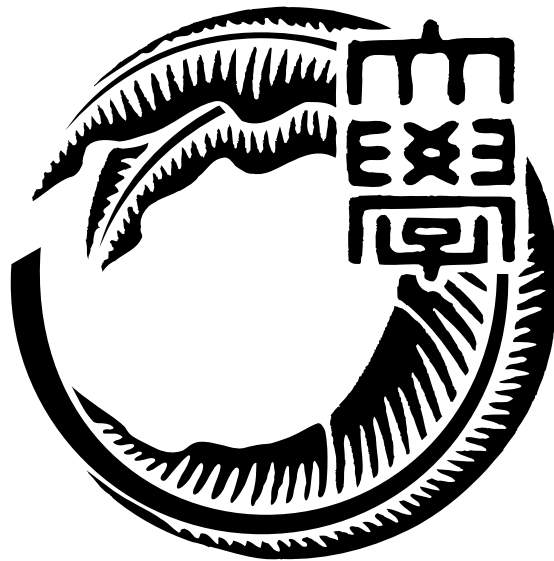


令和元年度 卒業論文

公開暗号方式を用いたSSH認証によるアカウント作成の簡略化

Simplify Creating account by SSH
authentication using public-key cryptography



琉球大学工学部情報工学科

165714D 与那嶺東

指導教員 長田智和, 谷口祐治

目 次

第 1 章	はじめに	1
1.1	背景と目的	1
1.2	論文の構成	1
第 2 章	基礎概念	2
2.1		2
2.2		2
第 3 章	提案手法	3
3.1	ボツ案	3
3.1.1	ログイン状態のセッション橋渡し	3
第 4 章	実験	7
4.1	実験説明	7
4.2		7
4.3	検証結果	7
4.4	考察	7
第 5 章	今後の課題	8

目 次

3.1 ログイン状態のセッション橋渡し案	3
3.2 ログイン成功した際の,HTTP レスポンスの抜粋	4
3.3 ブラウザの Cookie のセッション情報	4
3.4 2つのブラウザ (ログイン状態, ログインしていない状態)	5
3.5 2つのブラウザ (ログイン状態, ログイン状態)	5
3.6 ログイン状態のセッション橋渡し案がボツになった	6

表 目 次

第1章 はじめに

1.1 背景と目的

1.2 論文の構成

第2章 基礎概念

2.1

2.2

第3章 提案手法

3.1 ボツ案

3.1.1 ログイン状態のセッション橋渡し

実装したいこと

公開鍵暗号方式による ssh 認証を成功した際に，WEB サービスでログイン状態になる．ここでいう，ログイン状態は，セッションを用いて，ステートフルな通信をする (HTTP 通信はステートレスな通信)．

実装するための手段案

ログイン状態のセッション情報を橋渡しすることにより，実現しようと考案した．以下の図 3.1 を用いながら説明する．

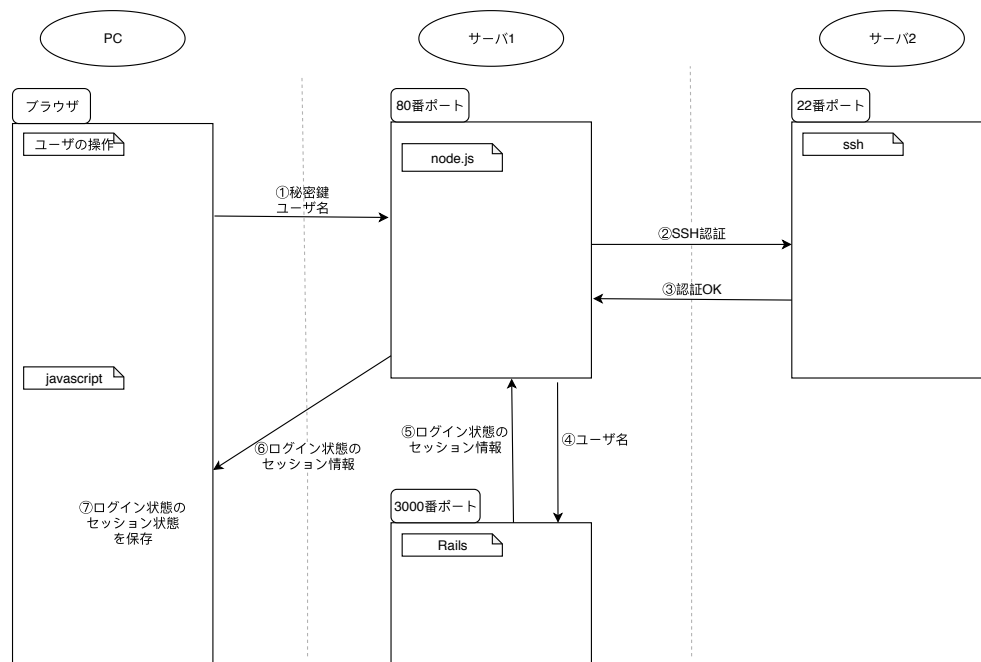


図 3.1: ログイン状態のセッション橋渡し案

図 3.1 の① ~ ③では,エンドユーザはブラウザを用いて,公開鍵暗号方式による ssh をしている。図 3.1 の④では,HTTP リクエストの POST をしている。図 3.1 の⑤では,HTTP レスポンスが来て,そのヘッダ情報に「セッション情報を Cookie に保存する」という情報が付与されている(のちのち,アクセス制限で,localhost からしかアクセスできないようにする)。図 3.1 の⑥では,図 3.1 の⑤の情報を,ブラウザの javascript 側に socket 通信で渡す。最後に,図 3.1 の⑦で,ログイン状態のセッション情報を Cookie に保存する。

ブラウザを用いての検証

ログイン中のセッション情報を,ブラウザの Cookie に保存することで,ログインすることが可能かを検証する。

検証環境

機器

MacBook Pro (Retina, 13-inch, Early 2015)
macOS High Sierra(バージョン 10.13.6)

サーバ側

Ruby on Rails(6.0.2.1)
localhost:3000

ユーザ側

2つのブラウザ

Google Chrome
Google Chrome Canary

Crome の開発環境 (デベロッパーツール) を用いて以下の検証を行った。

まず,ログインした際の動きとして,HTTP レスポンスのヘッダ情報に,set-cookie がある。Crome のデベロッパーツールでは,以下の図 3.2 と表示される。

```
Set-Cookie: _yes_password_session=TfIF9FG0jreVMXhw%2F366KAeIMzjdpBCBswAGpn12rrInXsgipqR0R4xMCF2R2tz2n%2FTqMvgAo568BLMYfc9Z2YpJFB%2BcmpXhAqPd6sUuTtmp5Nbn4FOykeF3itx4pw%2BKRpWxU8g10llyrGCXomkz%2F2FuCLK7yRL5chp469ocKPJWwMpdf45ktWY6ZCsSks0gJHN11uhUKpVLC%2F5R3hT0nUd9hTab%2BqZy4XJMt5i1JEWdgsLj%2FD2Kx1pAtvn0khG3p1AifUToLa6HRMgRfwm0g2P4nG0QadTz%2BcrveQ9tT7xVqgWGESxi%2BsRo--yUt1PA8ewS%2BwMmd6--0j%2BA%2B45WS1VBYaRwRS%2F5nw%3D%3D; path=/; HttpOnly
```

図 3.2: ログイン成功した際の,HTTP レスポンスの抜粋

次に,図 3.2 の HTTP レスポンスを元に,ブラウザの Cookie にセッション情報を保存する。Crome のデベロッパーツールでは,以下の図 3.3 と表示される。

上記の 3.2,3.3 は,HTTP レスポンスのヘッダ情報で,ブラウザの Cookie に値を保存して

Name	Value	Domain	Path	Expires / Max-Age	Size	HttpOnly	Secure	SameSite
_yes_password_session	TfIF9FG0jreVMXhw%2F366KAeIMzjdpBCBswAGpn12rrInXsgipqR0R4xMCF2R2tz2n%2FTqMvgAo568BLMYfc9Z2YpJFB%2BcmpXhAqPd6sUuTtmp5Nbn4FOykeF3itx4pw%2BKRpWxU8g10llyrGCXomkz%2F2FuCLK7yRL5chp469ocKPJWwMpdf45ktWY6ZCsSks0gJHN11uhUKpVLC%2F5R3hT0nUd9hTab%2BqZy4XJMt5i1JEWdgsLj%2FD2Kx1pAtvn0khG3p1AifUToLa6HRMgRfwm0g2P4nG0QadTz%2BcrveQ9tT7xVqgWGESxi%2BsRo--yUt1PA8ewS%2BwMmd6--0j%2BA%2B45WS1VBYaRwRS%2F5nw%3D%3D	localhost	/	Session	415	✓		

図 3.3: ブラウザの Cookie のセッション情報

いる。その後,HTTP リクエストで,Cookie 情報を付与 [1] することにより,ステートレスな

プロトコルである HTTP 上で，状態管理ができる [2](ログイン状態の維持)。
現状で以下の図 3.4 のように，ログインしているブラウザ，ログインしていないブラウザがある。

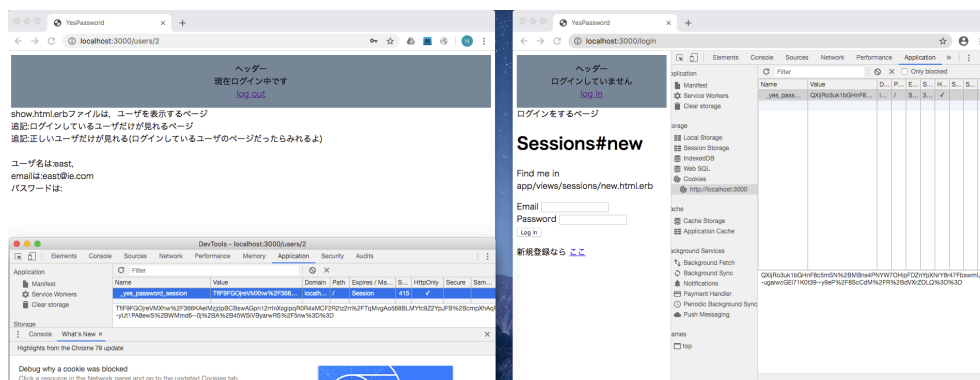


図 3.4: 2つのブラウザ (ログイン状態, ログインしていない状態)

上記の図 3.4 の状態から次の操作を行う。デベロッパーツールを用いて，ログインしているブラウザから，ログインしていないブラウザに，Cookie 情報の、コピーアンドペーストを行い、リロードする。その際の，状態が以下の図 3.5 になり，ログイン状態のセッションを橋渡し (Cookie に保存) することにより，ログインすることができるとわかる。

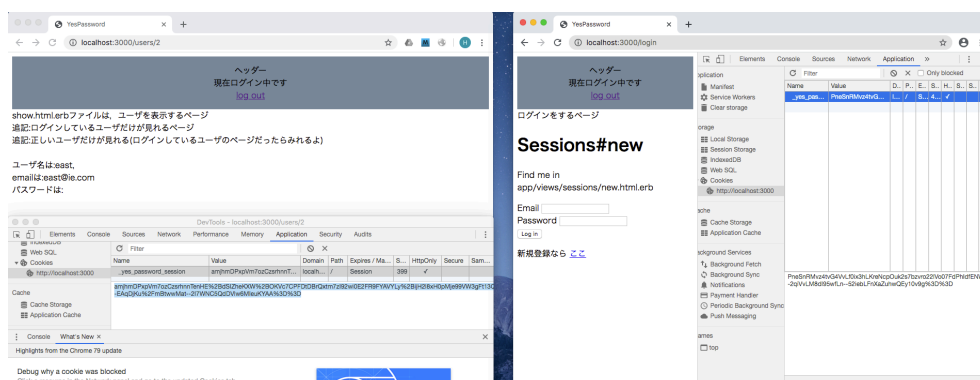


図 3.5: 2つのブラウザ (ログイン状態, ログイン状態)

図 3.5 の補足。

HTTP リクエストをする際に，新しい Cookie 情報が付与されることが確認できた。そのことにより，ログイン中の 2つのブラウザが別のセッションになっている。

実際に実装

実際に実装を進めて，図 3.1 の① ~ ⑥までできた。しかしながら，図 3.1 の⑦ができなかった。その理由を以下の図 3.6 を元に説明する。

図 3.6 の⑤ で,rails ではセッションを発行する際に,http-only 属性を付与している [3]。その

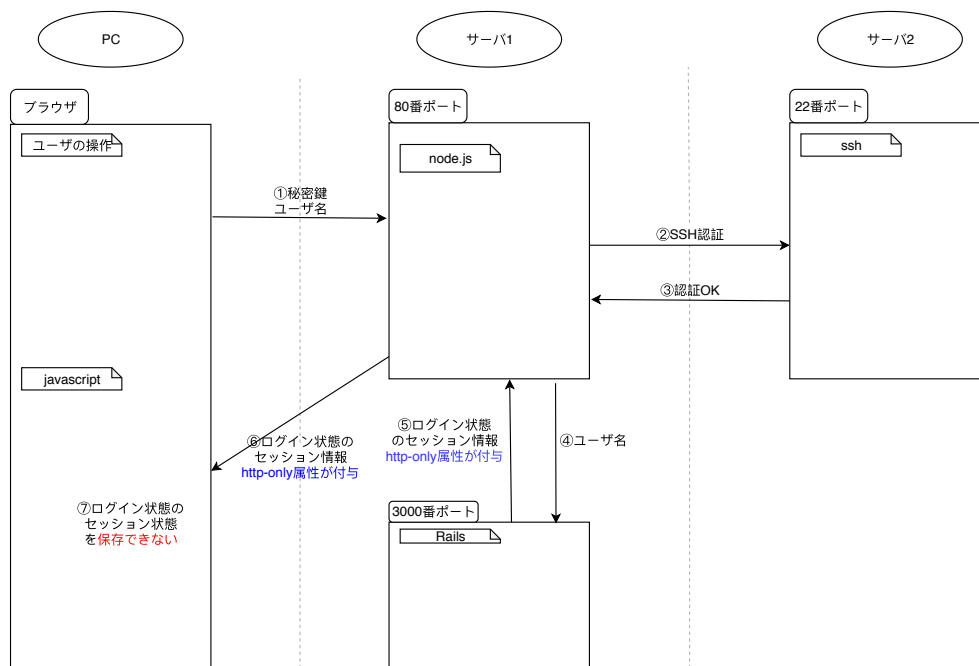


図 3.6: ログイン状態のセッション橋渡し案がボツになった

ことは、ログイン時の HTTP レスポンスである図 3.2 でも現れている。http-only 属性がついた Cookie は、javascript で扱うことができなくなり、セッションハイジャックの対策を行っている [4]。そのため、図 3.6 の⑦のように、セッション状態を保存することができなかったため、「ログイン状態のセッション橋渡し」の案はボツになった。

第4章 実験

4.1 実験説明

4.2

4.3 検証結果

4.4 考察

第5章 今後の課題

参考文献

- [1] <https://tools.ietf.org/html/rfc6265#section-1>
最終閲覧日:2020/1/20
- [2] <https://qiita.com/mogulla3/items/189c99c87a0fc827520e>
最終閲覧日:2020/1/20
- [3] <https://qiita.com/yasu/items/8ae3077bdb6e606681f6#cookiestore%E3%81%8C%E5%95%8F%E9%A1%8C%E3%81%AA%E3%81%AE%E3%81%8B>
最終閲覧日:2020/1/21
- [4] <https://developer.mozilla.org/ja/docs/Web/HTTP/Cookies>
最終閲覧日:2020/1/21

謝辞

本研究の遂行，また本論文の作成にあたり、御多忙にも関わらず終始懇切なる御指導と御教授を賜りました hoge 助教授に深く感謝いたします。

また、本研究の遂行及び本論文の作成にあたり、日頃より終始懇切なる御教授と御指導を賜りました hoge 教授に心より深く感謝致します。

数々の貴重な御助言と細かな御配慮を戴いた hoge 研究室の hoge 氏に深く感謝致します。

また一年間共に研究を行い、暖かな気遣いと励ましをもって支えてくれた hoge 研究室の hoge 君、hoge 君、hoge さん並びに hoge 研究室の hoge、hoge 君、hoge 君、hoge 君、hoge 君に感謝致します。

最後に、有意義な時間を共に過ごした情報工学科の学友、並びに物心両面で支えてくれた両親に深く感謝致します。

2010 年 3 月

hoge