Ruby master - Feature #10793

Infrastructure/Release-Management: Sign releases

01/28/2015 12:07 PM - rmoriz (Roland Moriz)

Status:	Open	
Priority:	Normal	
Assignee:		
Target version:		

Description

Hi,

currently Ruby releases are not cryptographically signed and distributed unencrypted via http. While there are some MD5-hashes on the web-site, it's cumbersome to automate and MD5 is already insecure.
This is a huge security risk because currently it just takes a simple HTTP MITM attack to inject a backdoored ruby to downstream projects and end users, like e.g. the official Docker image (see https://github.com/docker-library/ruby/blob/master/2.2/Dockerfile#L12).

Please sign the release files with a release/maintainer pgp/gpg key.

Other OSS projects already sign their releases, e.g.:

- PHP http://php.net/downloads.php
- Python https://www.python.org/downloads/release/python-278/

Thank you.

History

#1 - 01/29/2015 06:31 AM - naruse (Yui NARUSE)

As far as I remember we discussed this topic before (but I can't find the ticket/mail).

Anyway the conclusion is hash digests for tarballs should be available through https.
If people can get hash digest through a trusted way, people can trust the tarball.
(though MD5 is not suitable as you say)

A release announce has such hash digests through https.
You can use this https://www.ruby-lang.org/en/news/2014/12/25/ruby-2-2-0-released/

#2 - 02/11/2016 09:51 AM - aef (Alexander E. Fischer)

Yui NARUSE wrote:

> As far as I remember we discussed this topic before (but I can't find the ticket/mail).
>
> Anyway the conclusion is hash digests for tarballs should be available through https.
> If people can get hash digest through a trusted way, people can trust the tarball.
> (though MD5 is not suitable as you say)
>
> A release announce has such hash digests through https.
> You can use this https://www.ruby-lang.org/en/news/2014/12/25/ruby-2-2-0-released/

Several commonly used TLS libraries such as OpenSSL and GnuTLS are plagued by security vulnerabilities, some parts of the TLS standard have been backdoored by government agencies in the past and to be able use TLS you are expected to setup hundreds of highly non-transparent X.509 certificate authorities located in all kinds of jurisdictions all over the world. Even worse, the aforementioned CAs earn their livings by delegating your "trust" to mostly anyone that can afford it monetarily by providing intermediate certificate authorities.

HTTPS relies solely upon TLS for its security features. Therefore I argue that the security provided by the current HTTPS setup is very low.

Please reconsider to provide OpenPGP signatures (for example through GnuPG) for the release announcements including the release artifact checksums and/or the release artifacts (e.g. source code archives) themselves.

I am willing to provide support implementing this.

#3 - 02/12/2016 09:20 PM - shyouhei (Shyouhei Urabe)

I'm not against the idea of additionaly signing the releases but,

Alexander E. Fischer wrote:

> Several commonly used TLS libraries such as OpenSSL and GnuTLS are plagued by security vulnerabilities

Then how can you say GnuPG is safe instead? Where is the difference?

You are saying "SSL is insecure in general" and that is not a common idea I guess.

When HTTPS is in threat a system admin can and should fix their web server (maybe by upgrading the vulunerable SSL library, or by re-issueing the used certificate). Isn't this enough for securly downloading ruby? If you cannot trust our system admins will properly hande this situation and think they are malicious, then how on earth can you trust our products themselves? They can issue canonical releases at will. Or shouldn't they? Then should who?