

Re: Filter Hook

Source:

<http://www.tech-archive.net/Archive/Development/microsoft.public.development.device.drivers/2006-08/msg00601.html>

- *From:* hanzhu <hanzhu@xxxxxxxxxx>
 - *Date:* Mon, 07 Aug 2006 16:10:32 +0800
-

OK, Seems I got your problem.

For XP and the later, the callback routine is been invoked at DISPATCH_LEVEL. And you have to return the policy from your callback routine when it returns. However, just as what have been mentioned by the folks, you couldn't wait at DISPATCH_LEVEL. If you really have to keep such a queue for the packets, then the IP filter driver is not suitable for your needs since you have to process all the packets in the callback routine without any wait action(KeWaitFor...)and return it to the tcpip driver immediately when the callback routine returns.

Why not consider the NDISIM driver? It's supported by MS and I'm convinced it will fit for your requirement.

x856256@xxxxxxxxxx Ð´µÀ:

I tried at the beginning to process the packets on the same thread, but it caused the system to crash every time.

Since I found it very hard to program in kernel-mode, and even harder to debug it, I wanted to let a user-mode program process the packet and wait till it finishes in order to return. Was that a bad idea? If it is a bad idea, it means I should do all the processing in the driver, and on the same thread. In such a scenario I will have to synchronize somewhere because my packet process depends on other packets that were previously processed (so I must have a synchronized queue of packets) and again I have the same problem as the first one (or do I ?)

Thanks,
Guy

hanzhu wrote:

IIRC, IP filter driver should process these packets through the callback routine instead of submitting the IRP to the lower tcpip protocol driver. Why do you need to pass the IRPs and wait for its completion?

x856256@xxxxxxxxxx Ð´µÀ:

Hello,

Either queueing the job or doing it asynchronously I will have to do

Re: Filter Hook

the job in another thread, and wait for the process of the packet to finish (in order to return the drop, forward or pass reply), and this is exactly where my problem began, when I tried to use the KeWaitForSingleObject function. So let me ask this again: How can I use this wait function, or any other substitution that will allow me process the packet on another thread (queued or not) and wait for it to finish processing in order to return the value?

Thanks,
Guy

Doron Holan [MS] wrote:

you can't reduce IRQL inline, you can only raise it and then lower it. if you truly want to synchronously process something, then queue it to a work item. if you want your filter to not negatively affect performance, learn how to process the results in the completion routine asynchronously

d

--

Please do not send e-mail directly to this alias. this alias is for newsgroup purposes only. This posting is provided "AS IS" with no warranties, and confers no rights.

<x856256@xxxxxxxx> wrote in message
news:1154873993.596484.241370@xx

No,

The code is not taking any spinlocks.
Anyone has an idea how can I reduce the IRQL level inside the filter hook function?

Thanks,

Re: Filter Hook

Guy

Don Burn wrote:

You cannot
call a lot of
functions at
DISPATCH_LEVEL.
I am not
well
versed
in filter
hooks, so
you need to
determine if
the raising
to
DISPATCH
is
avoidable.
For
instance, if
the code
takes a
spinlock,
could it take
an
ERESOURCE
instead.

--

Don Burn
(MVP,
Windows
DDK)
Windows
2k/XP/2k3
Filesystem
and Driver
Consulting
<http://www.windrvr.com>
Remove
StopSpam
from the
email to
reply

<x856256@xxxxxxxxxx>

Re: Filter Hook

as
they
are
presented
to
it.
You
should
not
try
to
use
IoConnectInterrupt
with
a
software
interrupt.

What
IRQL
are
you
running
at
when
you
crash,
a
common
problem
here
would
be
calling
the
functions
you
mentioned
at
IRQL_DISPATCH
or
higher.

--
Don
Burn
(MVP,
Windows
DDK)
Windows

Re: Filter Hook

2k/XP/2k3
Filesystem
and
Driver
Consulting
<http://www.windrvr.com>
Remove
StopSpam
from
the
email
to
reply

<x856256@xxxxxxxx>
wrote
in
message
<news:1154814887.364085.142850@xxxxxxxxxxxxxxxx>

Hi,

Drivers
are
very
new
to
me,
can
someone
help
me
with
the
next
issue
please:

I
have
tried
to
set
up
a
Filter
Hook
driver.
I

Re: Filter Hook

took
an
example
code
from
codeproject
(DrvFltIp)
and
it
works
ok
on
my
computer
(Windows
XP).

When
I
try
to
process
the
packets
more
thoroughly
than
in
this
example
code,
my
computer
crashed
(freezes
or
restarts
immediately).
The
crash
happens
when
I
do
something
that
involves
a
KeWaitForSingleObject
or
a

Re: Filter Hook

KeDelayExecutionThread
call
(but
I
am
not
sure
these
are
the
only
scenarios
that
cause
a
crash).

I
looked
in
the
dump
file
with
WinDbg
after
the
crash
and
saw
that
one
of
the
last
actions
of
the
kernel
were
sending
an
interrupt
and
afterwards
an
exception.
So
my
guess
was

Re: Filter Hook

that
the
kernel
is
sending
a
trap
that
was
sent
to
it
by
the
hardware
to
my
driver
to
handle,
but
since
I
didn't
set
up
an
interrupt
call
back
function,
an
exception
is
raised.
The
next
thing
I
did
was
to
try
and
set
up
an
interrupt
call
back
function,

Re: Filter Hook

Re: Filter Hook

but
this
also
caused
a
similar
crash
to
the
system
(when
I
tried
to
pull
out
the
parameters
that
I
have
to
pass
to
the
function
"IoConnectInterrupt"
from
the
isrStack).
I
tried
to
call
the
function
"IoConnectInterrupt"
from
the
function
"DriverObject->MajorFunction[IRP_MJ_CR
when
got
"IRP_MJ_CREATE"
in
"irpStack->MajorFunction"
or
when
I
got
"IRP_MN_START_DEVICE

Re: Filter Hook

IRP"
in
"irpStack->MinorFunction".
When
I
tried
even
to
access
the
parameter
"irpStack->Parameters->StartDevice.Alloc
(even
without
calling
"IoConnectInterrupt")
it
caused
a
crash.

If
anyone
can
help
me
understand
what
I
am
doing
wrong,
and
why
I
am
getting
all
these
crashes
I
would
be
grateful.

Thanks,
Guy

Re: Filter Hook