1653251702  
Local IP addr defaults to Last IP address configed. This brks r-cmds
Document Information Table

Submitter Text

```
[FIDR:00234]
 Date:          1998.2.12
 OS REVISION:   LR(IC23E)
 SYSTEM:        T600
 SUBSYSTEM:     Network
 DESCRIPTION:   r-command fail by permission check when cluster
                is running

Problem
=======

When invoking r-command (such as rlogin and rcp) or cm-command of SG
(such as
cmrunnode) to other nodes from the node where package is running and
relocatable IP is set, it sometime fails with "Permission denied".

If adding relocatable IP address to rhost. or cmclnodelist for this
error case, r- or cm- command succeed.
```

Verifier Text

```
This problem occurred in NEC/NTT FITS Project which is very big (more
than 200 T600 systems) and important for both HP and NEC.

As we consider that this problem happens with Network.
SG lab engineer's analysis and comment are as below.

Analysis
========

When this problem happens, the output of "netstat -nn" on the node
(where relocatable IP is set) is as follows:

Routing tables
Dest/Netmask    Gateway         Flags  Refs       Use  Interface  Pmtu
localhost       localhost       UH        0     44548  lo0        4136
HA0BN011        HA0BN011        UH        0         6  lan4       4136
HA0BR011        HA0BR011        UH        0         0  lan4:1     4136
HA0BR020        HA0BR020        UH        0         0  lan8:1     4136
HA0BP001        HA0BP001        UH        0     13064  lan0       4136
HA0BR010        HA0BR010        UH        0         0  lan0:1     4136
HA0BN020        HA0BN020        UH        0        24  lan8       4136
HA0BN040        HA0BN040        UH        0       188  lan1       4136
10.145.148.48   HA0BN040        U         2         0  lan1       1500

>>>>
10.145.143.0    HA0BR010        U         3         0  lan0:1     4352
10.145.143.0    HA0BP001        U         3         0  lan0       4352
>>>>
10.145.143.128  HA0BR011        U         3         0  lan4:1     4352
10.145.143.128  HA0BN011        U         3         0  lan4       4352
10.145.146.0    HA0BR020        U         3         0  lan8:1    65280
10.145.146.0    HA0BN020        U         3         0  lan8      65280
10.145.145.128  HAB0LG00        UG        0         0  lan4       4352
127.0.0.0       localhost       U         0         0  lo0        4136
default         HAB0LG00        UG        0         0  lan4       4352
```

```
>>>> point shows the network route for the I/F which matches to
the hostname (= HA0B001). When the problem happens, the entry
for relocatable IP is located upper the entry of stationary IP.

Comment from SG lab
===================

At first, I contact MC/SG lab since this problem happens with MC/SG
pre-release version for HP-UX 11.0. SG lab engineer could reproduce
this problem without SG on HP-UX 11.0 using IP aliasing features.


> Although this probelm was seen with SG, I consider it is networking
> TCPIP issue (out side of SG). If so, we should go through the normal
> HP-UX 11.0 support process to accelerate the resolution.
> How do you consider ?

  I agree completely. I've attached to this message a transcript which
  shows the issue completely outside of ServiceGuard (editorial
  comments have been added to the transcript proceeded with >>).
  It relates to ServiceGuard only in so much as we use the IP aliasing
  capability to implement the concept of a "relocatable IP".

> I consider this problem may also happen with official MC/SG A.11.01.
> Do you agree ? If so, probably we can go through the normal HP-UX 11.0
> support process since SG A.11.01 was already officially released.

  Yup, you will see this behavior on HPUX 11.00 based systems so it will
  affect SG A.11.01 as well.


Reproduction in SG lab
======================

>>
>> This is the 10.2 based system. When we start, here are
>> the assigned IP's and the route table.
>>

newman:/tmp> netstat -in
Name  Mtu   Network      Address          Ipkts Ierrs  Opkts Oerrs  Coll
ni0*  0     none         none                 0     0      0     0     0
ni1*  0     none         none                 0     0      0     0     0
lo0   4608  127          127.0.0.1       778727     0 778727     0     0
lan0  1497  15.13.168.0  15.13.169.221 57319108     0 797373     2 12937
lan1  1497  192.168.1    192.168.1.1      65850     0  66257     0     0
lan1  1497  af17: 21.01                   65850     0  66257     0     0
lan2* 1500  none          none             215     0    218     0     0

newman:/tmp> netstat -rn
Routing tables
Destination     Gateway         Flags   Refs     Use  Interface  Pmtu
PmtuTime
15.13.169.221   127.0.0.1       UH         0  315750  lo0        4608
127.0.0.1       127.0.0.1       UH         0  461367  lo0        4608
192.168.1.1     127.0.0.1       UH         0    1610  lo0        4608
default         15.13.168.1     UG         1  166918  lan0       1497
15.13.168.0     15.13.169.221   U         10  592030  lan0       1497
192.168.1       192.168.1.1     U          0    1038  lan1       1497

>>
>> Now we check connectivity to the remote system. This system
>> has our IP address (192.168.1.1) in its rhosts. file.
>>

newman:/tmp> remsh 192.168.1.2 date
Wed Feb 18 19:25:26 PST 1998
```

```
>>
>> That works fine so now add an IP address onto lan1 and show the
>> new state of the IP's and route table. Note there are no new
>> route table entries.
>>

newman:/tmp> ifalias lan1 add 192.168.1.101
newman:/tmp> netstat -in
Name  Mtu   Network      Address          Ipkts Ierrs  Opkts Oerrs  Coll
ni0*  0     none         none                 0     0      0     0     0
ni1*  0     none         none                 0     0      0     0     0
lo0   4608  127          127.0.0.1       778761     0 778761     0     0
lan0  1497  15.13.168.0  15.13.169.221 57320778     0 797522     2 12937
lan1  1497  192.168.1    192.168.1.1      65861     0  66270     0     0
lan1  1497  af17: 21.01                   65861     0  66270     0     0
lan1  1497  192.168.1    192.168.1.5      65861     0  66270     0     0
lan1  1497  192.168.1    192.168.1.101    65861     0  66270     0     0
lan2* 1500  none         none               216     0    218     0     0

newman:/tmp> netstat -rn
Routing tables
Destination     Gateway         Flags   Refs     Use  Interface  Pmtu
PmtuTime
15.13.169.221   127.0.0.1       UH         0  315760  lo0        4608
127.0.0.1       127.0.0.1       UH         0  461391  lo0        4608
192.168.1.1     127.0.0.1       UH         0    1610  lo0        4608
default         15.13.168.1     UG         1  167065  lan0       1497
15.13.168.0     15.13.169.221   U         10  592034  lan0       1497
192.168.1       192.168.1.1     U          0    1048  lan1       1497

>>
>> Now retry the remsh to make sure it still works. As can be seen,
>> there is no change in connectivity

newman:/tmp> remsh 192.168.1.2 date
Wed Feb 18 19:26:53 PST 1998
newman:/tmp> exit


>>
>> Now we try the same experiment on a 11.00 based system (although I
>> suspect we'll get the same results on a 10.3 based system).
>>
>>
>> First the IP addresses and route table.
>

bermese[58]% netstat -in
Name         Mtu Network          Address              Ipkts      Opkts
lo0         4136 127.0.0.0        127.0.0.1          3865766    3865766
lan3        1500 15.13.168.0      15.13.171.235     21595565    9664250
lan2        1500 192.6.15.0       192.6.15.12        3080017    2513790
lan1        1500 192.8.15.0       192.8.15.12        3372648    1996403

bermese[59]% netstat -rn
Routing tables
Dest/Netmask      Gateway         Flags  Refs       Use  Interface  Pmtu
127.0.0.1         127.0.0.1       UH        0   3865773  lo0        4136
192.6.15.12       192.6.15.12     UH        0      7298  lan2       4136
192.8.15.12       192.8.15.12     UH        0         4  lan1       4136
15.13.171.235     15.13.171.235   UH        0   5402719  lan3       4136
192.6.15.0        192.6.15.12     U         2         0  lan2       1500
192.8.15.0        192.8.15.12     U         2         0  lan1       1500
15.13.168.0       15.13.171.235   U         2         0  lan3       1500
default           15.13.171.235   U         0         0  lan3       1500
127.0.0.0         127.0.0.1       U         0         0  lo0        4136
```

```
>>
>> Now the check for remsh capability. Again, the remote node has our
>> IP address (192.8.15.12) in its rhosts. file.
>>

bermese[60]% remsh 192.8.15.11 date
Wed Feb 18 19:21:04 PST 1998

>>
>> Now we add an IP address to lan1 (using the new streams way). Note
>> that a new route associated with this new IP address gets added
>> and that it comes before the route entry for the 192.8.15.12 address.
>>

bermese[61]% ifconfig lan1:1 inet 192.8.15.101

bermese[62]% netstat -in
Name       Mtu Network          Address                Ipkts      Opkts
lo0       4136 127.0.0.0        127.0.0.1             3866112    3866112
lan3      1500 15.13.168.0      15.13.171.235        21601516    9668348
lan2      1500 192.6.15.0       192.6.15.12           3080187    2514082
lan1      1500 192.8.15.0       192.8.15.12           3372829    1996687
lan1:1    1500 192.8.15.0       192.8.15.101                0         20
bermese[63]% netstat -rn
Routing tables
Dest/Netmask     Gateway          Flags  Refs       Use  Interface  Pmtu
127.0.0.1        127.0.0.1        UH       0   3866265  lo0        4136
192.6.15.12      192.6.15.12      UH       0      7300  lan2       4136
192.8.15.12      192.8.15.12      UH       0         6  lan1       4136
15.13.171.235    15.13.171.235    UH       0   5405618  lan3       4136
192.8.15.101     192.8.15.101     UH       0         0  lan1:1     4136
192.6.15.0       192.6.15.12      U        2         0  lan2       1500
192.8.15.0       192.8.15.101     U        3         0  lan1:1     1500
192.8.15.0       192.8.15.12      U        3         0  lan1       1500
15.13.168.0      15.13.171.235    U        2         0  lan3       1500
default          15.13.171.235    U        0         0  lan3       1500
127.0.0.0        127.0.0.1        U        0         0  lo0        4136

>>
>> Now we recheck the remsh command using exactly the same command
>> line as we did just a moment ago. Note that this fails. The
>> supposition is that the source address used for this was the new IP
>> that we just added (192.8.15.101) since its route entry came first.
>> This IP is not in the rhosts. file on the remote node.
>>

bermese[64]% remsh 192.8.15.11 date
remshd: Login incorrect.

>>
>> Now remove the IP address and verify we get remsh capability back.
>>

bermese[65]% ifconfig lan1:1 inet 0.0.0.0
bermese[66]% remsh 192.8.15.11 date
Wed Feb 18 19:23:07 PST 1998
bermese[67]% exit

>>
>> Clearly there is a change in behavior between 10.2 based systems
>> and 11.00 (although I assume 10.3 based systems behave the same way
>> as 11.00 based systems).
>>


Request
=======
```

This problem happens in big NEC/NTT project called FITS.
Please fix this problem and make patch ASAP.

****

I entered this SR for tracking purpose and describe the problem.
I will open WTEC call for this problem very soon, so please cooperate
with the lab to fix the defect immediately.
When you need to contact me please send message to both
<asano@jpn.hp.com> and <sprc@jpn.hp.com>, since I might be working in
NEC's office.

Masahiro Asano
(asano@jpn.hp.com)

Marketing Text

February 23, 1998 - Scott Millward

I am submitting this to the lab on Akio's behalf.

The lab has been working on a fix and they need this SR. I
moved this SR from Arpa Service to Internet Transport team.

Lab Text

DTS defect #INDaa29894 for project "net.bsd_arpa "
DTS Version Fixed is "b.11.rv "

Problem Text

When an IP address is added as a secondary interface -- e.g.
when a ServiceGuard (SG) package runs -- one of two errors
can result when certain conditions are met:

1. Using an r-command (e.g. rlogin, rcp) or an SG cm-command results in a "Permission denied" error if the secondary IP address is not in the rhosts. file on the remote system. This affects commands and applications that bind to INADDR_ANY.
2. SG loses its heartbeat and detects a spurious "failure" if the secondary IP address is a duplicate of one of the remote system's IP addresses, a configuration error. This happens when the system must send an ARP request to locate the remote system.

Cause Text

The problem is caused by the LIFO order of added routes
internally.

At MR, routes were added in FIFO order. However, this broke SG's heartbeat during switchover because the primary interface route would no longer be first after it is switched over. If switchover took too long (as it might for large numbers of secondary IP addresses), SG would detect a spurious failure.

The fix, in the first post-MR patch, was to add routes in LIFO order, which is generally consistent with 10.20 and earlier. However, a strict LIFO ordering causes routes added for secondary IP addresses to overshadow the primary IP route for the same subnet.

Thus, secondary IP addresses are used for the source IP address in packets sent over a socket bound to INADDR_ANY. And secondary IP addresses are also used as the source IP address in ARP requests. duplicate secondary IP addresses are used in ARP requests.

This is especially problemmatic for SG environments because each HA application might add one or more secondary IP addresses. HA application use these almost as a remote "name" of an application.

Secondary IP addresses are added and removed each time an HA application is started and stopped.
     This makes it impossible for a remote systems to maintain rhosts. files to reflect the dynamic identify of incoming packets from SG systems.
     Note: This is not a problem in 10.20 because, apparently, this LIFO ordering does not apply to secondary IP addresses added with ifalias. (I'm not exactly sure why.)

Fix Text

Fixed in 11.00. Nother fixes needed.