pfSense - Bug #2762

PF drops IPv6 packets with fragment header followed by a last fragment only

01/18/2013 03:25 AM - Chris Buechler

Status:	Resolved	**Start date:**	01/18/2013
Priority:	Normal	**Due date:**	
Assignee:	Luiz Souza	**% Done:**	0%
Category:	Operating System	**Estimated time:**	0.00 hour
Target version:	2.3		
Affected Version:	2.1-IPv6	**Affected Architecture:**	All

Description

PF has the same problem as is described here for ipfw.
http://lists.freebsd.org/pipermail/freebsd-net/2011-February/027838.html

This used to be replicable by doing this:
telnet -6 www.allstream.com 80
but as of 201509, this site no longer exhibits this behavior.

They set a frag header, offset = 0, M bit = 0, in all their SYN ACKs for some reason. That's valid per RFC 2460. pcap showing is attached.

PF logs it as follows:

```
Jan 18 02:48:56 fw1 pf: 00:00:00.242205 rule 5/0(match): block in on em0: (flowlabel 0xeb8da, hlim
 56, next-header Fragment (44) payload length: 48) 2607:f4e8:200:12:225:90ff:fe2a:a072 > 2610:160:
11:a033::230: frag (0xb5736529:0|40) 80 > 40842: Flags [S.], seq 3303787714, ack 1052652245, win 6
5535, options [mss 1140,nop,wscale 4,sackOK,TS val 260605935 ecr 179963673], length 0
Jan 18 02:48:59 fw1 pf: 00:00:02.934772 rule 5/0(match): block in on em0: (flowlabel 0xeb8da, hlim
 56, next-header Fragment (44) payload length: 48) 2607:f4e8:200:12:225:90ff:fe2a:a072 > 2610:160:
11:a033::230: frag (0xaf40f4e7:0|40) 80 > 40842: Flags [S.], seq 3303787714, ack 1052652245, win 6
5535, options [mss 1140,nop,wscale 4,sackOK,TS val 260605935 ecr 179963973], length 0
Jan 18 02:49:02 fw1 pf: 00:00:02.999317 rule 5/0(match): block in on em0: (flowlabel 0xeb8da, hlim
 56, next-header Fragment (44) payload length: 48) 2607:f4e8:200:12:225:90ff:fe2a:a072 > 2610:160:
11:a033::230: frag (0xf2d6888d:0|40) 80 > 40842: Flags [S.], seq 3303787714, ack 1052652245, win 6
5535, options [mss 1140,nop,wscale 4,sackOK,TS val 260605935 ecr 179963973], length 0
Jan 18 02:49:02 fw1 pf: 00:00:00.205661 rule 5/0(match): block in on em0: (flowlabel 0xeb8da, hlim
 56, next-header Fragment (44) payload length: 48) 2607:f4e8:200:12:225:90ff:fe2a:a072 > 2610:160:
11:a033::230: frag (0x8009c1bf:0|40) 80 > 40842: Flags [S.], seq 3303787714, ack 1052652245, win 6
5535, options [mss 1140,nop,wscale 4,sackOK,TS val 260605935 ecr 179964293], length 0
Jan 18 02:49:05 fw1 pf: 00:00:02.999839 rule 5/0(match): block in on em0: (flowlabel 0xeb8da, hlim
 56, next-header Fragment (44) payload length: 48) 2607:f4e8:200:12:225:90ff:fe2a:a072 > 2610:160:
11:a033::230: frag (0xe718b255:0|40) 80 > 40842: Flags [S.], seq 3303787714, ack 1052652245, win 6
5535, options [mss 1140,nop,wscale 4,sackOK,TS val 260605935 ecr 179964293], length 0
```

History

#1 - 02/05/2013 01:51 PM - Ermal Luçi

This is scrub in action.
Will see how to make this behave normally.

#2 - 07/04/2013 12:11 PM - Sander Steffann

PS: It is not broken or weird behaviour (according to the RFCs). RFC 6145 (translating IPv4 <-> IPv6) specifies:

> When the IPv4 sender does not set the DF bit, the translator SHOULD
> always include an IPv6 Fragment Header to indicate that the sender
> allows fragmentation. The translator MAY provide a configuration
> function that allows the translator not to include the Fragment

```
   Header for the non-fragmented IPv6 packets.
```

Translators that follow this RFC will generate such atomic fragments, and pfSense will break communication with them.

Also see RFC 6946 (Processing of IPv6 "Atomic" Fragments)

#3 - 07/05/2013 05:23 AM - Ermal Luçi

The only option for now seems to create rules with allow-option advanced setting set.

#4 - 07/12/2013 03:19 AM - Doktor Notor

Sadly, I keep hitting this with http://snapshots.pfsense.org:

```
Jul 12 10:15:58 gw pf: 00:00:07.982521 rule 5/0(match): block in on gif0: (flowlabel 0x3dfd0, hlim 55, next-he
ader Fragment (44) payload length: 36) 2610:1c0:1:25::51 > 2001:470:6f:xxx:yyy::zzz frag (0xbe5ada46:1432|28)
Jul 12 10:15:58 gw pf: 00:00:00.000058 rule 5/0(match): block in on gif0: (flowlabel 0x3dfd0, hlim 55, next-he
ader Fragment (44) payload length: 36) 2610:1c0:1:25::51 > 2001:470:6f:xxx:yyy::zzz frag (0x86c96d61:1432|28)
Jul 12 10:15:58 gw pf: 00:00:00.000700 rule 5/0(match): block in on gif0: (flowlabel 0x3dfd0, hlim 55, next-he
ader Fragment (44) payload length: 1440) 2610:1c0:1:25::51 > 2001:470:6f:xxx:yyy::zzz frag (0xbe5ada46:0|1432)
 80 > 63352: Flags [.], ack 982233318, win 65535, length 1412
Jul 12 10:15:58 gw pf: 00:00:00.000123 rule 5/0(match): block in on gif0: (flowlabel 0x3dfd0, hlim 55, next-he
ader Fragment (44) payload length: 1440) 2610:1c0:1:25::51 > 2001:470:6f:xxx:yyy::zzz frag (0x86c96d61:0|1432)
 80 > 63352: Flags [.], ack 982233318, win 65535, length 1412
Jul 12 10:16:02 gw pf: 00:00:03.688589 rule 5/0(match): block in on gif0: (flowlabel 0x3dfd0, hlim 55, next-he
ader Fragment (44) payload length: 1440) 2610:1c0:1:25::51 > 2001:470:6f:xxx:yyy::zzz frag (0x94e13b01:0|1432)
 80 > 63352: Flags [.], ack 982233318, win 65535, length 1412
Jul 12 10:16:02 gw pf: 00:00:00.000058 rule 5/0(match): block in on gif0: (flowlabel 0x3dfd0, hlim 55, next-he
ader Fragment (44) payload length: 36) 2610:1c0:1:25::51 > 2001:470:6f:xxx:yyy::zzz frag (0x94e13b01:1432|28)
Jul 12 10:16:11 gw pf: 00:00:03.440808 rule 5/0(match): block in on gif0: (flowlabel 0x3dfd0, hlim 55, next-he
ader Fragment (44) payload length: 1440) 2610:1c0:1:25::51 > 2001:470:6f:xxx:yyy::zzz frag (0xfdfc4caf:0|1432)
 80 > 63352: Flags [.], ack 982233318, win 65535, length 1412
Jul 12 10:16:11 gw pf: 00:00:00.000060 rule 5/0(match): block in on gif0: (flowlabel 0x3dfd0, hlim 55, next-he
ader Fragment (44) payload length: 36) 2610:1c0:1:25::51 > 2001:470:6f:xxx:yyy::zzz frag (0xfdfc4caf:1432|28)
Jul 12 10:16:43 gw pf: 00:00:03.721811 rule 5/0(match): block in on gif0: (flowlabel 0x3dfd0, hlim 55, next-he
ader Fragment (44) payload length: 1440) 2610:1c0:1:25::51 > 2001:470:6f:xxx:yyy::zzz frag (0xf4c1f620:0|1432)
 80 > 63352: Flags [.], ack 982233318, win 65535, length 1412
Jul 12 10:16:43 gw pf: 00:00:00.000059 rule 5/0(match): block in on gif0: (flowlabel 0x3dfd0, hlim 55, next-he
ader Fragment (44) payload length: 1440) 2610:1c0:1:25::51 > 2001:470:6f:xxx:yyy::zzz frag (0xbbcd08b4:0|1432)
 80 > 63352: Flags [.], ack 982233318, win 65535, length 1412
Jul 12 10:16:43 gw pf: 00:00:00.000052 rule 5/0(match): block in on gif0: (flowlabel 0x3dfd0, hlim 55, next-he
ader Fragment (44) payload length: 36) 2610:1c0:1:25::51 > 2001:470:6f:xxx:yyy::zzz frag (0xf4c1f620:1432|28)
Jul 12 10:16:43 gw pf: 00:00:00.000049 rule 5/0(match): block in on gif0: (flowlabel 0x3dfd0, hlim 55, next-he
ader Fragment (44) payload length: 36) 2610:1c0:1:25::51 > 2001:470:6f:xxx:yyy::zzz frag (0xbbcd08b4:1432|28)
Jul 12 10:16:44 gw pf: 00:00:00.012077 rule 5/0(match): block in on gif0: (flowlabel 0x1f70f, hlim 55, next-he
ader Fragment (44) payload length: 36) 2610:1c0:1:25::51 > 2001:470:6f:xxx:yyy::zzz frag (0xdc6772be:1432|28)
Jul 12 10:16:44 gw pf: 00:00:00.000603 rule 5/0(match): block in on gif0: (flowlabel 0x1f70f, hlim 55, next-he
ader Fragment (44) payload length: 1440) 2610:1c0:1:25::51 > 2001:470:6f:xxx:yyy::zzz frag (0xdc6772be:0|1432)
 80 > 63354: Flags [.], ack 3682091665, win 65535, length 1412
Jul 12 10:17:06 gw pf: 00:00:07.393108 rule 5/0(match): block in on gif0: (flowlabel 0x3dfd0, hlim 55, next-he
ader Fragment (44) payload length: 36) 2610:1c0:1:25::51 > 2001:470:6f:xxx:yyy::zzz frag (0xaf182ff3:1432|28)
Jul 12 10:17:06 gw pf: 00:00:00.000549 rule 5/0(match): block in on gif0: (flowlabel 0x3dfd0, hlim 55, next-he
ader Fragment (44) payload length: 1440) 2610:1c0:1:25::51 > 2001:470:6f:xxx:yyy::zzz frag (0xaf182ff3:0|1432)
 80 > 63352: Flags [.], ack 982233318, win 65535, length 1412
Jul 12 10:17:48 gw pf: 00:00:02.428094 rule 5/0(match): block in on gif0: (flowlabel 0x1f70f, hlim 55, next-he
ader Fragment (44) payload length: 1440) 2610:1c0:1:25::51 > 2001:470:6f:xxx:yyy::zzz frag (0x47368efe:0|1432)
 80 > 63354: Flags [.], ack 3682091665, win 65535, length 1412
Jul 12 10:17:48 gw pf: 00:00:00.001648 rule 5/0(match): block in on gif0: (flowlabel 0x1f70f, hlim 55, next-he
ader Fragment (44) payload length: 36) 2610:1c0:1:25::51 > 2001:470:6f:xxx:yyy::zzz frag (0x47368efe:1432|28)
Jul 12 10:18:52 gw pf: 00:00:07.936447 rule 5/0(match): block in on gif0: (flowlabel 0x1f70f, hlim 55, next-he
ader Fragment (44) payload length: 1440) 2610:1c0:1:25::51 > 2001:470:6f:xxx:yyy::zzz frag (0x3a1cb4de:0|1432)
 80 > 63354: Flags [.], ack 3682091665, win 65535, length 1412
Jul 12 10:18:52 gw pf: 00:00:00.000060 rule 5/0(match): block in on gif0: (flowlabel 0x1f70f, hlim 55, next-he
ader Fragment (44) payload length: 36) 2610:1c0:1:25::51 > 2001:470:6f:xxx:yyy::zzz frag (0x3a1cb4de:1432|28)
```

Rather annoying. +1 on lets stop blocking this.

#5 - 08/01/2013 10:25 AM - Ermal Luçi

- Target version changed from 2.1 to 2.2

This cannot be solved for now apart the workaround to allow fragments.

#6 - 08/01/2013 10:41 AM - Doktor Notor

I seriously don't really care whether it's a workaround or not... How I can prevent pf from dropping legitimate traffic!?

#7 - 01/12/2014 08:16 AM - Doktor Notor

Erm, guys, what's up with this?! Upstream apparently does NOT intend to fix this in any way, cf. http://www.freebsd.org/cgi/query-pr.cgi?pr=124933 and they do not intend to port pf-related updates from OpenBSD either (http://www.freebsd.org/cgi/query-pr.cgi?pr=167057&cat=kern and the fabulous bitchfest @ http://lists.freebsd.org/pipermail/freebsd-pf/2012-September/006740.html). This just "rocks".

How do I create something like

pass in on <iface> inet6 proto ipv6-frag all

via the GUI? This is breaking web pages, hit this all the time, oh the irony, with pfsense.org website itself. This issue also makes http://www.o2.cz completely unusable for me with IPv6 enabled!

#8 - 01/12/2014 08:21 AM - Doktor Notor

And another one on the broken scrub: http://www.freebsd.org/cgi/query-pr.cgi?pr=172648

#9 - 06/15/2014 09:06 PM - Jim Thompson

- Assignee set to Ermal Luçi

#10 - 11/10/2014 04:48 AM - Jens Groh

Just FYI:

The official bug (https://bugs.freebsd.org/bugzilla/show_bug.cgi?id=172648) got another mention:

-> https://lists.freebsd.org/pipermail/freebsd-net/2014-November/040319.html

Perhaps some of this could be integrated/backported to v10 as this was developed against FreeBSD 11-current.

#11 - 11/29/2014 12:53 AM - Jim Thompson

- Affected Documentation 0 added

Jens,

If you look at that thread, Ermal has the fix in-hand.

IJS...

#12 - 12/04/2014 11:47 AM - Chris Buechler

- Target version changed from 2.2 to 2.2.1

#13 - 01/16/2015 12:45 AM - Jim Thompson

- Target version changed from 2.2.1 to 2.3

- Affected Architecture set to All

I think this is going to want more testing than what we can afford in the 2.2.1 timeframe. That said, if it gets fixed for 2.2.1, I'm not upset.

#14 - 05/08/2015 03:55 AM - Klaus Steinberger

What is the time frame for fixing this? I was hit by this bug now by adding dnssec NSEC3 to my DNS which enlarged the payloads of my DNS Server over the MTU.

#15 - 08/04/2015 03:23 AM - Christian Felsing

- File firewall_rules_edit.diff added

I modified */usr/local/www/firewall_rules_edit.php* with enclosed patch (pfSense 2.2.4)

After that, protocol "IPV6-FRAG" is offered in GUI. A test rule with a Sixxs host did desired modification in pf ruleset. **pfctl -sa | less** shows

pass in log quick on em0 inet6 proto ipv6-frag from 2a01:258:8:2::4 to any keep state label "USER_RULE"

and **ping6 -n -s 2000 2a01:258:8:2::4** works fine.

Use this on your own risk, Due to application to WAN interface security issues may arise.

#16 - 08/31/2015 09:21 PM - Jim Thompson

- Assignee changed from Ermal Luçi to Luiz Souza

reassigned to Luiz. Maybe this is fixed in 10.2 (I'm looking mostly at MFS 286079 / r285999)

#17 - 09/22/2015 01:37 AM - Chris Buechler

- Description updated

- Status changed from New to Feedback

the real world systems I'm aware of that exhibited this behavior no longer do, due to infrastructure changes on their end. Original site in this ticket doesn't, IPv6 IP in the pcap no longer alive. I'm not aware of a replicable real-world scenario, but is testable by replaying the pcap.

I suspect this was fixed in the general fixing of pf+IPv6 fragmentation in FreeBSD, but needs confirmation.

If anyone's aware of a real-world site exhibiting the behavior, let us know.

#18 - 09/22/2015 01:49 AM - Klaus Steinberger

Chris Buechler wrote:

> the real world systems I'm aware of that exhibited this behavior no longer do, due to infrastructure changes on their end. Original site in this ticket doesn't, IPv6 IP in the pcap no longer alive. I'm not aware of a replicable real-world scenario, but is testable by replaying the pcap.
>
> I suspect this was fixed in the general fixing of pf+IPv6 fragmentation in FreeBSD, but needs confirmation.
>
> If anyone's aware of a real-world site exhibiting the behavior, let us know.

We do run a DNS Server with IPV6 and DNSSEC. to complete the the tests on dnsviz.net succesful we had to decrease the size of UDP answers in bind with these parameters in named.conf:

```
/*         *  Work around pfsense IPV6 fragmentation problem
*/
edns-udp-size   1024;
max-udp-size    1024;
```

If we get a fixed up pfsense version, we can verify the fix just with commenting out the parameters in bind and redoing the tests on dnsviz.net

Sincerly,
Klaus

#19 - 09/22/2015 01:59 AM - Chris Buechler

Klaus Steinberger wrote:

> We do run a DNS Server with IPV6 and DNSSEC. to complete the the tests on dnsviz.net succesful we had to decrease the size of UDP answers

That's the general issues with IPv6 fragmentation. This bug ticket is specific to the circumstance described in the original post, which isn't actually fragmented traffic at all.

2.3 has the fixes included in FreeBSD for general IPv6 fragmentation handling with pf, so your issue should be resolved in 2.3. Snapshots will be publicly available soon. Definitely would appreciate any feedback from your usage case on the 2.3 board of the forum (when it exists).

#20 - 11/04/2015 04:52 PM - Sander Steffann

PS: for those who want to test with a website that sends fragments try www.cbs.nl. It has an RFC 6145 SIIT box in front of it that always adds a fragmentation header.

Relevant part of the RFC:
When the IPv4 sender does not set the DF bit, the translator SHOULD
always include an IPv6 Fragment Header to indicate that the sender
allows fragmentation.

So it's a reliable source of fragments (until they change the implementation)

#21 - 11/18/2015 05:35 PM - Chris Buechler

- Status changed from Feedback to Resolved

Thanks Sander, that helps. This is definitely fixed in 2.3.

Files

allstream.pcap	6.21 KB	01/18/2013	Chris Buechler
firewall_rules_edit.diff	749 Bytes	08/04/2015	Christian Felsing