Tails - Bug #12362
MAC spoofing cannot be disabled for devices using modules from the staging directory
03/17/2017 07:24 PM - sajolida

Status: Resolved  Start date: 03/17/2017
Priority: Elevated  Due date: 
Assignee:  % Done: 100%
Category: Spoof MAC  Estimated time: 0.00 hour
Target version: Tails_3.0-rc1  Spent time: 0.00 hour
Feature Branch: bugfix/12362-mac-spoofing-vs-staging-drivers  Starter:
Type of work: Research  Affected tool: 
Blueprint: 

Description
As of 617151fc, if I:

- Start Tails with a USB WiFi dongle that doesn't support MAC spoofing
- Disable MAC spoofing in Tails Greeter

Then:

- MAC spoofing is still enabled and I can't use that card.

But, if I plug the USB WiFi dongle once the session is started. MAC spoofing is not applied to the new interface and I can use that card.

Associated revisions
Revision f5bae43a - 04/05/2017 04:26 PM - anonym
Test suite: test that MAC spoofing works for hotplugged devices.
Refs: #12362

Revision 89f38715 - 05/19/2017 04:10 PM - anonym
Try to automatically identify network drivers in the staging directory.
... so they can be blacklisted without us having to manually maintain this list.
Will-fix: #12362

Revision b21ab0e2 - 05/19/2017 05:08 PM - intrigeri
Fix typo (refs: #12362).

Revision 1c1b1991 - 05/19/2017 05:18 PM - intrigeri
Refactor 80-block-network to avoid a function whose name ends with "filter" to modify the filtered lines (refs: #12362).

Revision d68ff5c5 - 05/19/2017 05:20 PM - intrigeri
Reuse code instead of duplicating it (refs: #12362).
Refactor 80-block-network to be less dependent on state and code ordering.
refs: #12362

Fix another typo (refs: #12362).

Merge branch 'bugfix/12362-mac-spoofing-vs-staging-drivers' into feature/stretch
Fix-committed: #12362

Subject changed from MAC spoofing not disabled to MAC spoofing not disabled in 3.0~beta3

Subject changed from MAC spoofing not disabled in 3.0~beta3 to MAC spoofing cannot be disabled in 3.0~beta3

Subject changed from MAC spoofing cannot be disabled in 3.0~beta3 to MAC spoofing cannot be disabled
- Target version changed from Tails_3.0 to Tails_2.12

Actually it's the same on 2.11 :)

Status changed from Confirmed to In Progress
- Assignee changed from anonym to sajolida
- % Done changed from 0 to 10
- QA Check set to Info Needed
- Feature Branch set to test/12362-hotplug-vs-mac-spoofing

I couldn't for the life of me find my physical USB wifi dongle, so I could only test with virtual hardware. I even wrote automated tests, and they cannot reproduce this bug. Can you please retry? This time, please provide the output of macchanger DEV where DEV is the name of the interface device, e.g. eth0, wlan0 whatever.

Meanwhile, let's see what Jenkins thinks of the MAC address spoofing is disabled scenario for the feature branch.

File 12362.pgp added
- Assignee changed from sajolida to anonym
- QA Check changed from Info Needed to Dev Needed

I confirm that the bug is still here in 2.12. I cannot do macchanger DEV because the module for my dongle is removed and the device does not appear in ifconfig. My internal Wi-Fi card does appear with its permanent MAC address. I'm attaching to this ticket a full journalctl from this session, encrypted for you, intrigeri, and me. I hope that's useful.
#6 - 04/15/2017 07:10 AM - intrigeri

I'm attaching to this ticket a full journalctl from this session, encrypted for you, intrigeri, and me. I hope that's useful.

I hope you meant s/intrigeri/anonym/:) 

#7 - 04/19/2017 05:40 PM - anonym
- Target version changed from Tails_2.12 to Tails_3.0~rc1

#8 - 04/20/2017 02:28 PM - anonym
- Subject changed from MAC spoofing cannot be disabled to MAC spoofing cannot be disabled for devices using modules from the staging directory
- Assignee changed from anonym to sajolida
- QA Check changed from Dev Needed to Info Needed

So sajolida's r8188eu module is in the staging directory (/lib/modules/*/kernel/drivers/staging/) which currently isn't added to /etc/modprobe.d/all-net-blacklist.conf which explains everything. Looking at config/chroot_local-hooks/80-block-network, which generates that file. Yay, blacklist approaches suck. :) 

I don't see any general solution. The problem that needs to be solved is to tell whether a module is a network device driver or not. I'm afraid the best I can do for now is to add the obvious candidates /lib/modules/*/kernel/drivers/staging/*' to the list of directories where we treat all modules as such drivers (which is the case at the moment). This is implemented in the new feature branch (which is based on the old one, so we'll get the new test, FWIW).

sajolida, would you be willing to test an image once Jenkins has built one?

#9 - 04/22/2017 10:19 AM - sajolida
- Assignee changed from sajolida to anonym

Yes!

#10 - 04/22/2017 11:02 AM - anonym
- Assignee changed from anonym to sajolida
- Feature Branch changed from test/12362-hotplug-vs-mac-spoofing to bugfix/12362-mac-spoofing-vs-staging-drivers

Seems I messed up pushing the new branch, so I'll monitor if it actually builds and seems to create the intended blacklist before asking you to test it. BTW, IIRC you do not like running ISOs from jenkins; do you want me to build + sign and upload an image somewhere?

#11 - 04/23/2017 09:00 AM - sajolida
So I'm downloading it. I don't need it to be sign by you because I'll only use it for testing.

Just being curious: did you figure out why plugging the USB once the session is already started leads to the USB stick being usable (without its MAC not being spoofed of course)?

#12 - 05/02/2017 12:52 PM - sajolida
- Assignee changed from sajolida to anonym
- QA Check deleted (Info Needed)

I tested your ISO and it works as expected, cool!

Still, it made me spot #12500 ;(

#13 - 05/16/2017 04:49 PM - intrigeri
- Assignee changed from anonym to intrigeri
- % Done changed from 10 to 50
- QA Check set to Ready for QA

#14 - 05/16/2017 05:14 PM - intrigeri
- Assignee changed from intrigeri to anonym
- % Done changed from 50 to 60
- QA Check changed from Ready for QA to Dev Needed

Code review passes, except I'm not 100% convinced by /lib/modules/*/kernel/drivers/staging/rtl*:

- I see /lib/modules/4.9.0-3-amd64/kernel/drivers/staging/wlan-ng/prism2_usb.ko on my system: shouldn't it be blacklisted too?
- How will we update this list? Maybe add something about it to the release process doc?

#15 - 05/19/2017 04:19 PM - anonym
- Assignee changed from anonym to intrigeri
- QA Check changed from Dev Needed to Ready for QA

intrigeri wrote:

Code review passes, except I'm not 100% convinced by /lib/modules/*/kernel/drivers/staging/rtl*:

- I see /lib/modules/4.9.0-3-amd64/kernel/drivers/staging/wlan-ng/prism2_usb.ko on my system: shouldn't it be blacklisted too?

And vt6656/vt6656_stage.ko.

- How will we update this list? Maybe add something about it to the release process doc?

Urgh. I really don't want this if it can be avoided so I implemented an automatic approach, and documented its (reasonable for this context, IMHO) shortcomings. What do you think?

It can easily be tested on your own system by changing the path of the blacklist in the script to something writable. This way you can compare and diff, and you should get something like:
install pppox /bin/true
install ppp_synctty /bin/true
install pptp /bin/true
+install prism2_usb /bin/true
install qed /bin/true
install qede /bin/true
install qla3xxx /bin/true
@@ -255,6 +255,7 @@
@@ -265,6 +266,11 @@
install r6040 /bin/true
install r8152 /bin/true
install r8169 /bin/true
+install r8188eu /bin/true
+install r8192e_pci /bin/true
+install r8192u_usb /bin/true
+install r8712u /bin/true
+install r8723au /bin/true
install ray_cs /bin/true
install realtek /bin/true
install rfcl051 /bin/true
@@ -302,6 +308,10 @@
install r8723-common /bin/true
install r8821ae /bin/true
install rt18xxxu /bin/true
+install rtllib /bin/true
+install rtllib_crypt_ccmp /bin/true
+install rtllib_crypt_tkip /bin/true
+install rtllib_crypt_wep /bin/true
install rtl_pcie /bin/true
install rtl_usb /bin/true
install rtlwifi /bin/true
@@ -368,6 +378,7 @@
install vlsi_ir /bin/true
install vmxnet3 /bin/true
install vrf /bin/true
+install vt6656_stage /bin/true
install vxge /bin/true
install vxlan /bin/true
install w83977af_ir /bin/true

which, AFAICT, is exactly the diff we want.
Urgh. I really don't want this if it can be avoided so I implemented an automatic approach, and documented its (reasonable for this context, IMHO) shortcomings.
What do you think?

I like the idea a lot! I was no big fan of some aspects of the implementation (yeah, nitpicking forever…) so I've pushed some refactoring. The resulting script generates the exact same file on my system. Feel free to merge into feature/stretch yourself if you like it :)

Woo, I really like those refactorings! Merged (+ a tiny simplification with 98658a8 that intrigeri ACKed off-band)!

#19 - 05/23/2017 09:12 AM - intrigeri
- Status changed from Resolved to Resolved

Files

<table>
<thead>
<tr>
<th>File</th>
<th>Size</th>
<th>Date</th>
<th>Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>12362.pgp</td>
<td>50.8 KB</td>
<td>04/14/2017</td>
<td>sajolida</td>
</tr>
</tbody>
</table>