Foreman - Bug #14635

CVE-2016-3693 - `inspect` in a provisioning template exposes sensitive controller information

04/14/2016 03:13 AM - Dominic Cleal

Status:	Closed		
Priority:	High		
Assignee:	Ivan Necas		
Category:	Security		
Target version:	1.11.1		
Difficulty:		**Fixed in Releases:**	
Triaged:		**Found in Releases:**	
Bugzilla link:		**Red Hat JIRA:**	
Pull request:	https://github.com/theforeman/foreman/pull/3430		

Description

A provisioning template containing <%= inspect %> will expose sensitive information about the Rails controller and application when rendered when using Safemode rendering (default).

Safemode is initialised with a "delegate" object that is typically the Rails controller. When inspect is called on it, all information about the Rails app is exposed, including routes, secret tokens, caches and so on.

Thanks to Ivan Necas for reporting the security issue to foreman-security@googlegroups.com.

All versions of Foreman are vulnerable, CVE identifier will be requested.

Associated revisions

Revision 82f9b93c - 04/18/2016 10:42 AM - Ivan Necas

Fixes #14635 - bump safemode version to fix the unwanted inspect issue

Revision c4e5d9a2 - 04/18/2016 10:45 AM - Dominic Cleal

refs #14635 - require safemode 1.2.4

Revision 5d399a05 - 04/18/2016 02:44 PM - Ivan Necas

Fixes #14635 - bump safemode version to fix the unwanted inspect issue

(cherry picked from commit 82f9b93c54f72c5814df6bab7fad057eab65b2f2)

History

#1 - 04/14/2016 03:15 AM - Dominic Cleal

I'd suggest the rendering methods shouldn't be mixed directly into controllers and should instead be in a more isolated object, which would limit the amount of data being exposed.

It may be worth trying to get #inspect removed from safemode's default permitted methods due to its ability to expose instance variables.

#2 - 04/14/2016 04:17 AM - Ivan Necas

- Status changed from New to Assigned

- Assignee set to Ivan Necas

#3 - 04/14/2016 04:33 AM - Marek Hulán

> I'd suggest the rendering methods shouldn't be mixed directly into controllers and should instead be in a more isolated object, which would limit the amount of data being exposed.

That would be really awesome, one can get inspiration in remote execution plugin which implements it's own renderer . The only downside is that it would be quite big change for a security fix since it involves both TemplatesController and UnattendedController. So to fix this I'd just disable inspect

globally and as a second PR we could refactor rendering.

#4 - 04/14/2016 04:44 AM - Dominic Cleal

Marek Hulán wrote:

> The only downside is that it would be quite big change for a security fix since it involves both TemplatesController and UnattendedController. So to fix this I'd just disable inspect globally and as a second PR we could refactor rendering.

Yes, I agree. If removing #inspect isn't possible or accepted, then we can just fix this in the next major version with a refactoring.

#5 - 04/14/2016 06:21 AM - Ivan Necas

I looked into possibility to solve this in Foreman, but it's not nice at all: the problem is the inspect is allowed on the Safemode::Blankslate object and there is not easy way to remove it form there: we would need to override the `inspect` method on the objects that are used by safemode, which would affect their behaviour even outside of rendering.

Also, the problem is not just with the Safemode::Scope, but also with the Jail objects, where one can see attributes, that were not allowed in safemode.

Removing the inspect from the allowed methods seems like the best thing we can do right now.

#6 - 04/14/2016 06:38 AM - Ivan Necas

I've opened a PR against safemode to address the issue https://github.com/svenfuchs/safemode/pull/17

#7 - 04/14/2016 09:15 AM - Anonymous

safemode v1.2.4 that includes Ivan's fix was released today.

#8 - 04/14/2016 09:17 AM - The Foreman Bot

- Status changed from Assigned to Ready For Testing

- Pull request https://github.com/theforeman/foreman/pull/3430 added

#9 - 04/15/2016 04:11 AM - Dominic Cleal

- Subject changed from `inspect` in a provisioning template exposes sensitive controller information to CVE-2016-3693 - `inspect` in a provisioning template exposes sensitive controller information

CVE-2016-3693 has been assigned for this issue.

#10 - 04/18/2016 10:45 AM - Dominic Cleal

- translation missing: en.field_release set to 141

#11 - 04/18/2016 11:03 AM - Ivan Necas

- Status changed from Ready For Testing to Closed

- % Done changed from 0 to 100

Applied in changeset 82f9b93c54f72c5814df6bab7fad057eab65b2f2.