Ruby master - Bug #7759

Marshal.load is not documented to be dangerous

01/31/2013 08:59 AM - charliesome (Charlie Somerville)

Status: Closed
Priority: Normal
Assignee:
Target version: 2.0.0
Backport:

ruby -v:
ruby 2.0.0dev (2013-01-07 trunk 38733) [x86_64-darwin12.2.1]

Related issues:
Related to Ruby master - Bug #7780: Marshal & YAML should deserialize only ba...

Associated revisions
Revision 18823475 - 01/31/2013 12:55 PM - charliesome (Charlie Somerville)
marshal.c: warn against using Marshal.load on untrusted data

Revision 38999 - 01/31/2013 12:55 PM - charliesome (Charlie Somerville)
marshal.c: warn against using Marshal.load on untrusted data

Revision 38999 - 01/31/2013 12:55 PM - charliesome (Charlie Somerville)
marshal.c: warn against using Marshal.load on untrusted data

Revision 38999 - 01/31/2013 12:55 PM - charliesome (Charlie Somerville)
marshal.c: warn against using Marshal.load on untrusted data

Revision 38999 - 01/31/2013 12:55 PM - charliesome (Charlie Somerville)
marshal.c: warn against using Marshal.load on untrusted data

Revision 38999 - 01/31/2013 12:55 PM - charliesome (Charlie Somerville)
marshal.c: warn against using Marshal.load on untrusted data

Revision 38999 - 01/31/2013 12:55 PM - charliesome (Charlie Somerville)
marshal.c: warn against using Marshal.load on untrusted data

Revision 38999 - 01/31/2013 12:55 PM - charliesome (Charlie Somerville)
marshal.c: warn against using Marshal.load on untrusted data

Revision 38999 - 01/31/2013 12:55 PM - charliesome (Charlie Somerville)
marshal.c: warn against using Marshal.load on untrusted data

Description

Marshal.load is incredibly powerful, and also incredibly dangerous.

Unfortunately, many developers use it inappropriately and unmarshal user input. This can lead to a wide range of vulnerabilities, including remote code execution.

Marshal.load should be documented as dangerous and the documentation should also mention that it should only be used on trusted data.

Related to Ruby master - Bug #7780: Marshal & YAML should deserialize only ba...
Marshal.load is incredibly powerful, and also incredibly dangerous.

Unfortunately, many developers use it inappropriately and unmarshal user input. This can lead to a wide range of vulnerabilities, including remote code execution.

Marshal.load should be documented as dangerous and the documentation should also mention that it should only be used on trusted data.

Makes sense. Can you please consider to write down the explanation?

Unfortunately, many developers use it inappropriately and unmarshal user input. This can lead to a wide range of vulnerabilities, including remote code execution.

Can't you elaborate it, probably, at security@ruby-lang.org?

I've thought it's a common sense, isn't it?

You would imagine so, however I have seen a lot of code that does unmarshal untrusted data.

I will send an example to security@ruby-lang.org. Please note that I do not consider this a vulnerability in Ruby. Marshal is dangerous by design. This is an education problem - we need to document the fact that it is dangerous.
Unfortunately, many developers use it inappropriately and unmarshal user input. This can lead to a wide range of vulnerabilities, including remote code execution.

Could you explain attack scenario?

--
// SASADA Koichi at atdot dot net

#5 - 01/31/2013 09:55 PM - charliesome (Charlie Somerville)
- Status changed from Open to Closed
- % Done changed from 0 to 100

This issue was solved with changeset r38999.
Charlie, thank you for reporting this issue.
Your contribution to Ruby is greatly appreciated.
May Ruby be with you.

marshal.c: warn against using Marshal.load on untrusted data

- marshal.c (marshal_load): Add documentation warning against using Marshal.load on untrusted data [Bug #7759] [ruby-core:51765]