

Rudder - Bug #9347

sudo management isn't update-safe

2016-10-13 02:02 - Florian Heigl

Status:	Released	
Priority:	N/A	
Assignee:	Benoît PECCATTE	
Category:	Techniques	
Target version:	3.1.23	
Pull Request:	https://github.com/Normation/rudder-techniques/pull/1178	Priority: 65
Severity:	Critical - prevents main use of Rudder no workaround data loss security	Name check:
UX impact:		Fix check:
User visibility:	Getting started - demo first install level 1 Techniques	Regression:
Effort required:	Medium	
Description		
If you extend the list of allowed sudo commands, rudder will add another line with the second permission set. also, it adds it's own entries in the last line, after the #include statement. that's not proper, can you make it so it adds its stuff before #include since that is the last line by convention (no technical need, just style)		
Related issues:		
Related to Rudder - User story #11145: Add bundle in library to edit section,...		Released

Associated revisions

Revision 7ccfd21b - 2017-07-18 18:36 - Nicolas CHARLES

Refs #9347: Creation of sudoParameters version 3.1 from 3.0

Revision 30280bea - 2017-07-18 18:36 - Nicolas CHARLES

Fixes #9347: sudo management isn't update-safe

History

#1 - 2016-11-23 15:09 - Benoît PECCATTE

Yes, this is due to a limitation in how we write techniques and how cfengine convergence works.

To make sure a technique is update-safe, we are thinking at how to solve this but it needs long term changes.

To work around the limitations, the best thing is to use templates.

#2 - 2017-04-03 16:29 - François ARMAND

- Severity set to Critical - prevents main use of Rudder | no workaround | data loss | security

- User visibility set to Getting started - demo | first install | level 1 Techniques

- Priority set to 72

#3 - 2017-04-03 16:30 - François ARMAND

We should have an edit zone for rudder, and cleanly managed everything in that zone.

#4 - 2017-04-25 11:51 - Nicolas CHARLES

- Effort required set to Medium

- Priority changed from 72 to 70

Ok, there are two parts in this:

1. editing within a Rudder zone - strictly enforcing content there (but what about lines not in this zone, should they be moved within the zone when commands are managed ?). Question is: how do we know which Directives will trigger a change if several directives are editing this zone ?
2. extending list of commands: we need to be able that we are extending a line (so partial match?); or strictly enforce the content (see previous point)

Feedback is welcome on the expected behaviour

Estimated effort is more than a day (something in between checkGenericFileContent + ensure_key_value_parameters)

#5 - 2017-05-24 11:02 - Benoît PECCATTE

- Assignee set to Nicolas CHARLES

- Priority changed from 70 to 69

#6 - 2017-06-12 15:21 - Nicolas CHARLES

- Target version set to 3.1.21

We should use file_ensure_block_in_section to create and edit section

#7 - 2017-06-12 15:28 - Nicolas CHARLES

Nicolas CHARLES wrote:

We should use file_ensure_block_in_section to create and edit section

Actually, this would prevent detecting which command has been edited

So we could either create a new generic method that would have name of section as class parameter, or find another solution :/

#8 - 2017-06-13 10:49 - Nicolas CHARLES

To have a correct fix with generic method, we need to have composite keys for reporting, and this is quite a big change (it could only be in master for ncf), even if located only in ncf

So to have a suitable fix in Rudder 3.1, we'll create an ad-hoc code, like a generic method, in sudo technique, to edit section with proper reporting

#9 - 2017-06-15 10:51 - Vincent MEMBRÉ

- Target version changed from 3.1.21 to 3.1.22

- Priority changed from 69 to 68

#10 - 2017-06-16 12:32 - Nicolas CHARLES

- Status changed from New to In progress

#11 - 2017-06-16 17:25 - Nicolas CHARLES

there a huge question mark here.

We are moving from managing file like

```
# User privilege specification
root    ALL=(ALL:ALL) ALL
```

```
%admin  ALL=(ALL) ALL
```

```
%sudo   ALL=(ALL:ALL) ALL
```

to

```
# User privilege specification
root    ALL=(ALL:ALL) ALL
```

```
# Configuration name Name 1
%admin  ALL=(ALL) ALL
# End Configuration name Name 1
```

```
# Configuration name Name 2
%sudo   ALL=(ALL:ALL) ALL
# End Configuration name Name 2
```

without having duplicate lines

an idea would be to create a globally managed section

#Managed by Rudder

#End of section Managed by Rudder

and work in this section, and purge all duplicated line in and out this section; but resulting code gets pretty complex (but it's doable)

#12 - 2017-06-16 17:27 - Nicolas CHARLES

Work in progress here: <https://github.com/ncharles/rudder-techniques/commit/cc6cb63270bdaf37a6ee0bd7c77ccedae081b7b6>

#13 - 2017-06-16 17:27 - Nicolas CHARLES

Work in progress here: <https://github.com/ncharles/rudder-techniques/commit/6235842da742c2d26b3b3a6dc5e1a036b050c050>

#14 - 2017-06-20 10:41 - Nicolas CHARLES

Decided solution:

we check if expected line is there - if it is, but not in the section (and the section doesn't exist somewhere else), we wrap it around the section. Otherwise, if section is there, we edit it otherwise, we add the section

#15 - 2017-07-17 16:57 - Nicolas CHARLES

- Status changed from *In progress* to *Pending technical review*
- Pull Request set to <https://github.com/Normation/rudder-techniques/pull/1176>
- Priority changed from 68 to 67

PR <https://github.com/Normation/rudder-techniques/pull/1176>

#16 - 2017-07-17 17:04 - Nicolas CHARLES

- Status changed from *Pending technical review* to *In progress*
- Pull Request deleted (<https://github.com/Normation/rudder-techniques/pull/1176>)

it's still in progress (i did a rudder-dev wip, don't know why it created a PR)

#17 - 2017-07-18 18:32 - Nicolas CHARLES

- Related to User story #11145: Add bundle in library to edit section, and enforce its content, as well as deleting line matching regexp in all file added

#18 - 2017-07-18 18:35 - Nicolas CHARLES

- Status changed from *In progress* to *Pending technical review*
- Assignee changed from Nicolas CHARLES to Benoît PECCATTE
- Pull Request set to <https://github.com/Normation/rudder-techniques/pull/1178>

PR <https://github.com/Normation/rudder-techniques/pull/1178>

#19 - 2017-08-09 17:13 - Vincent MEMBRÉ

- Target version changed from 3.1.22 to 3.1.23
- Priority changed from 67 to 66

#20 - 2017-08-21 18:36 - Nicolas CHARLES

- Status changed from *Pending technical review* to *Pending release*

Applied in changeset [rudder-techniques|30280beaaa8c59d4173aff027a309a2c9df99bc6](https://github.com/Normation/rudder-techniques/commit/30280beaaa8c59d4173aff027a309a2c9df99bc6).

#21 - 2017-09-07 17:55 - Vincent MEMBRÉ

- Status changed from *Pending release* to *Released*
- Priority changed from 66 to 65

This bug has been fixed in Rudder 3.1.23, 4.1.7 and 4.2.0-rc1 which were released today.

- 3.1.23: [Announce Changelog](#)
- 4.1.7: [Announce Changelog](#)

- 4.2.0~rc1: [Announce Changelog](#)
- Download: <https://www.rudder-project.org/site/get-rudder/downloads/>