2段階認証

@vividmuimui

2017/08 社内LT資料

話すこと

- 2段階認証の概要
- 仕様決める段階で考えること
- rubyで実装する時

2段階認証とは

雑にいうと、
ID/PWなどでログインした後に、それ以外の情報で認証をすること

用語

そもそも、

- 2**要素**認証(多要素認証)
- 2**段階**認証(多段階認証)

は別
利用者から見たときはどっちでもいい話(よくないが)なので、「**2段階認証**」と書かれることが多い(気がする)
ので、ここでは「**2段階認証**」と呼ぶことにします

2段階認証の要素

- 記憶情報
  - SYK: Something You Know
  - ユーザーしか知らない情報
    - ID/PW/誕生日/秘密の質問など
- 所持情報
  - SYH: Something You Have
  - ユーザーしか持っていない情報
    - 携帯電話・ICカード・トークンデバイスなどを利用したもの
- 生体情報
  - SYA: Something You Are
  - ユーザーの生体情報
    - 指紋・虹彩など

これらの要素があり、2つ(以上)の要素を組み合わせて認証するので、2要素認証(多要素認証)という

2段階目の認証としてよくあるのが(僕目線)

- SMSでワンタイムパスワードを送る
- 「Google認証システム」アプリなどをつかった時間ベースのワンタイムパスワード
  - 30秒で切り替わるやつ
- 専用アプリにpush通知->承認ボタンを押す
  - 呼び方よくわからない。googleは「Google からのメッセージ」
  - push通知でやってるのかもよく知らない
  - twitterとか
  - 最近この方式がよく使われるようになっている印象
- yubikeyなどのデバイストークン(?)
- 専用IDやパスワードを郵送
  - お役所系のサイトに多い印象

仕様決める段階で考えること

- どの方式を選択するか？いくつ用意するか？
- リカバリーコードの方針
- 2段階認証はいつだすのか？
- 2段階認証が突破できない時どうするのか？

どの方式を選択するか？いくつ用意するか？ 1️⃣

とりあえずやるのなら

- 「Google認証システム」アプリなどをつかった時間ベースのワンタイムパスワード

一択だと思う

- 十分一般的
- 実装が楽。ウェブエンジニアの作業だけで済む

どの方式を選択するか？いくつ用意するか？2️⃣

電話番号を扱うノウハウが有るのなら

- SMSでワンタイムパスワード

は選択肢の1つ
電話番号をもっていると、問い合わせなどのほかの用途にも使えて便利そう。(端末やメールに比べて、電話番号は変わりにくいと思うので、何かと役立つ)

これ単体ではなく、ほかの方式と併用するのが良い
また、ここらへんは読んでおく必要がある

- SMSを使った二要素認証を非推奨～禁止へ、米国立技術規格研究所NISTの新ガイダンス案
- SMSによる2要素認証は本当に非推奨なのか？

どの方式を選択するか？いくつ用意するか？3️⃣

何かしらの専用スマホアプリがあるのなら

- 専用アプリにpush通知->承認ボタンを押す

がオススメ
数字を入力するなどの必要が無いので、ユーザーの手間が少ない
ナウい😎

これは単体では使えず、ほかの方式と併用する必要がある
(おそらく。そのアプリに初めてログインするときの2段階認証は？となるので)

Google

(2017/08/02 現時点の話)
googleは、2段階認証を初めて設定する際は、
電話番号(SMS or 音声)を利用した方式を設定する必要がある

それ以降は、
「Google認証システム」や「Google からのメッセージ」などいくつかの種類の方式から、
好きなものを設定できる

リカバリーコードの方針

これら検討する必要がある

- 使い捨てか
  - 使い捨てしか見たこと無い
- いくつ用意するのか？
  - 1 or 10あたり

などなど検討することはあるが、重要なのは↓

- ユーザーが現在有効なリカバリーコードを確認出来る必要があるか？
  - リカバリーコードをDBに保存する際に、ハッシュ化するのか暗号化するのか
  - ログインのパスワードと重要度は変わらないし、ハッシュ化するのが望ましそう(要件によるが)
  - ハッシュ化の場合は、リカバリーコードを確認したいようなケースの際は常に再発行することになる

2段階認証はいつ出すのか？

基本方針は、「ユーザーを認証するタイミングで出す」
なので、通常のログイン以外にも、パスワード再発行とき等にも出す必要はありそう

- ワンタイムパスワードをメールで送信
- 入力画面で、ID/ワンタイムパスワードを入力
- 2段階認証
- 新しいパスワード設定画面

という流れ
だが、実装によるし、本当に出す必要があるか、は議論の余地がある 😕

- ユーザーが「受け取ったメール」、は本当に本人しか受け取れないものか？**所持情報**？
- そもそも、メール以外で本人しか受け取れないルートはないか？
  - 専用アプリの2段階認証ができてるのであれば、その専用アプリにワンタイムパスワード送るとかでも良さそう
    - (たぶん)
  - Google認証システムでの2段階認証が登録されてれば、それをワンタイムパスワードとして「入力画面で、ID/ワンタイムパスワードを入力」に入力してもらう
- ほかの方法でユーザーを認証できないか
  - 秘密の質問・一つ前のパスワード・アカウント年月・電話番号などいくつかの質問を重ねていって本人と確認する
    - Googleがこの方式(詳しくは見てない)

とかとか

素直に「ワンタイムパスワードをメールで送信」で実装するのなら、安全に倒して2段階認証出しておくのが良いと思う
ユーザーの手間が増えるが、パスワードの再発行とかは頻繁に行われないので問題ない

2段階認証が突破できない時どうするのか？

- 「信用できる端末(この端末からは2段階認証をこれ以降要求しない)」の設定がされた端末があればそれでログインしてもらう
- SMSを送れるのであればそれ経由でワンタイムパスワード
- 2段階認証のいずれでもダメ、リカバリーコードも控えてない😱
  - ユーザーサポートに問い合わせをしてもらうしか無い。
  - 本人確認をし、新しいリカバリーコードを渡す、新しい携帯にワンタイムパスワードを送るなどする
  - ⚠️ 2段階認証の設定を強制的にOFFにしてから、ユーザーにログインさせるのは良くない
    - アカウント乗っ取りの可能性が増える

rubyで実装する時 💎

「Google認証システムなどをつかった時間ベースのワンタイムパスワード」この方式を実装する時の話

- deviseを使っているのなら
  - devise-two-factorをつかうのが手っ取り早くて良さそう
  - 試してない
- devise以外なら
  - rotpをつかうのが良い
  - (+ QRコードを表示するためのgem)

Gemの選択基準/実装時に気をつけること

- ユーザーごとにsecret keyを生成・保存する必要があるが、適切に保存されているか？
  - 暗号化されておくのが望ましい
- 一度認証を通したコード(ワンタイムパスワード)は、使えないようにできるか？
  - 対策しないと、その30秒の間は何度も同じコードで認証できる
  - => 悪意あるユーザーが認証を突破できてしまう可能性が増える
  - rotpでは#verify_with_drift_and_priorで対策できる。
    - READMEにやり方書いてある

おわり

「Google認証システムなどをつかった時間ベースのワンタイムパスワード」で実装するのは、
要件・仕様さえ決まればサクッとできる

おまけ

reCAPTCHAの組み込みはもっと楽だった

おまけ 2️⃣

「2段階認証(2要素認証)」を英語にすると、
「two factor authentication」だが、

- 実装する上で two_factor_authentication は
  - 長い
  - typoしやすい
- 2faと略そうとしても、数字から始まるメソッド名はつけれない(ruby)
- otp(one time password)と略すのはちょっと違う
  - otpはotpなので

良い落とし所が見つからない。。
😢