I got this result:

```bash
# rake db:encrypt RAILS_ENV=production --trace
(in /www/sites/duckcorp/projects.duckcorp.org/redmine-1.2.0)
```

```
NOTE: SourceIndex.new(hash) is deprecated; From
/www/sites/duckcorp/projects.duckcorp.org/redmine-1.2.0/config/../vendor/rails/railties/lib/rails/vendor_gem_source_index.rb:100:in `new'.
0:in `new'.
** Invoke db:encrypt (first_time)
** Invoke environment (first_time)
** Execute environment
** Execute db:encrypt
rake aborted!
data must not be empty
```

```
/www/sites/duckcorp/projects.duckcorp.org/redmine-1.2.0/lib/redmine/ciphering.rb:34:in `update'
/www/sites/duckcorp/projects.duckcorp.org/redmine-1.2.0/lib/redmine/ciphering.rb:34:in `encrypt_text'
/www/sites/duckcorp/projects.duckcorp.org/redmine-1.2.0/lib/redmine/ciphering.rb:92:in `write_ciphered_attribute'
/www/sites/duckcorp/projects.duckcorp.org/redmine-1.2.0/app/models/repository.rb:58:in `password='
/www/sites/duckcorp/projects.duckcorp.org/redmine-1.2.0/lib/redmine/ciphering.rb:66:in `send'
/www/sites/duckcorp/projects.duckcorp.org/redmine-1.2.0/lib/redmine/ciphering.rb:66:in `encrypt_all'
/www/sites/duckcorp/projects.duckcorp.org/redmine-1.2.0/lib/redmine/ciphering.rb:64:in `each'
/www/sites/duckcorp/projects.duckcorp.org/redmine-1.2.0/lib/redmine/ciphering.rb:64:in `encrypt_all'
```

```
/ract/database_statements.rb:136:in `transaction'
/www/sites/duckcorp/projects.duckcorp.org/redmine-1.2.0/vendor/rails/activerecord/lib/active_record/connection_adapters/abstract/database_statements.rb:136:in `transaction'
```

```
/usr/lib/ruby/1.8/rake.rb:63:in `call'
/usr/lib/ruby/1.8/rake.rb:63:in `execute'
/usr/lib/ruby/1.8/rake.rb:63:in `each'
/usr/lib/ruby/1.8/rake.rb:63:in `top_level'
/usr/lib/ruby/1.8/rake.rb:2068:in `standard_exception_handling'
```

```
/ract/database_statements.rb:136:in `transaction'
/www/sites/duckcorp/projects.duckcorp.org/redmine-1.2.0/vendor/rails/activerecord/lib/active_record/transactions.rb:182:in `transaction'
```

```
/us/lib/ruby/1.8/rake.rb:63:in `call'
/us/lib/ruby/1.8/rake.rb:63:in `execute'
/us/lib/ruby/1.8/rake.rb:63:in `each'
/us/lib/ruby/1.8/rake.rb:63:in `execute'
/us/lib/ruby/1.8/rake.rb:597:in `invoke_with_call_chain'
/us/lib/ruby/1.8/monitor.rb:242:in `synchronize'
/us/lib/ruby/1.8/rake.rb:590:in `invoke_with_call_chain'
/us/lib/ruby/1.8/rake.rb:583:in `invoke'
/us/lib/ruby/1.8/rake.rb:2051:in `invoke_task'
/us/lib/ruby/1.8/rake.rb:2029:in `top_level'
/us/lib/ruby/1.8/rake.rb:2029:in `each'
/us/lib/ruby/1.8/rake.rb:2029:in `top_level'
/us/lib/ruby/1.8/rake.rb:2068:in `standard_exception_handling'
/us/lib/ruby/1.8/rake.rb:2023:in `top_level'
/us/lib/ruby/1.8/rake.rb:2001:in `run'
```
In my database, the 'login' and 'passwords' of all 'repositories' are empty strings (not NULL, but not sure it makes a difference). Trying to encrypt such an empty string later fails. The one-liner attached patch fixes this issue.

**Associated revisions**

Revision 7985 - 2011-11-29 20:47 - Jean-Philippe Lang

Fixed: crypting blank passwords raises an error (#9137).

**History**

#1 - 2011-09-17 13:40 - Etienne Massip
- Target version set to Candidate for next minor release

#2 - 2011-11-29 20:48 - Jean-Philippe Lang
- Subject changed from db:encrypt fails to handle repositories without auth to db:encrypt fails to handle repositories with blank password
- Status changed from New to Resolved
- Target version changed from Candidate for next minor release to 1.2.3
- Resolution set to Fixed

Fixed in r7985.

#3 - 2011-11-29 22:12 - Jean-Philippe Lang
- Status changed from Resolved to Closed

Merged in r7997.

**Files**

db_encrypt_failure.patch  481 Bytes  2011-08-27  Marc Dequènes