동일출처정책과 CORS에러

SI2팀 엄예지

“ *Same-origin 정책과 CORS이슈 해결 방법이 무엇인가요?* ”

Same-origin 정책과 CORS이슈 해결 방법이 무엇인가요?

동일 출처 정책 (Same-Origin Policy)

- 어떤 출처에서 불러온 문서나 스크립트가 다른 출처에서 가져온 리소스와 상호작용하는 것을 제한하는 중요한 보안 방식
- 동일 출처 정책은 잠재적으로 해로울 수 있는 문서를 분리해서 공격받을수 있는 경로를 줄임
- 웹 브라우저 보안을 위해 프로토콜, 호스트, 포트가 동일한 서버로만 ajax요청을 주고 받을수 있도록 한 정책

Same-origin 정책과 CORS이슈 해결 방법이 무엇인가요?

동일 출처 정책 (Same-Origin Policy)

- 프로토콜, 호스트, 포트는 개발자 도구를 열고 location을 입력하면 정보를 확인할 수 있다.

- http://www.same-domain.com/ → http://same-domain.com
Same-origin

- http://www.same-domain.com → http://www.cross-domain.com
Cross-origin

(여기서 origin이란 요청을 보낸 도메인을 의미)

Same-origin 정책과 CORS이슈 해결 방법이 무엇인가요?

동일 출처 정책 (Same-Origin Policy)

XMLHttpRequest cannot load 'http://localhost:3000'. No 'Access-Control-Allow-Origin' header is present on the requested resource. Origin 'http://localhost:8080' is therefore not allowed access.

- 클라이언트를 3000포트, 서버를 8080포트에서 실행한 후에 클라이언트 서버로 AJAX요청을 보냈을시 .....
- 같은 도메인으로 요청을 보낸 것이 아니기 때문에 동일 출처 정책에 의해 에러가 나는것

Same-origin 정책과 CORS이슈 해결 방법이 무엇인가요?

Cross-Oring Resource Sharing(CORS)

- 추가 HTTP헤더를 사용하여 브라우저가 한 출처에서 실행중인 웹 애플리케이션에 선택된 엑세스 권한을 부여하도록 하는 메커니즘
- 웹 응용 프로그램은 자체와 다른 출처(도메인, 프로토콜, 포트)를 가진 리소스를 요청할때 cross-origin HTTP 요청을 실행
- 출처가 다른 도메인에서의 AJAX요청이라도 서버 단에서 데이터 접근 권한을 허용하는 정책

Same-origin 정책과 CORS이슈 해결 방법이 무엇인가요?

Cross-Oring Resource Sharing(CORS)

< 서버 단 해결 방법 >

- 발생 원인 : 클라이언트와 서버의 도메인이 달랐을때 보안상의 이유로 응답을 받지 못하도록 막은 것
- 해결 방법 : 서버 단에서 특정 도메인 혹은 모든 도메인을 허용하도록 설정!

Same-origin 정책과 CORS이슈 해결 방법이 무엇인가요?

Cross-Oring Resource Sharing(CORS)

1. Access-Control-Allow-Origin response header

- ACAO 응답 헤더는 이 응답이 주어진 origin으로부터의 요청 코드와 공유될 수 있는지를 나타냄
- 서버측 응답에서 접근권한을 주는 헤더를 추가해 cors에러를 해결하는 방법

```
app.use((req, res, next) => {
    res.header("Access-Control-Allow-Origin", "*"); // 모든 도메인
    res.header("Access-Control-Allow-Origin", "https://example.com"); // 특정 도메인
});
```

Same-origin 정책과 CORS이슈 해결 방법이 무엇인가요?

Cross-Oring Resource Sharing(CORS)

- 특정 메소드를 추가하는것도 가능

```
Access-Control-Allow-Method
Access-Control-Max-Age
Access-Control-Allow-Headers
```

- 접근 권한을 허용하는 모든 요청의 응답에 추가해주어야 하므로 만약 모든 요청에 권한 허용을 한다면 모든 응답 전에 헤더를 추가해야 한다

Same-origin 정책과 CORS이슈 해결 방법이 무엇인가요?

Cross-Oring Resource Sharing(CORS)

2. Node.js Express 미들웨어 CORS

- node.js에 미들웨어를 설치하고 App.js에 미들웨어를 꽂으며 세팅 완료

```
npm i cors --save
```

```
const express = require('express');
const cors = require('cors');
const app = express();

app.use(cors());
```

- 아무런 옵션 없이 app.use(cors());로 설정한다면 모든 cross-origin 요청에 대해 응답을 해줌
- 특정 요청에만 응답하는 옵션도 설정 가능

Same-origin 정책과 CORS이슈 해결 방법이 무엇인가요?

Cross-Oring Resource Sharing(CORS)

특정 도메인 접근 허용

```
const corsOptions = {
  origin: 'http://example.com', // 접근 권한을 부여하는 도메인
  credentials: true, // 응답 헤더에 Access-Control-Allow-Credentials 추가해줌
  optionsSuccessStatus: 200 // 응답 상태 200으로 설정
};

app.use(cors(corsOptions));
```

Same-origin 정책과 CORS이슈 해결 방법이 무엇인가요?

Cross-Oring Resource Sharing(CORS)

특정 요청 접근 허용

```
app.get('/products/:id', cors(), function (req, res, next) {
  res.json({msg: 'This is CORS-enabled for a Single Route'})
});
```

- 이외에도 Access-Control-* 헤더 설정 방식처럼 특정 method등을 옵션으로도 설정할 수 있다.

Same-origin 정책과 CORS이슈 해결 방법이 무엇인가요?

Cross-Oring Resource Sharing(CORS)

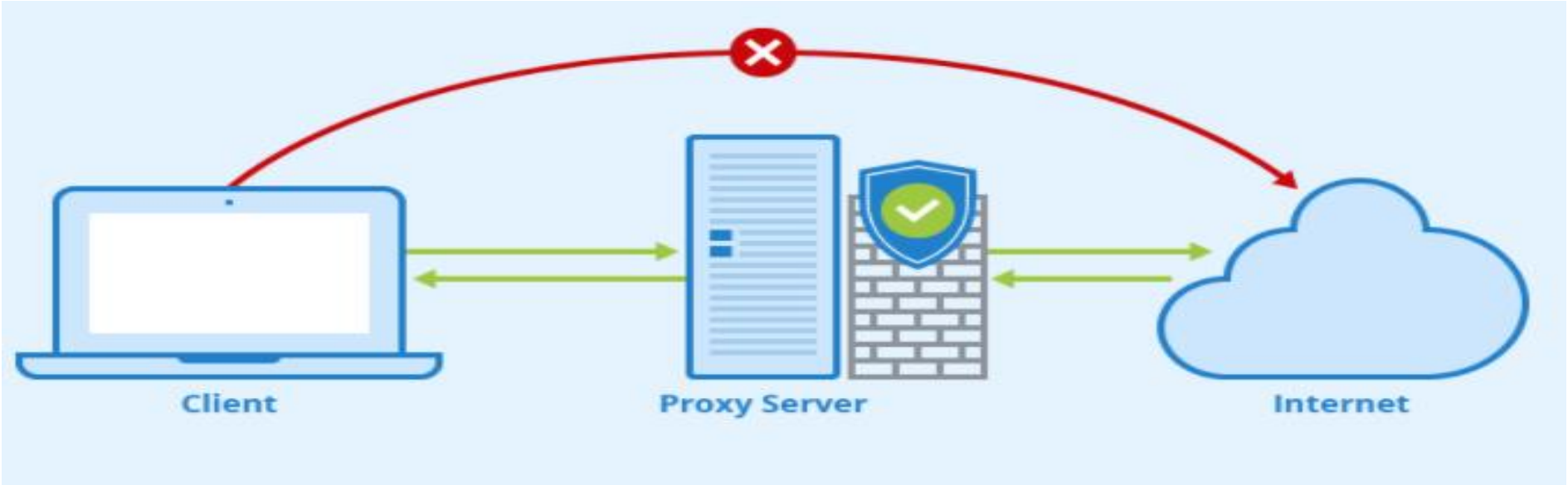
< 클라이언트 단 해결 방법 >

- 직접 구현한 서버가 없어 클라이언트 내에서만 해결을 해야 할 경우
- 해결 방법 : 프록시 설정

Same-origin 정책과 CORS이슈 해결 방법이 무엇인가요?

Cross-Oring Resource Sharing(CORS)

- 프록시란? 직접 통신하지 못하는 두개의 컴퓨터 사이에서 서로 통신할수 있도록 돕는 역할



Same-origin 정책과 CORS이슈 해결 방법이 무엇인가요?

Cross-Oring Resource Sharing(CORS)

- 클라이언트-서버 중간에서 요청을 받아 Access-Control-Aoolw-Origin 헤더를 추가해주는 중간 다리 역할을 놔주는 방법으로 해결

```
https://cors-anywhere.herokuapp.com
```

- Url앞에 적어두고 요청을 보내기. 끝!
- Ajax 요청이 프록시 서버를 건너며 Access-Control-*설정이 되어 요청에 대한 응답을 받을 수 있음

Same-origin 정책과 CORS이슈 해결 방법이 무엇인가요?

Cross-Oring Resource Sharing(CORS)

```
function convertToCorsUrl(url){
    var protocol = (window.location.protocol === 'http:' ? 'http:' : 'https:');
    return protocol + '//cors-anywhere.herokuapp.com/' + url;
}
```

- Url을 파라미터로 넘겨주어 rsult를 받아오기
- 내부 작업은 cors-anywhere.herocuapp.com에서 모두 처리하여 cors방지 로직이 적용된 서버에는 접근하지 못한다.
- https://github.com/Rob--W/cors-anywhere

Same-origin 정책과 CORS이슈 해결 방법이 무엇인가요?

Cross-Oring Resource Sharing(CORS)

- 직접 proxy를 만들기
- JSONP 요청을 보내 스크립트 파일을 호출하여 우회

감사합니다