Prelude-LML - Bug #232

ssh.rules does not handle IPv6 address

06/05/2007 08:41 PM - prmarino1-gmail-com -

Status:	Closed	**Start date:**	
Priority:	Normal	**Due date:**	
Assignee:	Yoann VANDOORSELAERE	**% Done:**	0%
Category:		**Estimated time:**	0.00 hour
Target version:	0.9.10.1		
Resolution:	fixed		

Description

attached to this ticket is a diff of the ssh.rules
I added a workaround for buggy output I'v seen on Redhat AS4 where ::ffff: is prepended to ipv4 addresses

```
Jun 5 15:50:35 somehost sshdr17740: Accepted publickey for someuser from ::ffff:192.168.0.22 port
59610 ssh2
```

also I altered rule number 1909 due to the fact it always reported targer user root even though the regex only matches non root users

```
- target(0).user.user_id(0).name=root; \
+ target(0).user.user_id(0).name=$2; \
```

Associated revisions

Revision 08662aae - 07/13/2007 11:05 AM - Yoann VANDOORSELAERE

SSH IPv6 compatibility fixes.

Make all SSH rules IPv6 compliant, allowing to merge old
IPv6 only rules with IPv4 rules. Some additional minor
bug fixes.

Include fix for incorrect target user assignment, as well as
incorrect variable in assessment.impact.description by
Paul Robert Marino <prmarino1@gmail.com>

This fixes #232.

git-svn-id: file:///home/yoann/dev/prelude/git/nok/SVN/prelude-lml/trunk@9683 09c5ec92-17d4-0310-903a-819935f44dba

History

#1 - 06/06/2007 12:03 AM - prmarino1-gmail-com -

I added on more additional patch to be applied after the original patch
i had forgoten to update assessment.impact.description on rule id=1900

#2 - 06/06/2007 10:26 AM - Yoann VANDOORSELAERE

- Status changed from New to Assigned

::ffff:192.168.0.22 is a valid IPv6 mapped IPv4 address. The regular expression should be updated to match IPv6 address by changing it from **([\d\.]+)** to something like **([a-zA-Z\d\.:]+)**.

#3 - 06/06/2007 06:29 PM - prmarino1-gmail-com -

you are correct and there is a later rule which uses a regex already to handles ipv6 the problem is the behavior of ssh is to map all ipv4 addresses as ipv6 in this manner when the user connected using ipv4 it also maps this way if ipv6 is used even if the user who connects uses standard hex notation to connect. this can confus an operator who is not familiar with ipv6.

though this is probably more of an issue to take up with open ssh
maybe we change modify the rule so it reports both ipv4 and ipv6 version of the address in this case.

either way rule 1909 needs to be corrected
it is the existing ipv6 rule for user authentication and the fact that it reports all non root users as root is a major bug

#4 - 06/06/2007 07:09 PM - Yoann VANDOORSELAERE

Replying to [comment:3 prmarino1@gmail.com]:

> you are correct and there is a later rule which uses a regex already to handles ipv6 the problem is the behavior of ssh is to map all ipv4 addresses as ipv6 in this manner when the user connected using ipv4 it also maps this way if ipv6 is used even if the user who connects uses standard hex notation to connect. this can confus an operator who is not familiar with ipv6.

I agree, although another solution would be to use the Prelude-Manager normalization plugin to fix the problem, since other products could exhibit the same behavior as [[OpenSSH]]. Then maybe we should fix Normalize so that it always remap IPv6 mapped IPv4 address to IPv4.

> though this is probably more of an issue to take up with open ssh
> maybe we change modify the rule so it reports both ipv4 and ipv6 version of the address in this case.

In the attached patch, all the rule now use (\S+), allowing to merge IPv4 and IPv6 rules together, and let the Normalization plugin set the *address.category* member.

> either way rule 1909 needs to be corrected
> it is the existing ipv6 rule for user authentication and the fact that it reports all non root users as root is a major bug

This is also ported in the attached file.

#5 - 06/06/2007 10:10 PM - prmarino1-gmail-com -

I like the idea of having the prelude-manager nomalize the addresses because you are right I've looked through my logs and found cfengine also does this too. the only reservation I have is ([A-Fa-f\d:\.]+) should be used instead of (\S+) followed by an identical rule using (\S+) but maps to source(0).node.name because I have seen implementations in the past that use hostnames where avalable instead of addresses this way we can catch the ones that match ip addresses first and any thing that does not match an ip address can be cought by the second rule as a hostname

#6 - 06/06/2007 10:17 PM -

Replying to [comment:5 prmarino1@gmail.com]:

> I like the idea of having the prelude-manager nomalize the addresses because you are right I've looked through my logs and found cfengine also does this too. the only reservation I have is ([A-Fa-f\d:\.]+) should be used instead of (\S+) followed by an identical rule using (\S+) but maps to source(0).node.name because I have seen implementations in the past that use hostnames where avalable instead of addresses this way we can catch the ones that match ip addresses first and any thing that does not match an ip address can be cought by the second rule as a hostname

The normalize plugin for Prelude-Manager should be able to handle this, and handle it in a more consistent manner than a bunch of different rules. The more rules you make for the end-sensors, the more complex the ruleset becomes and the more difficult it is to maintain. If, instead, you let Prelude-Manager handle this, you get a more comprehensive solution that takes into account all of the sensors (including sensors that haven't even been written yet), instead of a single sensor at a time.

#7 - 06/07/2007 12:25 PM - Yoann VANDOORSELAERE

I don't like the idea of having the Normalizer moving entry from *node.address.address* to *node.name*, since it would encourage bad ruleset/alert writing behavior.

Original, IPv4 and IPv6 compliant version:

```
regex=Accepted (\S+) for root from ([A-Fa-f\d:\.]+) port (\d+); \
 source(0).node.address(0).address=$2; \
 source(0).service.port=$3; \
```

Version that can handle IPv4, IPv6, as well as hostname:

```
regex=Accepted (\S+) for root from (([A-Fa-f\d:\.]+)|(\S+)) port (\d+); \
 source(0).node.address(0).address=$3; \
 source(0).node.name = $4; \
 source(0).service.port=$5; \
```

Using multiple regular expression to capture different type of input work, but it make rules difficult to understand. Paradoxically, it make it more clear to the experienced reader what is currently going on by exclusively relying on PCRE.

Another solution would be to create some kind of PCRE macro:

```
regex=Accepted (\S+) for root from $get_name_or_address($host1, $addr1) port (\d+); \
 source(0).node.address(0).address=$addr1; \
 source(0).node.name = $host1; \
 source(0).service.port=$2; \
```

Or:

```
regex=Accepted (\S+) for root from $get_name_or_address(source(0).node.name, source(0).node.address(0).address
) port (\d+); \
 source(0).service.port=$2; \
```

Comments?

#8 - 06/18/2007 09:38 AM - Yoann VANDOORSELAERE

(In r9659) Improve IPv4 / IPv6 address normalization.

IPv4 mapped IPv6 addresses are now mapped back to IPv4.
Additionally, the Normalize plugin now provide two additionals option:

- ipv6-only: Map any incoming IPv4 address to IPv6.
- keep-ipv4-mapped-ipv6: do not map IPv4 mapped IPv6 addresses back to IPv4.

refs #232.

#9 - 07/13/2007 11:05 AM - Yoann VANDOORSELAERE

- Status changed from Assigned to Closed

- Resolution set to fixed

(In r9683) SSH IPv6 compatibility fixes.

Make all SSH rules IPv6 compliant, allowing to merge old
IPv6 only rules with IPv4 rules. Some additional minor
bug fixes.

Include fix for incorrect target user assignment, as well as
incorrect variable in assessment.impact.description by
Paul Robert Marino <prmarino1@gmail.com>

This fixes #232.

#10 - 04/29/2009 12:21 PM - Yoann VANDOORSELAERE

- Project changed from PRELUDE SIEM to Prelude-LML

- Category deleted (4)

- Target version deleted (0.9.10.1)

Files

sshrules.diff	7.37 KB	06/05/2007	prmarino1-gmail-com -
ssh.rules.diff2	531 Bytes	06/05/2007	prmarino1-gmail-com -
ssh-update.diff	12.7 KB	06/06/2007	Yoann VANDOORSELAERE