subr.o cspi.o espi.o ixfl1010.o mc3.o mc4.o mc5.o \
1898     mv88e1xxx.o mv88x201x.o my3126.o pm3393.o tp.o ulp.o \
1899     vsc7321.o vsc7326.o xpak.o
1901 #
1902 #     Chelsio Terminator 4 10G NIC nexus driver module
1903 #
1904 CXGBE_FW_OBJS = t4_fw.o t4_cfg.o
1905 CXGBE_COM_OBJS = t4_hw.o common.o
1906 CXGBE_NEX_OBJS = t4_nexus.o t4_sge.o t4_mac.o t4_ioctl.o shared.o \
1907     t4_l2t.o adapter.o osdep.o
1909 #

```

## new/usr/src/uts/common/Makefile.files

30

```

1910 #     Chelsio Terminator 4 10G NIC driver module
1911 #
1912 CXGBE_OBJS = cxgbe.o
1914 #
1915 #     PCI strings file
1916 #
1917 PCI_STRING_OBJS = pci_strings.o
1919 NET_DACF_OBJS += net_dacf.o
1921 #
1922 #     Xframe 10G NIC driver module
1923 #
1924 XGE_OBJS = xge.o xgell.o
1926 XGE_HAL_OBJS = xgehal-channel.o xgehal-fifo.o xgehal-ring.o xgehal-config.o \
1927     xgehal-driver.o xgehal-mm.o xgehal-stats.o xgehal-device.o \
1928     xge-queue.o xgehal-mgmt.o xgehal-mgmtaux.o
1930 #
1931 #     e1000/igb common objs
1932 #
1933 #     Historically e1000g and igb had separate copies of all of the common
1934 #     code. At this time while they are now sharing the same copy of it, they
1935 #     are building it into their own modules which is due to the differences
1936 #     in the osdep and debug portions of their code.
1937 #
1938 E1000API_OBJS += e1000_80003es2lan.o e1000_82540.o e1000_82541.o e1000_82542.o \
1939     e1000_82543.o e1000_82571.o e1000_api.o e1000_ich8lan.o \
1940     e1000_mac.o e1000_manage.o e1000_nvmm.o e1000_phy.o \
1941     e1000_82575.o e1000_i210.o e1000_mbx.o e1000_vf.o
1943 #
1944 #     e1000g module
1945 #
1946 E1000G_OBJS += e1000g_debug.o e1000g_main.o e1000g_alloc.o \
1947     e1000g_tx.o e1000g_rx.o e1000g_stat.o \
1948     e1000g_osdep.o e1000g_workarounds.o
1949
1951 #
1952 #     Intel 82575 1G NIC driver module
1953 #
1954 IGB_OBJS = igb_buf.o igb_debug.o igb_gld.o igb_log.o igb_main.o \
1955     igb_rx.o igb_stat.o igb_tx.o igb_osdep.o
1957 #
1958 #     Intel Pro/100 NIC driver module
1959 #
1960 IPRB_OBJS = iprb.o
1962 #
1963 #     Intel 10GbE PCIE NIC driver module
1964 #
1965 IXGBE_OBJS = ixgbe_82598.o ixgbe_82599.o ixgbe_api.o \
1966     ixgbe_common.o ixgbe_phy.o \
1967     ixgbe_buf.o ixgbe_debug.o ixgbe_gld.o \
1968     ixgbe_log.o ixgbe_main.o \
1969     ixgbe_osdep.o ixgbe_rx.o ixgbe_stat.o \
1970     ixgbe_tx.o ixgbe_x540.o ixgbe_mbx.o
1972 #
1973 #     NIU 10G/1G driver module
1974 #
1975 NXGE_OBJS = nxge_mac.o nxge_ipp.o nxge_rxdma.o \

```



## new/usr/src/uts/common/Makefile.files

31

```

1976          nxge_txdma.o nxge_txc.o nxge_main.o          \
1977          nxge_hw.o nxge_fzc.o nxge_virtual.o          \
1978          nxge_send.o nxge_classify.o nxge_fflp.o       \
1979          nxge_fflp_hash.o nxge_ndd.o nxge_kstats.o     \
1980          nxge_zcp.o nxge_fm.o nxge_espc.o nxge_hv.o    \
1981          nxge_hio.o nxge_hio_guest.o nxge_intr.o      \

1983 NXGE_NPI_OBJS = \
1984          napi.o napi_mac.o napi_ipp.o                  \
1985          napi_txdma.o napi_rxdma.o napi_txc.o          \
1986          napi_zcp.o napi_espc.o napi_fflp.o           \
1987          napi_vir.o

1989 NXGE_HCALL_OBJS = \
1990          nxge_hcall.o

1992 #
1993 # Virtio modules
1994 #

1996 # Virtio core
1997 VIRTIO_OBJS = virtio.o

1999 # Virtio block driver
2000 VIOBLK_OBJS = vioblk.o

2002 #
2003 #          kiconv modules
2004 #
2005 KICONV_EMEA_OBJS += kiconv_emea.o

2007 KICONV_JA_OBJS += kiconv_ja.o

2009 KICONV_KO_OBJS += kiconv_cck_common.o kiconv_ko.o

2011 KICONV_SC_OBJS += kiconv_cck_common.o kiconv_sc.o

2013 KICONV_TC_OBJS += kiconv_cck_common.o kiconv_tc.o

2015 #
2016 #          AAC module
2017 #
2018 AAC_OBJS = aac.o aac_ioctl.o

2020 #
2021 #          sdcard modules
2022 #
2023 SDA_OBJS =          sda_cmd.o sda_host.o sda_init.o sda_mem.o sda_mod.o sda_slot.o
2024 SDHOST_OBJS =      sdhost.o

2026 #
2027 #          hxge 10G driver module
2028 #
2029 HXGE_OBJS =          hxge_main.o hxge_vmac.o hxge_send.o          \
2030          hxge_txdma.o hxge_rxdma.o hxge_virtual.o          \
2031          hxge_fm.o hxge_fzc.o hxge_hw.o hxge_kstats.o       \
2032          hxge_ndd.o hxge_pfc.o                               \
2033          hpi.o hpi_vmac.o hpi_rxdma.o hpi_txdma.o          \
2034          hpi_vir.o hpi_pfc.o

2036 #
2037 #          MEGARAID_SAS module
2038 #
2039 MEGA_SAS_OBJS = megaraid_sas.o

2041 #

```

## new/usr/src/uts/common/Makefile.files

32

```

2042 #          MR_SAS module
2043 #
2044 MR_SAS_OBJS = ld_pd_map.o mr_sas.o mr_sas_tbolt.o mr_sas_list.o

2046 #
2047 #          CPQARY3 module
2048 #
2049 CPQARY3_OBJS = cpqary3.o cpqary3_noe.o cpqary3_talk2ctrl.o \
2050          cpqary3_isr.o cpqary3_transport.o cpqary3_mem.o \
2051          cpqary3_scsi.o cpqary3_util.o cpqary3_ioctl.o \
2052          cpqary3_bd.o

2054 #
2055 #          ISCSI_INITIATOR module
2056 #
2057 ISCSI_INITIATOR_OBJS = chap.o iscsi_io.o iscsi_thread.o \
2058          iscsi_ioctl.o iscsid.o iscsi.o \
2059          iscsi_login.o isns_client.o iscsiAuthClient.o \
2060          iscsi_lun.o iscsiAuthClientGlue.o \
2061          iscsi_net.o nvfile.o iscsi_cmd.o \
2062          iscsi_queue.o persistent.o iscsi_conn.o \
2063          iscsi_sess.o radius_auth.o iscsi_crc.o \
2064          iscsi_stats.o radius_packet.o iscsi_doorctl.o \
2065          iscsi_targetparam.o utils.o kifconf.o

2067 #
2068 #          ntxn 10Gb/1Gb NIC driver module
2069 #
2070 NTXN_OBJS =          unm_nic_init.o unm_gem.o unm_nic_hw.o unm_ndd.o \
2071          unm_nic_main.o unm_nic_isr.o unm_nic_ctx.o niu.o

2073 #
2074 #          Myricom 10Gb NIC driver module
2075 #
2076 MYRI10GE_OBJS = myril0ge.o myril0ge_lro.o

2078 #          nulldriver module
2079 #
2080 NULLDRIVER_OBJS =          nulldriver.o

2082 TPM_OBJS =          tpm.o tpm_hcall.o

```

```

*****
2562 Sat Nov 22 13:16:58 2014
new/usr/src/uts/common/inet/sockmods/datafilt.c
1403 Support deferring TCP accept()s
*****
1 /*
2  * This file and its contents are supplied under the terms of the
3  * Common Development and Distribution License ("CDDL"), version 1.0.
4  * You may only use this file in accordance with the terms of version
5  * 1.0 of the CDDL.
6  *
7  * A full copy of the text of the CDDL should have accompanied this
8  * source. A copy of the CDDL is also available via the Internet at
9  * http://www.illumos.org/license/CDDL.
10 */

12 /*
13  * Copyright (c) 2012, OmniTI Computer Consulting, Inc. All rights reserved.
14  */

16 /*
17  * This file implements a socketfilter used to deter TCP connections.
18  * To defer a connection means to delay the return of accept(3SOCKET)
19  * until at least one byte is ready to be read(2). This filter may be
20  * applied automatically or programmatically through the use of
21  * soconfig(1M) and setsockopt(3SOCKET).
22  */

24 #include <sys/kmem.h>
25 #include <sys/system.h>
26 #include <sys/stropts.h>
27 #include <sys/strsun.h>
28 #include <sys/socketvar.h>
29 #include <sys/socketfilter.h>
30 #include <sys/note.h>
31 #include <sys/taskq.h>

33 #define DATAFILT_MODULE "datafilt"

35 static struct modlmisc dataf_modlmisc = {
36     &mod_miscops,
37     "Kernel data-ready socket filter"
38 };

40 static struct modlinkage dataf_modlinkage = {
41     MODREV_1,
42     &dataf_modlmisc,
43     NULL
44 };

46 static sof_rval_t
47 dataf_attach_passive_cb(sof_handle_t handle, sof_handle_t ph,
48     void *parg, struct sockaddr *laddr, socklen_t laddrlen,
49     struct sockaddr *faddr, socklen_t faddrlen, void **cookiep)
50 {
51     _NOTE(ARGUNUSED(handle, ph, parg, laddr, laddrlen, faddr, faddrlen,
52     cookiep));
53     return (SOF_RVAL_DEFER);
54 }

56 static void
57 dataf_detach_cb(sof_handle_t handle, void *cookie, cred_t *cr)
58 {
59     _NOTE(ARGUNUSED(handle, cookie, cr));
60 }

```

```

62 static mblk_t *
63 dataf_data_in_cb(sof_handle_t handle, void *cookie, mblk_t *mp, int flags,
64     size_t *lenp)
65 {
66     _NOTE(ARGUNUSED(handle, cookie, mp, flags, lenp));

68     if (mp != NULL && MBLKL(mp) > 0) {
69         sof_newconn_ready(handle);
70         sof_bypass(handle);
71     }

73     return (mp);
74 }

76 static sof_ops_t dataf_ops = {
77     .sofop_attach_passive = dataf_attach_passive_cb,
78     .sofop_detach = dataf_detach_cb,
79     .sofop_data_in = dataf_data_in_cb
80 };

82 int
83 _init(void)
84 {
85     int err;

87     err = sof_register(SOF_VERSION, DATAFILT_MODULE, &dataf_ops, 0);
88     if (err != 0)
89         return (err);
90     if ((err = mod_install(&dataf_modlinkage)) != 0)
91         (void) sof_unregister(DATAFILT_MODULE);

93     return (err);
94 }

96 int
97 _fini(void)
98 {
99     int err;

101     if ((err = sof_unregister(DATAFILT_MODULE)) != 0)
102         return (err);

104     return (mod_remove(&dataf_modlinkage));
105 }

107 int
108 _info(struct modinfo *modinfop)
109 {
110     return (mod_info(&dataf_modlinkage, modinfop));
111 }

```

```

*****
7807 Sat Nov 22 13:16:58 2014
new/usr/src/uts/common/inet/sockmods/httpfilt.c
1403 Support deferring TCP accept()s
*****
1 /*
2  * This file and its contents are supplied under the terms of the
3  * Common Development and Distribution License ("CDDL"), version 1.0.
4  * You may only use this file in accordance with the terms of version
5  * 1.0 of the CDDL.
6  *
7  * A full copy of the text of the CDDL should have accompanied this
8  * source. A copy of the CDDL is also available via the Internet at
9  * http://www.illumos.org/license/CDDL.
10 */
12 /*
13  * Copyright (c) 2012, OmniTI Computer Consulting, Inc. All rights reserved.
14  * Copyright 2014 Ryan Zezeski
15 */
17 /*
18  * This filter implements a socketfilter used to defer HTTP connections.
19  * To defer a connection means to delay the return of accept(3SOCKET)
20  * until the entire HTTP request is ready (or majority in case of PUT
21  * and POST). This filter may be applied automatically or
22  * programmatically through the use of soconfig(1M) and
23  * setsockopt(3SOCKET).
24  *
25  * The point of this filter is to defer connections, not parse them.
26  * It is up to the webserver to service the request. This filter
27  * accepts both valid and invalid HTTP requests. A valid HTTP request
28  * is deferred until the CRLF sequence that separates the headers from
29  * the optional body is seen. An invalid request is one that doesn't
30  * start with a valid HTTP verb. In either case the connection will be
31  * accepted once MAX_DEFERRED_BYTES have been seen.
32  *
33  * This filter does its best to be accommodating to non-standard HTTP
34  * verbs, instead of restricting deferment to a small subset of verbs.
35  * It accepts a subset of the BNF specified in RFC2616: any sequence
36  * of capital ASCII letters, hyphen, or underscore.
37  *
38  * [A-Z_]+
39  *
40  * This means that the filter can be tricked into thinking a request
41  * is HTTP when it is not. E.g. a request of 'GET ...' where <...>
42  * contains no LFCR or LFLF sequence. In this case, if the max bytes
43  * are never reached, the connection will stay deferred until either
44  * the client closes or the socketfilter system feels pressure and
45  * schedules this connection for close (see so_newconn() and
46  * sof_sonode_drop_deferred()).
47  *
48  * Note that req_fsm() looks for a sequence of LFCR or LFLF to
49  * indicate a complete request. This is a deliberate deviation from
50  * strict CRLFCRLF. Not all clients respect RFC2616. Popular
51  * webserver, such as nginx, will accept just LF, with no need for
52  * CR. The sequences LFCR and LFLF accept strict and non-strict
53  * clients.
54 */
56 #include <sys/kmem.h>
57 #include <sys/system.h>
58 #include <sys/stropts.h>
59 #include <sys/strsun.h>
60 #include <sys/socketvar.h>
61 #include <sys/sockfilter.h>

```

```

62 #include <sys/note.h>
63 #include <sys/taskq.h>
65 #define HTTPFILT_MODULE      "httpfilt"
66 #define MAX_DEFERRED_BYTES  8192
68 enum req_state { REQS_START, REQS_VERB, REQS_HEAD, REQS_CRLF, REQS_END };
70 /*
71  * This structure is created during initialization and then passed to
72  * all callbacks so that state may be kept as message blocks arrive.
73  * This information determines when a connection will be accepted at
74  * the user socket level.
75 */
76 typedef struct httpf {
77     size_t      httpf_bytes_in; /* bytes read */
78     enum req_state httpf_rs;    /* request state */
79 } httpf_t;
81 static struct modlmisc httpf_modlmisc = {
82     &mod_miscops,
83     "Kernel HTTP socket filter"
84 };
86 static struct modlinkage httpf_modlinkage = {
87     MODREV_1,
88     &httpf_modlmisc,
89     NULL
90 };
92 /*
93  * An FSM to determine how long the socket shall remain deferred.
94  * Given the current state and char, return the next state. The
95  * REQS_END indicates that the socket should be accepted.
96  *
97  *
98  *
99  * +-----+
100  * | START |
101  * +-----+
102  * | [A-Z_-] |
103  * |         |
104  * |         |
105  * | [^A-Z_-] V V |
106  * | +-----+ |
107  * | | VERB | |
108  * | +-----+ |
109  * |         |
110  * |         |
111  * |         |
112  * |         |
113  * |         |
114  * |         |
115  * |         |
116  * |         |
117  * |         |
118  * |         |
119  * |         |
120  * |         |
121  * |         |
122  * |         |
123  * |         |
124  * |         |
125  * |         |
126  * |         |
127  * +-----+

```

A request must start with an HTTP verb. This FSM will accept any verb consisting of all capital ASCII letters, hyphen, and underscore. Any violation will move the FSM into the END state.

This state transverses the rest of the the request line and all the headers. A line feed potentially indicates the end of the headers and thus the CRLF state is entered.

If an LFCR or LFLF sequence is seen then all headers have been seen and the end of the request has been reached (the optional body is not considered for deferment). If the end of the request has not been reached then return to HEAD.

```

128 * +->|      END      |      Either the end of the request has been
129 * +-----+      |      reached or it's not an HTTP request at
130 *                |      all. In any case the connection is
131 *                |      accepted.
132 */
133 static int
134 req_fsm(enum req_state s, char ch)
135 {
136     switch (s) {
137     case REQS_START:
138         if ((ch >= 'A' && ch <= 'Z') || ch == '-' || ch == '_')
139             return (REQS_VERB);
140         else
141             return (REQS_END);
142
143     case REQS_VERB:
144         if ((ch >= 'A' && ch <= 'Z') || ch == '-' || ch == '_')
145             return (REQS_VERB);
146         else if (ch == ' ')
147             return (REQS_HEAD);
148         else
149             return (REQS_END);
150
151     case REQS_HEAD:
152         return (ch == '\n' ? REQS_CRLF : REQS_HEAD);
153
154     case REQS_CRLF:
155         return ((ch == '\r' || ch == '\n') ? REQS_END : REQS_HEAD);
156
157     case REQS_END:
158         /*
159          * Should never get here, make the compiler happy.
160          */
161         return (REQS_END);
162     }
163
164     /* Make the compiler happy. */
165     return (REQS_END);
166 }
167
168 /*
169  * Return 0 to continue processing, 1 to accept the connection.
170  */
171 static int
172 httpf_process_input(httpf_t *httpf, mblk_t *mp)
173 {
174     int i, dlen = MBLKL(mp);
175
176     for (i = 0; i < dlen; i++, httpf->httpf_bytes_in++) {
177         httpf->httpf_rs = req_fsm(httpf->httpf_rs, mp->b_rptr[i]);
178         if (httpf->httpf_rs == REQS_END)
179             return (1);
180     }
181
182     if (httpf->httpf_bytes_in >= MAX_DEFERRED_BYTES)
183         return (1);
184
185     return (0);
186 }
187
188 static sof_rval_t
189 httpf_attach_passive_cb(sof_handle_t handle, sof_handle_t ph,
190 void *parg, struct sockaddr *laddr, socklen_t laddrlen,
191 struct sockaddr *faddr, socklen_t faddrlen, void **cookiep)
192 {
193     httpf_t *new;

```

```

195     _NOTE(ARGUNUSED(handle, ph, parg, faddr, faddrlen, laddr, laddrlen));
196
197     new = kmem_zalloc(sizeof (httpf_t), KM_NOSLEEP | KM_NORMALPRI);
198     if (new == NULL)
199         return (SOF_RVAL_ENOMEM);
200
201     new->httpf_bytes_in = 0;
202     new->httpf_rs = REQS_START;
203     *cookiep = new;
204
205     return (SOF_RVAL_DEFER);
206 }
207
208 static void
209 httpf_detach_cb(sof_handle_t handle, void *cookie, cred_t *cr)
210 {
211     httpf_t *httpf = (httpf_t *)cookie;
212
213     _NOTE(ARGUNUSED(handle, cr));
214
215     if (httpf == NULL)
216         return;
217
218     kmem_free(httpf, sizeof (httpf_t));
219 }
220
221 static mblk_t *
222 httpf_data_in_cb(sof_handle_t handle, void *cookie, mblk_t *mp, int flags,
223 size_t *lenp)
224 {
225     httpf_t *httpf = cookie;
226
227     _NOTE(ARGUNUSED(flags, lenp));
228
229     if (httpf == NULL) {
230         sof_bypass(handle);
231         return (mp);
232     }
233
234     if (mp == NULL)
235         return (mp);
236
237     if (httpf_process_input(httpf, mp) == 1) {
238         sof_newconn_ready(handle);
239         sof_bypass(handle);
240     }
241
242     return (mp);
243 }
244
245 static sof_ops_t httpf_ops = {
246     .sofop_attach_passive = httpf_attach_passive_cb,
247     .sofop_detach = httpf_detach_cb,
248     .sofop_data_in = httpf_data_in_cb,
249 };
250
251 int
252 _init(void)
253 {
254     int err;
255
256     err = sof_register(SOF_VERSION, HTTPFILT_MODULE, &httpf_ops, 0);
257     if (err != 0)
258         return (err);
259     if ((err = mod_install(&httpf_modlinkage)) != 0)

```

```
260         (void) sof_unregister(HTTPFILT_MODULE);
262         return (err);
263     }
265     int
266     _fini(void)
267     {
268         int error;
270         if ((error = sof_unregister(HTTPFILT_MODULE)) != 0)
271             return (error);
273         return (mod_remove(&httpf_modlinkage));
274     }
276     int
277     _info(struct modinfo *modinfop)
278     {
279         return (mod_info(&httpf_modlinkage, modinfop));
280     }
```

```

*****
16076 Sat Nov 22 13:16:59 2014
new/usr/src/uts/intel/Makefile.intel
1403 Support deferring TCP accept()s
*****
1 # CDDL HEADER START
2 #
3 # The contents of this file are subject to the terms of the
4 # Common Development and Distribution License (the "License").
5 # You may not use this file except in compliance with the License.
6 #
7 # You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
8 # or http://www.opensolaris.org/os/licensing.
9 # See the License for the specific language governing permissions
10 # and limitations under the License.
11 #
12 # When distributing Covered Code, include this CDDL HEADER in each
13 # file and include the License file at usr/src/OPENSOLARIS.LICENSE.
14 # If applicable, add the following below this CDDL HEADER, with the
15 # fields enclosed by brackets "[]" replaced with your own identifying
16 # information: Portions Copyright [yyyy] [name of copyright owner]
17 #
18 # CDDL HEADER END
19 #
21 # Copyright (c) 2005, 2010, Oracle and/or its affiliates. All rights reserved.
22 # Copyright (c) 2014 Nexenta Systems, Inc. All rights reserved.
23 # Copyright (c) 2013 Andrew Stormont. All rights reserved.
25 #
26 # This makefile contains the common definitions for all intel
27 # implementation architecture independent modules.
28 #
30 #
31 # Machine type (implementation architecture):
32 #
33 PLATFORM = i86pc
35 #
36 # Everybody needs to know how to build modstubs.o and to locate unix.o.
37 # Note that unix.o must currently be selected from among the possible
38 # "implementation architectures". Note further, that unix.o is only
39 # used as an optional error check for undefines so (theoretically)
40 # any "implementation architectures" could be used. We choose i86pc
41 # because it is the reference port.
42 #
43 UNIX_DIR = $(UTSBASE)/i86pc/unix
44 GENLIB_DIR = $(UTSBASE)/intel/genunix
45 IPDRV_DIR = $(UTSBASE)/intel/ip
46 MODSTUBS_DIR = $(UNIX_DIR)
47 DSF_DIR = $(UTSBASE)/$(PLATFORM)/genassym
48 LINTS_DIR = $(OBJS_DIR)
49 LINT_LIB_DIR = $(UTSBASE)/intel/lint-libs/$(OBJS_DIR)
51 UNIX_O = $(UNIX_DIR)/$(OBJS_DIR)/unix.o
52 GENLIB = $(GENLIB_DIR)/$(OBJS_DIR)/libgenunix.so
53 MODSTUBS_O = $(MODSTUBS_DIR)/$(OBJS_DIR)/modstubs.o
54 LINT_LIB = $(UTSBASE)/i86pc/lint-libs/$(OBJS_DIR)/llib-lunix.ln
55 GEN_LINT_LIB = $(UTSBASE)/intel/lint-libs/$(OBJS_DIR)/llib-lgenunix.ln
57 #
58 # Include the makefiles which define build rule templates, the
59 # collection of files per module, and a few specific flags. Note
60 # that order is significant, just as with an include path. The
61 # first build rule template which matches the files name will be

```

```

62 # used. By including these in order from most machine dependent
63 # to most machine independent, we allow a machine dependent file
64 # to be used in preference over a machine independent version
65 # (Such as a machine specific optimization, which preserves the
66 # interfaces.)
67 #
68 include $(UTSBASE)/intel/Makefile.files
69 include $(UTSBASE)/common/Makefile.files
71 #
72 # ----- TRANSITIONAL SECTION -----
73 #
75 #
76 # Not everything which *should* be a module is a module yet. The
77 # following is a list of such objects which are currently part of
78 # genunix but which might someday become kmods. This must be
79 # defined before we include Makefile.uts, or else genunix's build
80 # won't be as parallel as we might like.
81 #
82 NOT_YET_KMODS = $(OLDPTY_OBJS) $(PTY_OBJS) $(VCONS_CONF_OBJS) $(MOD_OBJS)
84 #
85 # ----- END OF TRANSITIONAL SECTION -----
86 #
87 # Include machine independent rules. Note that this does not imply
88 # that the resulting module from rules in Makefile.uts is machine
89 # independent. Only that the build rules are machine independent.
90 #
91 include $(UTSBASE)/Makefile.uts
93 #
94 # The following must be defined for all implementations:
95 #
96 MODSTUBS = $(UTSBASE)/intel/ia32/ml/modstubs.s
98 #
99 # Define supported builds
100 #
101 DEF_BUILDS = $(DEF_BUILDS64) $(DEF_BUILDS32)
102 ALL_BUILDS = $(ALL_BUILDS64) $(ALL_BUILDS32)
104 #
105 # x86 or amd64 inline templates
106 #
107 INLINES_32 = $(UTSBASE)/intel/ia32/ml/ia32.il
108 INLINES_64 = $(UTSBASE)/intel/amd64/ml/amd64.il
109 INLINES += $(INLINES_$(CLASS))
111 #
112 # kernel-specific optimizations; override default in Makefile.master
113 #
115 CFLAGS_XARCH_32 = $(i386_CFLAGS)
116 CFLAGS_XARCH_64 = $(amd64_CFLAGS)
117 CFLAGS_XARCH = $(CFLAGS_XARCH_$(CLASS))
119 COPTFLAG_32 = $(COPTFLAG)
120 COPTFLAG_64 = $(COPTFLAG64)
121 COPTIMIZE = $(COPTFLAG_$(CLASS))
123 CFLAGS = $(CFLAGS_XARCH)
124 CFLAGS += $(COPTIMIZE)
125 CFLAGS += $(INLINES) -D_ASM_INLINES
126 CFLAGS += $(CCMODE)
127 CFLAGS += $(SPACEFLAG)

```

## new/usr/src/uts/intel/Makefile.intel

3

```

128 CFLAGS          += $(CCUNBOUND)
129 CFLAGS          += $(CFLAGS_uts)
130 CFLAGS          += -xstrconst

132 ASFLAGS_XARCH_32 = $(i386_ASFLAGS)
133 ASFLAGS_XARCH_64 = $(amd64_ASFLAGS)
134 ASFLAGS_XARCH    = $(ASFLAGS_XARCH_$(CLASS))

136 ASFLAGS        += $(ASFLAGS_XARCH)

138 #
139 #       Define the base directory for installation.
140 #
141 BASE_INS_DIR    = $(ROOT)

143 #
144 #       Debugging level
145 #
146 #       Special knowledge of which special debugging options affect which
147 #       file is used to optimize the build if these flags are changed.
148 #
149 DEBUG_DEFS_OBJ32 =
150 DEBUG_DEFS_DBG32 = -DDEBUG
151 DEBUG_DEFS_OBJ64 =
152 DEBUG_DEFS_DBG64 = -DDEBUG
153 DEBUG_DEFS       = $(DEBUG_DEFS_$(BUILD_TYPE))

155 DEBUG_COND_OBJ32 = $(POUND_SIGN)
156 DEBUG_COND_DBG32 =
157 DEBUG_COND_OBJ64 = $(POUND_SIGN)
158 DEBUG_COND_DBG64 =
159 IF_DEBUG_OBJ     = $(DEBUG_COND_$(BUILD_TYPE))$(OBJS_DIR)/

161 $(IF_DEBUG_OBJ)syscall.o      :=      DEBUG_DEFS      += -DSYSCALLTRACE
162 $(IF_DEBUG_OBJ)clock.o       :=      DEBUG_DEFS      += -DKSLICE=1

164 #
165 #       Collect the preprocessor definitions to be associated with *all*
166 #       files.
167 #
168 ALL_DEFS          = $(DEBUG_DEFS) $(OPTION_DEFS)

170 #
171 #       The kernels modules which are "implementation architecture"
172 #       specific for this machine are enumerated below. Note that most
173 #       of these modules must exist (in one form or another) for each
174 #       architecture.
175 #
176 #       Common Drivers (usually pseudo drivers) (/kernel/drv)
177 #       DRV_KMODS are built both 32-bit and 64-bit
178 #       DRV_KMODS_32 are built only 32-bit
179 #       DRV_KMODS_64 are built only 64-bit
180 #
181 DRV_KMODS        += aac
182 DRV_KMODS        += aggr
183 DRV_KMODS        += ahci
184 DRV_KMODS        += amd64_gart
185 DRV_KMODS        += amr
186 DRV_KMODS        += agpgart
187 DRV_KMODS        += srn
188 DRV_KMODS        += agptarget
189 DRV_KMODS        += arn
190 DRV_KMODS        += arp
191 DRV_KMODS        += asy
192 DRV_KMODS        += ata
193 DRV_KMODS        += ath

```

## new/usr/src/uts/intel/Makefile.intel

4

```

194 DRV_KMODS        += atu
195 DRV_KMODS        += audio
196 DRV_KMODS        += audio1575
197 DRV_KMODS        += audio810
198 DRV_KMODS        += audiocmi
199 DRV_KMODS        += audiocmihd
200 DRV_KMODS        += audioemul0k
201 DRV_KMODS        += audioens
202 DRV_KMODS        += audiohd
203 DRV_KMODS        += audioixp
204 DRV_KMODS        += audiols
205 DRV_KMODS        += audiopl6x
206 DRV_KMODS        += audiopci
207 DRV_KMODS        += audiosolo
208 DRV_KMODS        += audiotst
209 DRV_KMODS        += audiovia823x
210 DRV_KMODS_32    += audiovia97
211 DRV_KMODS        += bl
212 DRV_KMODS        += blkdev
213 DRV_KMODS        += bge
214 DRV_KMODS        += bofi
215 DRV_KMODS        += bpf
216 DRV_KMODS        += bridge
217 DRV_KMODS        += bscbus
218 DRV_KMODS        += bscv
219 DRV_KMODS        += chxge
220 DRV_KMODS        += cxgbe
221 DRV_KMODS        += ntxn
222 DRV_KMODS        += myril0ge
223 DRV_KMODS        += clone
224 DRV_KMODS        += cmdk
225 DRV_KMODS        += cn
226 DRV_KMODS        += conskbd
227 DRV_KMODS        += consms
228 DRV_KMODS        += cpqary3
229 DRV_KMODS        += cpuid
230 DRV_KMODS        += cpunex
231 DRV_KMODS        += crypto
232 DRV_KMODS        += cryptoadm
233 DRV_KMODS        += dca
234 DRV_KMODS        += devinfo
235 DRV_KMODS        += dld
236 DRV_KMODS        += dlpistub
237 DRV_KMODS_32    += dnet
238 DRV_KMODS        += dump
239 DRV_KMODS        += ecpp
240 DRV_KMODS        += emlxs
241 DRV_KMODS        += fd
242 DRV_KMODS        += fdc
243 DRV_KMODS        += fm
244 DRV_KMODS        += fssnap
245 DRV_KMODS        += hxge
246 DRV_KMODS        += i8042
247 DRV_KMODS        += i915
248 DRV_KMODS        += icmp
249 DRV_KMODS        += icmp6
250 DRV_KMODS        += intel_nb5000
251 DRV_KMODS        += intel_nhm
252 DRV_KMODS        += ip
253 DRV_KMODS        += ip6
254 DRV_KMODS        += ipd
255 DRV_KMODS        += ipf
256 DRV_KMODS        += ipnet
257 DRV_KMODS        += ippctl
258 DRV_KMODS        += ipsecah
259 DRV_KMODS        += ipsecesp

```

```

260 DRV_KMODS += ipw
261 DRV_KMODS += iwh
262 DRV_KMODS += iwi
263 DRV_KMODS += iwk
264 DRV_KMODS += iwp
265 DRV_KMODS += iwscn
266 DRV_KMODS += kb8042
267 DRV_KMODS += keysock
268 DRV_KMODS += kssl
269 DRV_KMODS += kstat
270 DRV_KMODS += ksyms
271 DRV_KMODS += kmdb
272 DRV_KMODS += llcl
273 DRV_KMODS += lofi
274 DRV_KMODS += log
275 DRV_KMODS += logindmux
276 DRV_KMODS += mega_sas
277 DRV_KMODS += mc-amd
278 DRV_KMODS += mm
279 DRV_KMODS += mouse8042
280 DRV_KMODS += mpt_sas
281 DRV_KMODS += mr_sas
282 DRV_KMODS += mwl
283 DRV_KMODS += nca
284 DRV_KMODS += nsmb
285 DRV_KMODS += nulldriver
286 DRV_KMODS += nv_sata
287 DRV_KMODS += nxge
288 DRV_KMODS += oce
289 DRV_KMODS += openeep
290 DRV_KMODS += pci_pci
291 DRV_KMODS += pcic
292 DRV_KMODS += pcieb
293 DRV_KMODS += phymem
294 DRV_KMODS += pit_beeper
295 DRV_KMODS += pm
296 DRV_KMODS += poll
297 DRV_KMODS += pool
298 DRV_KMODS += power
299 DRV_KMODS += pseudo
300 DRV_KMODS += ptc
301 DRV_KMODS += ptm
302 DRV_KMODS += pts
303 DRV_KMODS += ptsl
304 DRV_KMODS += qlge
305 DRV_KMODS += radeon
306 DRV_KMODS += ral
307 DRV_KMODS += ramdisk
308 DRV_KMODS += random
309 DRV_KMODS += rds
310 DRV_KMODS += rdsv3
311 DRV_KMODS += rpcib
312 DRV_KMODS += rsm
313 DRV_KMODS += rts
314 DRV_KMODS += rtw
315 DRV_KMODS += rum
316 DRV_KMODS += rwd
317 DRV_KMODS += rwn
318 DRV_KMODS += sad
319 DRV_KMODS += sd
320 DRV_KMODS += sdhost
321 DRV_KMODS += sgen
322 DRV_KMODS += si3124
323 DRV_KMODS += smbios
324 DRV_KMODS += skd
325 DRV_KMODS += softmac

```

```

326 DRV_KMODS += spdsock
327 DRV_KMODS += sbsrv
328 DRV_KMODS += smp
329 DRV_KMODS += spps
330 DRV_KMODS += sppptun
331 DRV_KMODS += srpt
332 DRV_KMODS += st
333 DRV_KMODS += sy
334 DRV_KMODS += sysevent
335 DRV_KMODS += sysmsg
336 DRV_KMODS += tcp
337 DRV_KMODS += tcp6
338 DRV_KMODS += tl
339 DRV_KMODS += tnf
340 DRV_KMODS += tpm
341 DRV_KMODS += trill
342 DRV_KMODS += udp
343 DRV_KMODS += udp6
344 DRV_KMODS += ucode
345 DRV_KMODS += ural
346 DRV_KMODS += uath
347 DRV_KMODS += urtw
348 DRV_KMODS += vgatext
349 DRV_KMODS += heci
350 DRV_KMODS += vnic
351 DRV_KMODS += vscan
352 DRV_KMODS += wc
353 DRV_KMODS += winlock
354 DRV_KMODS += wpi
355 DRV_KMODS += xge
356 DRV_KMODS += yge
357 DRV_KMODS += zcons
358 DRV_KMODS += zyd
359 DRV_KMODS += simnet
360 DRV_KMODS += stmf
361 DRV_KMODS += stmf_sbd
362 DRV_KMODS += fct
363 DRV_KMODS += fcoe
364 DRV_KMODS += fcoet
365 DRV_KMODS += fcoei
366 DRV_KMODS += qlt
367 DRV_KMODS += iscsit
368 DRV_KMODS += pppt
369 DRV_KMODS += ncall nsctl sdbc nskern sv
370 DRV_KMODS += ii rdc rdcsrv rdcstub
371 DRV_KMODS += iptun

373 #
374 # Common code drivers
375 #

377 DRV_KMODS += afe
378 DRV_KMODS += atge
379 DRV_KMODS += bfe
380 DRV_KMODS += dmfe
381 DRV_KMODS += el000g
382 DRV_KMODS += efe
383 DRV_KMODS += elxl
384 DRV_KMODS += hme
385 DRV_KMODS += mxfe
386 DRV_KMODS += nge
387 DRV_KMODS += pcn
388 DRV_KMODS += rge
389 DRV_KMODS += rtls
390 DRV_KMODS += sfe
391 DRV_KMODS += amd8111s

```



```

392 DRV_KMODS      += igb
393 DRV_KMODS      += ipmi
394 DRV_KMODS      += iprb
395 DRV_KMODS      += ixgbe
396 DRV_KMODS      += vr

398 #
399 # Virtio drivers
400 #

402 # Virtio core
403 DRV_KMODS      += virtio

405 # Virtio block driver
406 DRV_KMODS      += vioblk

408 #
409 #      DTrace and DTrace Providers
410 #
411 DRV_KMODS      += dtrace
412 DRV_KMODS      += fbt
413 DRV_KMODS      += lockstat
414 DRV_KMODS      += profile
415 DRV_KMODS      += sdt
416 DRV_KMODS      += systrace
417 DRV_KMODS      += fasttrap
418 DRV_KMODS      += dcpc

420 #
421 #      I/O framework test drivers
422 #
423 DRV_KMODS      += pshot
424 DRV_KMODS      += gen_drv
425 DRV_KMODS      += tvhci tphci tclient
426 DRV_KMODS      += emul64

428 #
429 #      Machine Specific Driver Modules (/kernel/drv):
430 #
431 DRV_KMODS      += options
432 DRV_KMODS      += scsi_vhci
433 DRV_KMODS      += pmcs
434 DRV_KMODS      += pmcs8001fw
435 DRV_KMODS      += arcmsr
436 DRV_KMODS      += fcp
437 DRV_KMODS      += fcip
438 DRV_KMODS      += fcsm
439 DRV_KMODS      += fp
440 DRV_KMODS      += qlc
441 DRV_KMODS      += iscsi

443 #
444 #      PCMCIA specific module(s)
445 #
446 DRV_KMODS      += pcs
447 MISC_KMODS     += cardbus

449 #
450 #      SCSI Enclosure Services driver
451 #
452 DRV_KMODS      += ses

454 #
455 #      USB specific modules
456 #
457 DRV_KMODS      += hid

```

```

458 DRV_KMODS      += hwarc hwhac
459 DRV_KMODS      += hubd
460 DRV_KMODS      += uhci
461 DRV_KMODS      += ehci
462 DRV_KMODS      += ohci
463 DRV_KMODS      += usb_mid
464 DRV_KMODS      += usb_ia
465 DRV_KMODS      += scsa2usb
466 DRV_KMODS      += usbprn
467 DRV_KMODS      += ugen
468 DRV_KMODS      += usbser
469 DRV_KMODS      += usbsacm
470 DRV_KMODS      += usbsksp
471 DRV_KMODS      += usbsprl
472 DRV_KMODS      += usb_ac
473 DRV_KMODS      += usb_as
474 DRV_KMODS      += usbskel
475 DRV_KMODS      += usbvc
476 DRV_KMODS      += usbftdi
477 DRV_KMODS      += wusb_df
478 DRV_KMODS      += wusb_ca
479 DRV_KMODS      += usbecm

481 #
482 #      1394 modules
483 #
484 MISC_KMODS     += s1394 sbp2
485 DRV_KMODS      += hci1394 scsa1394
486 DRV_KMODS      += av1394
487 DRV_KMODS      += dcaml394

489 #
490 #      InfiniBand pseudo drivers
491 #
492 DRV_KMODS      += ib ibp eibnx eoib rdsib sdp iser daplt hermon tavor sol_ucma
493 DRV_KMODS      += sol_umad

495 #
496 #      LVM modules
497 #
498 DRV_KMODS      += md
499 MISC_KMODS     += md_stripe md_hotspares md_mirror md_raid md_trans md_notify
500 MISC_KMODS     += md_sp

502 #
503 #      Brand modules
504 #
505 BRAND_KMODS    += snl_brand s10_brand

507 #
508 #      Exec Class Modules (/kernel/exec):
509 #
510 EXEC_KMODS     += elfexec intpexec shbinexec javaexec

512 #
513 #      Scheduling Class Modules (/kernel/sched):
514 #
515 SCHED_KMODS    += IA RT TS RT_DPTBL TS_DPTBL FSS FX FX_DPTBL SDC

517 #
518 #      File System Modules (/kernel/fs):
519 #
520 FS_KMODS       += autofs cachefs ctfs dcfs dev devfs fdfs fifofs hsf s lofs
521 FS_KMODS       += mntfs namefs nfs objfs zfs zut
522 FS_KMODS       += pcfs procfs sockfs specfs tmpfs udfs ufs sharefs
523 FS_KMODS       += smbfs

```

```

525 #
526 #     Streams Modules (/kernel/strmod):
527 #
528 STRMOD_KMODS += bufmod connld dedump ldterm pckt pfmod pipemod
529 STRMOD_KMODS += ptem redirmod rpcmod rlmod telmod timod
530 STRMOD_KMODS += spppsyn spppcomp
531 STRMOD_KMODS += tirdwr ttcompat
532 STRMOD_KMODS += usbkbm
533 STRMOD_KMODS += usbms
534 STRMOD_KMODS += usbwcm
535 STRMOD_KMODS += usb_ah
536 STRMOD_KMODS += drcompat
537 STRMOD_KMODS += cryptmod
538 STRMOD_KMODS += vuid2ps2
539 STRMOD_KMODS += vuid3ps2
540 STRMOD_KMODS += vuidm3p
541 STRMOD_KMODS += vuidm4p
542 STRMOD_KMODS += vuidm5p

544 #
545 #     'System' Modules (/kernel/sys):
546 #
547 SYS_KMODS += c2audit
548 SYS_KMODS += doorfs
549 SYS_KMODS += exacctsys
550 SYS_KMODS += inst_sync
551 SYS_KMODS += kaio
552 SYS_KMODS += msgsys
553 SYS_KMODS += pipe
554 SYS_KMODS += portfs
555 SYS_KMODS += pset
556 SYS_KMODS += semsys
557 SYS_KMODS += shmsys
558 SYS_KMODS += sysacct
559 SYS_KMODS += acctctl

561 #
562 #     'Misc' Modules (/kernel/misc)
563 #     MISC_KMODS are built both 32-bit and 64-bit
564 #     MISC_KMODS_32 are built only 32-bit
565 #     MISC_KMODS_64 are built only 64-bit
566 #
567 MISC_KMODS += ac97
568 MISC_KMODS += acpica
569 MISC_KMODS += agpmaster
570 MISC_KMODS += bignum
571 MISC_KMODS += bootdev
572 MISC_KMODS += busra
573 MISC_KMODS += cmlb
574 MISC_KMODS += consconfig
575 MISC_KMODS += ctf
576 MISC_KMODS += dadk
577 MISC_KMODS += dcopy
578 MISC_KMODS += dls
579 MISC_KMODS += drm
580 MISC_KMODS += fssnap_if
581 MISC_KMODS += gda
582 MISC_KMODS += gld
583 MISC_KMODS += hidparser
584 MISC_KMODS += hook
585 MISC_KMODS += hpcsvc
586 MISC_KMODS += ibcm
587 MISC_KMODS += ibdm
588 MISC_KMODS += ibdma
589 MISC_KMODS += ibmf

```

```

590 MISC_KMODS += ibtl
591 MISC_KMODS += idm
592 MISC_KMODS += idmap
593 MISC_KMODS += iomulib
594 MISC_KMODS += ipc
595 MISC_KMODS += kbtrans
596 MISC_KMODS += kcf
597 MISC_KMODS += kgssapi
598 MISC_KMODS += kmecch_dummy
599 MISC_KMODS += kmecch_krb5
600 MISC_KMODS += ksocket
601 MISC_KMODS += mac
602 MISC_KMODS += mii
603 MISC_KMODS += mwlfw
604 MISC_KMODS += net80211
605 MISC_KMODS += nfs_dlboot
606 MISC_KMODS += nfssrv
607 MISC_KMODS += neti
608 MISC_KMODS += pci_autoconfig
609 MISC_KMODS += pcicfg
610 MISC_KMODS += pcihp
611 MISC_KMODS += pcmcia
612 MISC_KMODS += rpcsec
613 MISC_KMODS += rpcsec_gss
614 MISC_KMODS += rsmops
615 MISC_KMODS += sata
616 MISC_KMODS += scsi
617 MISC_KMODS += sda
618 MISC_KMODS += sol_ofs
619 MISC_KMODS += spuni
620 MISC_KMODS += strategy
621 MISC_KMODS += strplumb
622 MISC_KMODS += tem
623 MISC_KMODS += tlimod
624 MISC_KMODS += usba usb10 usbs49_fw
625 MISC_KMODS += scsi_vhci_f_sym_hds
626 MISC_KMODS += scsi_vhci_f_sym
627 MISC_KMODS += scsi_vhci_f_tpgs
628 MISC_KMODS += scsi_vhci_f_asym_sun
629 MISC_KMODS += scsi_vhci_f_tape
630 MISC_KMODS += scsi_vhci_f_tpgs_tape
631 MISC_KMODS += fctl
632 MISC_KMODS += emlxs_fw
633 MISC_KMODS += qlc_fw_2200
634 MISC_KMODS += qlc_fw_2300
635 MISC_KMODS += qlc_fw_2400
636 MISC_KMODS += qlc_fw_2500
637 MISC_KMODS += qlc_fw_6322
638 MISC_KMODS += qlc_fw_8100
639 MISC_KMODS += hwal480_fw
640 MISC_KMODS += uathfw
641 MISC_KMODS += uwba

643 MISC_KMODS += klmmod klmops

645 #
646 #     Software Cryptographic Providers (/kernel/crypto):
647 #
648 CRYPTO_KMODS += aes
649 CRYPTO_KMODS += arcfour
650 CRYPTO_KMODS += blowfish
651 CRYPTO_KMODS += des
652 CRYPTO_KMODS += ecc
653 CRYPTO_KMODS += md4
654 CRYPTO_KMODS += md5
655 CRYPTO_KMODS += rsa

```

```
656 CRYPTO_KMODS += sha1
657 CRYPTO_KMODS += sha2
658 CRYPTO_KMODS += swrand

660 #
661 #     IP Policy Modules (/kernel/ipp)
662 #
663 IPP_KMODS      += dlcosmk
664 IPP_KMODS      += flowacct
665 IPP_KMODS      += ipgpc
666 IPP_KMODS      += dscpmk
667 IPP_KMODS      += tokenmt
668 IPP_KMODS      += tswtclmt

670 #
671 #     generic-unix module (/kernel/genunix):
672 #
673 GENUNIX_KMODS  += genunix

675 #
676 #     Modules eXcluded from the product:
677 #

679 #
680 #     'Dacf' Modules (/kernel/dacf):
681 #

683 #
684 #     Performance Counter BackEnd modules (/usr/kernel/pcbe)
685 #
686 PCBE_KMODS     += p123_pcbe p4_pcbe opteron_pcbe core_pcbe

688 #
689 #     MAC-Type Plugin Modules (/kernel/mac)
690 #
691 MAC_KMODS      += mac_6to4
692 MAC_KMODS      += mac_ether
693 MAC_KMODS      += mac_ipv4
694 MAC_KMODS      += mac_ipv6
695 MAC_KMODS      += mac_wifi
696 MAC_KMODS      += mac_ib

698 #
699 #     socketmod (kernel/socketmod)
700 #
701 SOCKET_KMODS   += sockpfp
702 SOCKET_KMODS   += socksctp
703 SOCKET_KMODS   += socksdp
704 SOCKET_KMODS   += sockrds
705 SOCKET_KMODS   += ksslf
706 SOCKET_KMODS   += httpfilt
707 SOCKET_KMODS   += datafilt

709 #
710 #     kiconv modules (/kernel/kiconv):
711 #
712 KICONV_KMODS   += kiconv_emea kiconv_ja kiconv_ko kiconv_sc kiconv_tc

714 #
715 #     'Dacf' Modules (/kernel/dacf):
716 #
717 DACF_KMODS     += net_dacf

719 #
720 # Ensure that the variable member of the cpu_t (cpu_m) is defined
721 # for the lint builds so as not to cause lint errors during the
```

```
722 # global cross check.
723 #
724 LINTFLAGS      += -D_MACHDEP -I$(UTSBASE)/i86pc
```

\*\*\*\*\*

1436 Sat Nov 22 13:16:59 2014

new/usr/src/uts/intel/datafilt/Makefile

1403 Support deferring TCP accept()s

\*\*\*\*\*

```

1 #
2 # This file and its contents are supplied under the terms of the
3 # Common Development and Distribution License ("CDDL"), version 1.0.
4 # You may only use this file in accordance with the terms of version
5 # 1.0 of the CDDL.
6 #
7 # A full copy of the text of the CDDL should have accompanied this
8 # source. A copy of the CDDL is also available via the Internet at
9 # http://www.illumos.org/license/CDDL.
10 #

12 #
13 # Copyright (c) 2011, OmniTI Computer Consulting, Inc. All rights reserved.
14 # Copyright 2012, Nexenta Systems, Inc. All rights reserved.
15 #

```

```

18 #
19 # Path to the base of the uts directory tree (usually /usr/src/uts).
20 #
21 UTSBASE = ../..

```

```

23 #
24 # Define the module and object file sets.
25 #
26 MODULE      = datafilt
27 OBJECTS     = $(DATAFILT_OBJS:%=$(OBJS_DIR)/%)
28 LINTS       = $(DATAFILT_OBJS:%.o=$(LINTS_DIR)/%.ln)
29 ROOTMODULE  = $(ROOT SOCK_DIR)/$(MODULE)

```

```

31 #
32 # Include common rules.
33 #
34 include $(UTSBASE)/intel/Makefile.intel

```

```

36 #
37 # Define targets
38 #
39 ALL_TARGET  = $(BINARY)
40 LINT_TARGET = $(MODULE).lint
41 INSTALL_TARGET = $(BINARY) $(ROOTMODULE)

```

```

43 #
44 # lint pass one enforcement and OS version
45 #
46 CFLAGS += $(CCVERBOSE)

```

```

48 LDFLAGS += -dy -Nfs/sockfs -Ndrv/ip

```

```

50 #
51 # Default build targets.
52 #
53 .KEEP_STATE:

```

```

55 def:          $(DEF_DEPS)

57 all:         $(ALL_DEPS)

59 clean:       $(CLEAN_DEPS)

61 clobber:     $(CLOBBER_DEPS)

```

```

63 lint:        $(LINT_DEPS)

65 modlintlib:  $(MODLINTLIB_DEPS)

67 clean.lint:  $(CLEAN_LINT_DEPS)

69 install:     $(INSTALL_DEPS)

71 #
72 # Include common targets.
73 #
74 include $(UTSBASE)/intel/Makefile.targ

```

```

*****
1435 Sat Nov 22 13:16:59 2014
new/usr/src/uts/intel/httpfilt/Makefile
1403 Support deferring TCP accept()s
*****
1 #
2 # This file and its contents are supplied under the terms of the
3 # Common Development and Distribution License ("CDDL"), version 1.0.
4 # You may only use this file in accordance with the terms of version
5 # 1.0 of the CDDL.
6 #
7 # A full copy of the text of the CDDL should have accompanied this
8 # source. A copy of the CDDL is also available via the Internet at
9 # http://www.illumos.org/license/CDDL.
10 #
11 #
12 #
13 # Copyright (c) 2011, OmniTI Computer Consulting, Inc. All rights reserved.
14 # Copyright 2012, Nexenta Systems, Inc. All rights reserved.
15 #
16 #
17 #
18 # Path to the base of the uts directory tree (usually /usr/src/uts).
19 #
20 UTSBASE = ../..
21 #
22 #
23 # Define the module and object file sets.
24 #
25 MODULE = httpfilt
26 OBJECTS = $(HTTPFILT_OBJS:%=$(OBJS_DIR)/%)
27 LINTS = $(HTTPFILT_OBJS:%.o=$(LINTS_DIR)/%.ln)
28 ROOTMODULE = $(ROOT SOCK_DIR)/$(MODULE)
29 #
30 #
31 # Include common rules.
32 #
33 include $(UTSBASE)/intel/Makefile.intel
34 #
35 #
36 # Define targets
37 #
38 ALL_TARGET = $(BINARY)
39 LINT_TARGET = $(MODULE).lint
40 INSTALL_TARGET = $(BINARY) $(ROOTMODULE)
41 #
42 #
43 # lint pass one enforcement and OS version
44 #
45 CFLAGS += $(CCVERBOSE)
46 #
47 LDFLAGS += -dy -Nfs/sockfs -Ndrv/ip
48 #
49 #
50 # Default build targets.
51 #
52 .KEEP_STATE:
53 #
54 def: $(DEF_DEPS)
55 #
56 all: $(ALL_DEPS)
57 #
58 clean: $(CLEAN_DEPS)
59 #
60 clobber: $(CLOBBER_DEPS)

```

```

62 lint: $(LINT_DEPS)
63 #
64 modlintlib: $(MODLINTLIB_DEPS)
65 #
66 clean.lint: $(CLEAN_LINT_DEPS)
67 #
68 install: $(INSTALL_DEPS)
69 #
70 #
71 # Include common targets.
72 #
73 include $(UTSBASE)/intel/Makefile.targ

```

```

*****
13450 Sat Nov 22 13:16:59 2014
new/usr/src/uts/sparc/Makefile.sparc
1403 Support deferring TCP accept()s
*****
1 #
2 # CDDL HEADER START
3 #
4 # The contents of this file are subject to the terms of the
5 # Common Development and Distribution License (the "License").
6 # You may not use this file except in compliance with the License.
7 #
8 # You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9 # or http://www.opensolaris.org/os/licensing.
10 # See the License for the specific language governing permissions
11 # and limitations under the License.
12 #
13 # When distributing Covered Code, include this CDDL HEADER in each
14 # file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 # If applicable, add the following below this CDDL HEADER, with the
16 # fields enclosed by brackets "[]" replaced with your own identifying
17 # information: Portions Copyright [yyyy] [name of copyright owner]
18 #
19 # CDDL HEADER END
20 #

22 # Copyright (c) 2005, 2010, Oracle and/or its affiliates. All rights reserved.
23 # Copyright 2012, Nexenta Systems, Inc. All rights reserved.
24 # Copyright (c) 2013 Andrew Stormont. All rights reserved.

27 #
28 # This makefile contains the common definitions for all sparc
29 # implementation architecture independent modules.
30 #

32 #
33 # Define supported builds
34 #
35 DEF_BUILDS = $(DEF_BUILDS64)
36 ALL_BUILDS = $(ALL_BUILDS64)

38 #
39 # Everybody needs to know how to build modstubs.o and to locate unix.o.
40 # Note that unix.o must currently be selected from among the possible
41 # "implementation architectures". Note further, that unix.o is only
42 # used as an optional error check for undefines so (theoretically)
43 # any "implementation architectures" could be used. We choose sun4u
44 # because it is the reference port.
45 #
46 UNIX_DIR = $(UTSBASE)/sun4u/unix
47 GENLIB_DIR = $(UTSBASE)/sun4u/genunix
48 IPDRV_DIR = $(UTSBASE)/sparc/ip
49 MODSTUBS_DIR = $(UNIX_DIR)
50 DSF_DIR = $(UNIX_DIR)
51 LINTS_DIR = $(OBJS_DIR)
52 LINT_LIB_DIR = $(UTSBASE)/sparc/lint-libs/$(OBJS_DIR)

54 UNIX_O = $(UNIX_DIR)/$(OBJS_DIR)/unix.o
55 MODSTUBS_O = $(MODSTUBS_DIR)/$(OBJS_DIR)/modstubs.o
56 GENLIB = $(UTSBASE)/sun4u/lint-libs/$(OBJS_DIR)/libgenunix.so

58 LINT_LIB_32 = $(UTSBASE)/sun4u/lint-libs/$(OBJS_DIR)/llib-lunix.ln
59 GEN_LINT_LIB_32 = $(UTSBASE)/sun4u/lint-libs/$(OBJS_DIR)/llib-lgenunix.ln

61 LINT_LIB_64 = $(UTSBASE)/sun4u/lint-libs/$(OBJS_DIR)/llib-lunix.ln

```

```

62 GEN_LINT_LIB_64 = $(UTSBASE)/sun4u/lint-libs/$(OBJS_DIR)/llib-lgenunix.ln
64 LINT_LIB = $(LINT_LIB_$(CLASS))
65 GEN_LINT_LIB = $(GEN_LINT_LIB_$(CLASS))

67 LINT32_DIRS = $(LINT32_BUILDS:=$(UTSBASE)/sparc/lint-libs/%)
68 LINT32_FILES = $(LINT32_DIRS:=%/llib-l$(MODULE).ln)

70 LINT64_DIRS = $(LINT64_BUILDS:=$(UTSBASE)/sparc/lint-libs/%)
71 LINT64_FILES = $(LINT64_DIRS:=%/llib-l$(MODULE).ln)

73 #
74 # Include the makefiles which define build rule templates, the
75 # collection of files per module, and a few specific flags. Note
76 # that order is significant, just as with an include path. The
77 # first build rule template which matches the files name will be
78 # used. By including these in order from most machine dependent
79 # to most machine independent, we allow a machine dependent file
80 # to be used in preference over a machine independent version
81 # (Such as a machine specific optimization, which preserves the
82 # interfaces.)
83 #
84 include $(UTSBASE)/sparc/Makefile.files
85 include $(UTSBASE)/sparc/v9/Makefile.files
86 include $(UTSBASE)/sun/Makefile.files
87 include $(UTSBASE)/common/Makefile.files

89 #
90 # ----- TRANSITIONAL SECTION -----
91 #

93 #
94 # Not everything which *should* be a module is a module yet. The
95 # following is a list of such objects which are currently part of
96 # genunix but which might someday become kmods. This must be
97 # defined before we include Makefile.uts, or else genunix's build
98 # won't be as parallel as we might like.
99 #
100 NOT_YET_KMODS = $(OLDPTY_OBJS) $(PTY_OBJS) $(VCONS_CONF_OBJS) $(MOD_OBJS)

102 #
103 # ----- END OF TRANSITIONAL SECTION -----
104 #
105 # Include machine independent rules. Note that this does not imply
106 # that the resulting module from rules in Makefile.uts is machine
107 # independent. Only that the build rules are machine independent.
108 #
109 include $(UTSBASE)/Makefile.uts

111 #
112 # machine specific optimization, override default in Makefile.master
113 #
114 XARCH_32 = -xarch=v8
115 XARCH_64 = -m64
116 XARCH = $(XARCH_$(CLASS))

118 COPTIMIZE_32 = -xO3
119 COPTIMIZE_64 = -xO3
120 COPTIMIZE = $(COPTIMIZE_$(CLASS))

122 CCMODE = -Xa

124 CFLAGS_32 = -xcg92
125 CFLAGS_64 = -xchip=ultra $(CCABS32) $(CCREGSYM)
126 CFLAGS = $(CFLAGS_$(CLASS))

```

```

128 CFLAGS      += $(XARCH)
129 CFLAGS      += $(COPTIMIZE)
130 CFLAGS      += $(EXTRA_CFLAGS)
131 CFLAGS      += $(XAOPT)
132 CFLAGS      += $(INLINES) -D_ASM_INLINES
133 CFLAGS      += $(CCMODE)
134 CFLAGS      += $(SPACEFLAG)
135 CFLAGS      += $(CERRWARN)
136 CFLAGS      += $(CTF_FLAGS_$(CLASS))
137 CFLAGS      += $(C99MODE)
138 CFLAGS      += $(CCUNBOUND)
139 CFLAGS      += $(CCSTATICSYM)
140 CFLAGS      += $(CC32BITCALLERS)
141 CFLAGS      += $(CCNOAUTOINLINE)
142 CFLAGS      += $(IROPTFLAG)
143 CFLAGS      += $(CGLOBALSTATIC)
144 CFLAGS      += -xregs=no%float
145 CFLAGS      += -xstrconst
146 CFLAGS      += $(CSOURCEDEBUGFLAGS)
147 CFLAGS      += $(CUSERFLAGS)

149 ASFLAGS     += $(XARCH)

151 LINT_DEFS_32 =
152 LINT_DEFS_64 = -m64
153 LINT_DEFS    += $(LINT_DEFS_$(CLASS))

155 #
156 #   The following must be defined for all implementations:
157 #
158 #   MODSTUBS:      Module stubs source file.
159 #
160 MODSTUBS     = $(UTSBASE)/sparc/ml/modstubs.s

162 #
163 #   Define the actual specific platforms - obviously none.
164 #
165 MACHINE_DEFS  =

167 #
168 #   Debugging level
169 #
170 #   Special knowledge of which special debugging options effect which
171 #   file is used to optimize the build if these flags are changed.
172 #
173 #   XXX: The above could possibly be done for more flags and files, but
174 #   is left as an experiment to the interested reader. Be forewarned,
175 #   that excessive use could lead to maintenance difficulties.
176 #
177 DEBUG_DEFS_OBJ32 =
178 DEBUG_DEFS_DBG32 = -DDEBUG
179 DEBUG_DEFS_OBJ64 =
180 DEBUG_DEFS_DBG64 = -DDEBUG
181 DEBUG_DEFS       = $(DEBUG_DEFS_$(BUILD_TYPE))

183 DEBUG_COND_OBJ32 = $(POUND_SIGN)
184 DEBUG_COND_DBG32 =
185 DEBUG_COND_OBJ64 = $(POUND_SIGN)
186 DEBUG_COND_DBG64 =
187 IF_DEBUG_OBJ     = $(DEBUG_COND_$(BUILD_TYPE))$(OBJS_DIR)/

189 $(IF_DEBUG_OBJ)syscall.o      :=      DEBUG_DEFS      += -DSYSCALLTRACE
190 $(IF_DEBUG_OBJ)clock.o       :=      DEBUG_DEFS      += -DKSLICE=1

192 # Comment these out if you don't want dispatcher lock statistics.

```

```

194 # $(IF_DEBUG_OBJ)disp_lock.o      := DEBUG_DEFS      += -DDISP_LOCK_STATS

196 #
197 #   Collect the preprocessor definitions to be associated with *all*
198 #   files.
199 #
200 ALL_DEFS      = $(MACHINE_DEFS) $(DEBUG_DEFS) $(OPTION_DEFS)
201 #
202 #
203 #   The kernels modules which are "implementation architecture"
204 #   specific for this machine are enumerated below. Note that most
205 #   of these modules must exist (in one form or another) for each
206 #   architecture.
207 #
208 #   Common Drivers (usually pseudo drivers) (/kernel/drv):
209 #
210 DRV_KMODS     += aggr arp audio bl blkdev bofi clone cn conskbd consms cpuid
211 DRV_KMODS     += crypto cryptoadm devinfo dump
212 DRV_KMODS     += dtrace fasttrap fbt lockstat profile sdt systrace dcpd
213 DRV_KMODS     += fssnap icmp icmp6 ip ip6 ipnet ipsecah
214 DRV_KMODS     += ipsecesp iptun iwscn keysock kmdb kstat ksyms llc1
215 DRV_KMODS     += lofi
216 DRV_KMODS     += log logindmux kssl mm nca phymem pm poll pool
217 DRV_KMODS     += pseudo ptc ptm pts ptsl ramdisk random rsm rts sad
218 DRV_KMODS     += simnet softmac sPPP sPPPtun sy sysevent sysmsg
219 DRV_KMODS     += spdsock
220 DRV_KMODS     += tcp tcp6 tl tnf ttymux udp udp6 wc winlock zcons
221 DRV_KMODS     += ippctl
222 DRV_KMODS     += dld
223 DRV_KMODS     += ipd
224 DRV_KMODS     += ipf
225 DRV_KMODS     += rpcib
226 DRV_KMODS     += dlpistub
227 DRV_KMODS     += vnuc
228 DRV_KMODS     += xge
229 DRV_KMODS     += rds
230 DRV_KMODS     += rdsv3
231 DRV_KMODS     += chxge
232 DRV_KMODS     += sbsrv
233 DRV_KMODS     += vscan
234 DRV_KMODS     += nsmb
235 DRV_KMODS     += fm
236 DRV_KMODS     += nulldriver
237 DRV_KMODS     += bridge trill
238 DRV_KMODS     += bpf
239 DRV_KMODS     += dca

241 #
242 #   Hardware Drivers in common space
243 #
244 #
245 DRV_KMODS     += afe
246 DRV_KMODS     += audio1575
247 DRV_KMODS     += audioens
248 DRV_KMODS     += audiols
249 DRV_KMODS     += audiopl6x
250 DRV_KMODS     += audiopci
251 DRV_KMODS     += audiot
252 DRV_KMODS     += e1000g
253 DRV_KMODS     += efe
254 DRV_KMODS     += hxge
255 DRV_KMODS     += mxfe
256 DRV_KMODS     += rge
257 DRV_KMODS     += rtls
258 DRV_KMODS     += sfe
259 DRV_KMODS     += aac

```

```

260 DRV_KMODS      += igb
261 DRV_KMODS      += ixgbe
262 DRV_KMODS      += vr
263 DRV_KMODS      += mr_sas
264 DRV_KMODS      += yge

266 #
267 #      Machine Specific Driver Modules (/kernel/drv):
268 #
269 DRV_KMODS      += audiocs
270 DRV_KMODS      += bge dmfe eri fas hme qfe
271 DRV_KMODS      += openeep options sd ses st
272 DRV_KMODS      += ssd
273 DRV_KMODS      += ecpp
274 DRV_KMODS      += hid hubd ehci ohci uhci usb_mid usb_ia scsa2usb usbprn ugen
275 DRV_KMODS      += usbser usbsacm usbsksp usbsprl
276 DRV_KMODS      += usb_as usb_ac
277 DRV_KMODS      += usbskel
278 DRV_KMODS      += usbvsc
279 DRV_KMODS      += usbftdi
280 DRV_KMODS      += wusb_df hwahc hwarc wusb_ca
281 DRV_KMODS      += usbecm
282 DRV_KMODS      += hcil394 avl394 scsa1394 dcaml394
283 DRV_KMODS      += sbp2
284 DRV_KMODS      += ib ibp eibnx eoib rdsib sdp iser daplt hermon tavor sol_ucma
285 DRV_KMODS      += sol_umad
286 DRV_KMODS      += pci_pci pcieb pcieb_bcm
287 DRV_KMODS      += i8042 kb8042 mouse8042
288 DRV_KMODS      += fcode
289 DRV_KMODS      += mpt_sas
290 DRV_KMODS      += socal
291 DRV_KMODS      += sgen
292 DRV_KMODS      += myril0ge
293 DRV_KMODS      += smp
294 DRV_KMODS      += dad
295 DRV_KMODS      += scsi_vhci
296 DRV_KMODS      += fcp
297 DRV_KMODS      += fcip
298 DRV_KMODS      += fcsn
299 DRV_KMODS      += fp
300 DRV_KMODS      += qlc
301 DRV_KMODS      += qlge
302 DRV_KMODS      += stmf
303 DRV_KMODS      += stmf_sbd
304 DRV_KMODS      += fct
305 DRV_KMODS      += fcoe
306 DRV_KMODS      += fcoet
307 DRV_KMODS      += fcoel
308 DRV_KMODS      += qlt
309 DRV_KMODS      += iscsit
310 DRV_KMODS      += pppt
311 DRV_KMODS      += ncall nsctl sdbc nskern sv
312 DRV_KMODS      += ii rdc rdcsrv rdcstub
313 DRV_KMODS      += iscsi
314 DRV_KMODS      += emlxs
315 DRV_KMODS      += oce
316 DRV_KMODS      += srpt
317 DRV_KMODS      += pmcs
318 DRV_KMODS      += pmcs8001fw

320 #
321 #      I/O framework test drivers
322 #
323 DRV_KMODS      += pshot
324 DRV_KMODS      += gen_drv
325 DRV_KMODS      += tvhci tphci tclient

```

```

326 DRV_KMODS      += emul64

328 #
329 #      PCMCIA specific module(s)
330 #
331 DRV_KMODS      += pcs
332 MISC_KMODS     += busra cardbus dada pcmcia
333 DRV_KMODS      += pcic

335 # Add lvm
336 #
337 DRV_KMODS      += md
338 MISC_KMODS     += md_mirror md_stripe md_hotspares md_raid md_trans md_notify
339 MISC_KMODS     += md_sp

341 #
342 #      Exec Class Modules (/kernel/exec):
343 #
344 EXEC_KMODS     += aoutexec elfexec intpexec shbinexec javaexec

346 #
347 #      Scheduling Class Modules (/kernel/sched):
348 #
349 SCHED_KMODS    += RT TS RT_DPTBL TS_DPTBL IA FSS FX FX_DPTBL SDC

351 #
352 #      File System Modules (/kernel/fs):
353 #
354 FS_KMODS       += dev devfs fdfs fifofs hsfds lofs namefs nfs pcfs tmpfs zfs
355 FS_KMODS       += zut specfs udfs ufs autofs cacheofs procfs sockfs mntfs
356 FS_KMODS       += ctfs objfs sharefs dcfs smbfs

358 #
359 #      Streams Modules (/kernel/strmod):
360 #
361 STRMOD_KMODS   += bufmod connld dedump ldterm ms pckt pfmod
362 STRMOD_KMODS   += pipemod ptem redirmod rpcmod rlmmod telmod timod
363 STRMOD_KMODS   += sppasyn spppcomp
364 STRMOD_KMODS   += tirdwr ttcompat
365 STRMOD_KMODS   += usbkcm usbms usbwcm usb_ah
366 STRMOD_KMODS   += drcompat
367 STRMOD_KMODS   += cryptmod
368 STRMOD_KMODS   += vuid3ps2

370 #
371 #      'System' Modules (/kernel/sys):
372 #
373 SYS_KMODS      += c2audit
374 SYS_KMODS      += exacctsyst
375 SYS_KMODS      += inst_sync kaio msgsys semsys shmsys sysacct pipe
376 SYS_KMODS      += doorfs pset acctctl portfs

378 #
379 #      'User' Modules (/kernel/misc):
380 #
381 MISC_KMODS     += ac97
382 MISC_KMODS     += bignum
383 MISC_KMODS     += consconfig gld ipc nfs_dlboot nfssrv scsi
384 MISC_KMODS     += strplumb swapgeneric tlimod
385 MISC_KMODS     += rpcsec rpcsec_gss kgssapi kmecch_dummy
386 MISC_KMODS     += kmecch_krb5
387 MISC_KMODS     += fssnap_if
388 MISC_KMODS     += hidparser kbtrans usba usba10 usbs49_fw
389 MISC_KMODS     += sl394
390 MISC_KMODS     += hpcsvc pcihp
391 MISC_KMODS     += rsmops

```



```

392 MISC_KMODS      += kcf
393 MISC_KMODS      += ksocket
394 MISC_KMODS      += ibcm
395 MISC_KMODS      += ibdm
396 MISC_KMODS      += ibdma
397 MISC_KMODS      += ibmf
398 MISC_KMODS      += ibtl
399 MISC_KMODS      += sol_ofs
400 MISC_KMODS      += idm
401 MISC_KMODS      += idmap
402 MISC_KMODS      += hook
403 MISC_KMODS      += neti
404 MISC_KMODS      += ctf
405 MISC_KMODS      += mac_dls
406 MISC_KMODS      += cmlb
407 MISC_KMODS      += tem
408 MISC_KMODS      += pcicfg_fcodem_fcpci
409 MISC_KMODS      += scsi_vhci_f_sym scsi_vhci_f_tpgs scsi_vhci_f_asym_sun
410 MISC_KMODS      += scsi_vhci_f_sym_hds
411 MISC_KMODS      += scsi_vhci_f_tape scsi_vhci_f_tpgs_tape
412 MISC_KMODS      += fctl
413 MISC_KMODS      += emlxs_fw
414 MISC_KMODS      += qlc_fw_2200
415 MISC_KMODS      += qlc_fw_2300
416 MISC_KMODS      += qlc_fw_2400
417 MISC_KMODS      += qlc_fw_2500
418 MISC_KMODS      += qlc_fw_6322
419 MISC_KMODS      += qlc_fw_8100
420 MISC_KMODS      += spuni
421 MISC_KMODS      += hwal480_fw_uwba
422 MISC_KMODS      += mii

424 MISC_KMODS      += klmmod klmops

426 #
427 #       Software Cryptographic Providers (/kernel/crypto):
428 #
429 CRYPTO_KMODS    += aes
430 CRYPTO_KMODS    += arcfour
431 CRYPTO_KMODS    += blowfish
432 CRYPTO_KMODS    += des
433 CRYPTO_KMODS    += md4
434 CRYPTO_KMODS    += md5
435 CRYPTO_KMODS    += ecc
436 CRYPTO_KMODS    += rsa
437 CRYPTO_KMODS    += sha1
438 CRYPTO_KMODS    += sha2
439 CRYPTO_KMODS    += swrand

441 #
442 # IP Policy Modules (/kernel/ipp):
443 #
444 IPP_KMODS       += dlcosmk
445 IPP_KMODS       += flowacct
446 IPP_KMODS       += ipgpc
447 IPP_KMODS       += dscpmk
448 IPP_KMODS       += tokenmt
449 IPP_KMODS       += tswtclmt

451 #
452 # 'Dacf' modules (/kernel/dacf)
453 DACF_KMODS      += consconfig_dacf

455 #
456 #       SVVS Testing Modules (/kernel/strmod):
457 #

```

```

458 #       These are streams and driver modules which are not to be
459 #       delivered with a released system. However, during development
460 #       it is convenient to build and install the SVVS kernel modules.
461 #
462 SVVS_KMODS      += lmodb lmode lmodr lmodt svvslo tidg tivc tmuX

464 #
465 #       Modules eXcluded from the product:
466 #
467 XMODS           +=

469 #
470 #       'Dacf' Modules (/kernel/dacf):
471 #
472 DACF_KMODS      += net_dacf

474 #
475 #       MAC-Type Plugin Modules (/kernel/mac)
476 #
477 MAC_KMODS       += mac_6to4
478 MAC_KMODS       += mac_ether
479 MAC_KMODS       += mac_ipv4
480 MAC_KMODS       += mac_ipv6
481 MAC_KMODS       += mac_wifi
482 MAC_KMODS       += mac_ib

484 #
485 # socketmod (kernel/socketmod)
486 #
487 SOCKET_KMODS    += sockpfp
488 SOCKET_KMODS    += socksctp
489 SOCKET_KMODS    += socksdp
490 SOCKET_KMODS    += sockrds
491 SOCKET_KMODS    += ksslf
492 SOCKET_KMODS    += httpfilt
493 SOCKET_KMODS    += datafilt

495 #
496 #       kiconv modules (/kernel/kiconv):
497 #
498 KICONV_KMODS    += kiconv_emea kiconv_ja kiconv_ko kiconv_sc kiconv_tc

500 #
501 # Ensure that the variable member of the cpu_t (cpu_m) is defined
502 # for the lint builds so as not to cause lint errors during the
503 # global cross check.
504 #
505 $(LINTFLAGSSUPPRESS)LINTFLAGS += -D_MACHDEP -I$(UTSBASE)/sun4 \
506                                -I$(UTSBASE)/sun4u -I$(UTSBASE)/sfmmu

```

```

*****
1435 Sat Nov 22 13:16:59 2014
new/usr/src/uts/sparc/datafilt/Makefile
1403 Support deferring TCP accept()s
*****
1 #
2 # This file and its contents are supplied under the terms of the
3 # Common Development and Distribution License ("CDDL"), version 1.0.
4 # You may only use this file in accordance with the terms of version
5 # 1.0 of the CDDL.
6 #
7 # A full copy of the text of the CDDL should have accompanied this
8 # source. A copy of the CDDL is also available via the Internet at
9 # http://www.illumos.org/license/CDDL.
10 #
11 #
12 #
13 # Copyright (c) 2011, OmniTI Computer Consulting, Inc. All rights reserved.
14 # Copyright 2012, Nexenta Systems, Inc. All rights reserved.
15 #
16 #
17 #
18 # Path to the base of the uts directory tree (usually /usr/src/uts).
19 #
20 UTSBASE = ../..
21 #
22 #
23 # Define the module and object file sets.
24 #
25 MODULE          = datafilt
26 OBJECTS         = $(DATAFILT_OBJS:%=$(OBJS_DIR)/%)
27 LINTS           = $(DATAFILT_OBJS:%.o=$(LINTS_DIR)/%.ln)
28 ROOTMODULE     = $(ROOT SOCK_DIR)/$(MODULE)
29 #
30 #
31 # Include common rules.
32 #
33 include $(UTSBASE)/sparc/Makefile.sparc
34 #
35 #
36 # Define targets
37 #
38 ALL_TARGET      = $(BINARY)
39 LINT_TARGET     = $(MODULE).lint
40 INSTALL_TARGET  = $(BINARY) $(ROOTMODULE)
41 #
42 #
43 # lint pass one enforcement and OS version
44 #
45 CFLAGS += $(CCVERBOSE)
46 #
47 LDFLAGS += -dy -Nfs/sockfs -Ndrv/ip
48 #
49 #
50 # Default build targets.
51 #
52 .KEEP_STATE:
53 #
54 def:            $(DEF_DEPS)
55 #
56 all:           $(ALL_DEPS)
57 #
58 clean:         $(CLEAN_DEPS)
59 #
60 clobber:       $(CLOBBER_DEPS)

```

```

62 lint:         $(LINT_DEPS)
63 #
64 modlintlib:   $(MODLINTLIB_DEPS)
65 #
66 clean.lint:   $(CLEAN_LINT_DEPS)
67 #
68 install:      $(INSTALL_DEPS)
69 #
70 #
71 # Include common targets.
72 #
73 include $(UTSBASE)/sparc/Makefile.targ

```

```

*****
1435 Sat Nov 22 13:16:59 2014
new/usr/src/uts/sparc/httpfilt/Makefile
1403 Support deferring TCP accept()s
*****
1 #
2 # This file and its contents are supplied under the terms of the
3 # Common Development and Distribution License ("CDDL"), version 1.0.
4 # You may only use this file in accordance with the terms of version
5 # 1.0 of the CDDL.
6 #
7 # A full copy of the text of the CDDL should have accompanied this
8 # source. A copy of the CDDL is also available via the Internet at
9 # http://www.illumos.org/license/CDDL.
10 #
11 #
12 #
13 # Copyright (c) 2011, OmniTI Computer Consulting, Inc. All rights reserved.
14 # Copyright 2012, Nexenta Systems, Inc. All rights reserved.
15 #
16 #
17 #
18 # Path to the base of the uts directory tree (usually /usr/src/uts).
19 #
20 UTSBASE = ../..
21 #
22 #
23 # Define the module and object file sets.
24 #
25 MODULE = httpfilt
26 OBJECTS = $(HTTFFILT_OBJS:%=$(OBJS_DIR)/%)
27 LINTS = $(HTTFFILT_OBJS:%.o=$(LINTS_DIR)/%.ln)
28 ROOTMODULE = $(ROOT SOCK_DIR)/$(MODULE)
29 #
30 #
31 # Include common rules.
32 #
33 include $(UTSBASE)/sparc/Makefile.sparc
34 #
35 #
36 # Define targets
37 #
38 ALL_TARGET = $(BINARY)
39 LINT_TARGET = $(MODULE).lint
40 INSTALL_TARGET = $(BINARY) $(ROOTMODULE)
41 #
42 #
43 # lint pass one enforcement and OS version
44 #
45 CFLAGS += $(CCVERBOSE)
46 #
47 LDFLAGS += -dy -Nfs/sockfs -Ndrv/ip
48 #
49 #
50 # Default build targets.
51 #
52 .KEEP_STATE:
53 #
54 def: $(DEF_DEPS)
55 #
56 all: $(ALL_DEPS)
57 #
58 clean: $(CLEAN_DEPS)
59 #
60 clobber: $(CLOBBER_DEPS)

```

```

62 lint: $(LINT_DEPS)
63 #
64 modlintlib: $(MODLINTLIB_DEPS)
65 #
66 clean.lint: $(CLEAN_LINT_DEPS)
67 #
68 install: $(INSTALL_DEPS)
69 #
70 #
71 # Include common targets.
72 #
73 include $(UTSBASE)/sparc/Makefile.targ

```