



## #1 / Security shoud not be so scary!

What do you imagine when someone says "securiy"?

누군가 "보안" 이라고 말했을 때, 당신은 무엇을 떠올리십니까?

---

Hackers? Attacks? Defenses?

A programmer in a black hoodie in a dark room?

해커들? 공격하는 사람들? 방어하는 사람들?

어두운 방에서 검정색 후드를 뒤집어쓴 사람?

---

When the word "security" comes to mind,  
it's usually in the context of bad news.

"보안"이라는 말을 들을 때면,  
보통 나쁜 소식과 관계가 있는 경우가 대부분입니다.

You often encounter headlines like  
"A big social network leaked login passwords" or  
"an attacker stole credit card information from a shopping site".

당신은 종종 "대규모 소셜 네트워크 서비스가 로그인 패스워드를 유출했다"거나

"공격자가 어느 쇼핑 사이트로부터 신용카드 정보를 훔쳤다"는 등의  
헤드라인들을 마주쳤을 것입니다.

---

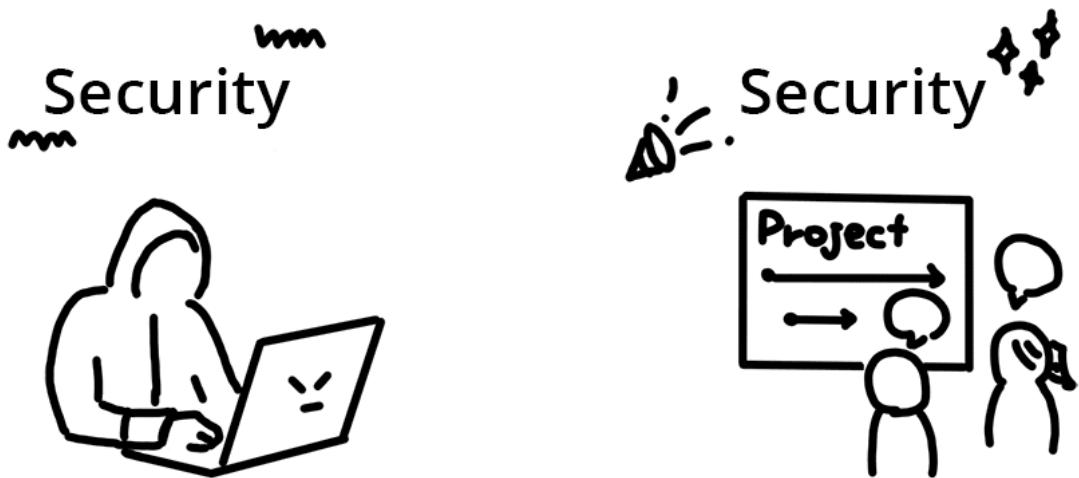
But security is something to be taken as positive and necessary part of

web development

just like "user experience" or "accessibility"

그러나 보안은 "사용자 경험"이나 "접근성"처럼

웹개발의 필수적이고 긍정적인 부분으로 받아들여져야 합니다.



A hacker in hoodie is negative security image.

A team working on a project together is a positive security image.

후드를 뒤집어쓴 해커는 보안의 부정적인 이미지를 상징합니다.

함께 프로젝트를 진행하고 있는 팀은 보안의 긍정적인 이미지를 상징합니다.

In the next few guides,  
you'll learn how to keep your business and your users' content secure.

다음 몇개의 가이드를 통해,

당신은 어떻게 당신의 사업과 당신 사용자들의 컨텐츠를 안전하게 지킬 수 있는지 배울 수 있을 것입니다.

---

## What is a security vulnerability?

### 보안 취약점이란 무엇인가?

In software development,

when an application does not work the way it is intended to work,

it's called "a bug".

소프트웨어 개발에 있어서,  
어플리케이션이 의도한대로 동작하지 않는 경우,  
그것은 "버그" 라고 불리워 집니다.

Sometimes a bug displays wrong information or crashes on a certain action.  
때때로 버그는 잘못된 정보를 화면에 띄우거나 특정 동작에서 크래쉬를 일으킵니다.

A **vulnerability** (sometimes called a **security bug**)  
is a type of bug that could be used for abuse.

취약점 (때때로 보안 버그라고 불리우는) 은  
남용되거나 악용될 수 있는 유형의 버그를 뜻합니다.

---

Bugs are common in the day to day activities of a developer.  
개발자들의 일상에서 버그는 굉장히 흔합니다.

Which means, vulnerabilities are also frequently introduced into applications.

이는, 취약점또한 어플리케이션에 자주 들어오게 된다는 것을 뜻합니다.

What's important is that you are aware of common vulnerabilities in order to mitigate them as much as you can.

중요한 점은 이러한 취약점을 가능한한 많이 다루기 위해서  
당신은 항상 흔히 발생할 수 있는 취약점들을 주의하고 있어야 한다는 것입니다.

It is just like minimizing other bugs  
by following common patterns and techniques.

이는 개발자들 사이에서 지켜지는 관습이나 자주 사용되는 기술들을 따름으로써  
여러 버그들의 발생을 최소화 하는 것과 같다고 할 수 있습니다.

---

Most security techniques are just good programming,  
for example:

대체로 보안 관련 기술들은 좋은 프로그래밍을 하는 데 있어 도움을 줍니다,  
예를 들어:

- Check values entered by a user  
(not null, not an empty string, checking the amount of data).  
사용자로부터 들어온 입력을 검사한다.  
(NULL 인지, 빈 문자열인지, 데이터의 양은 얼마나되는지)
  - Ensure a single user can't take up too much time.  
한 사용자가 너무 많은 시간을 소모하지 않도록 한다.
  - Build unit tests so security bugs can't slip in by accident.  
우연한 버그의 발생을 막기위해 유닛 테스트를 작성한다.
- 

## What are security features?

### 보안 관련 기능이란 무엇인가?

Your first lines of defense are security features such as HTTPS and CORS.

(You'll learn about these acronyms later so don't worry about them for now.)

당신의 첫번째 방어선은 HTTP 나 CORS 같은 보안 관련 기능입니다.  
(나중에 이들에 대해 배우게 될테니 지금 너무 겁먹지 마십시오.)

For example, encrypting data using HTTPS might not be fixing a bug, but it protects the data you're exchanging with users to other parties.  
(Intercepting data is a common attack.)

예를 들어, HTTPS 를 사용해 데이터를 암호화하는 것은  
버그를 수정하지는 않을테지만,  
당신과 당신의 사용자가 교환하고 있는 데이터를 제 3자로부터 보호합니다.  
(데이터를 가로채는 것은 흔한 공격방식중 하나입니다.)

---

## What's the impact?

### 어떤 영향이 있는가?

When an application is not secure, different people could be affected.

만약 어플리케이션이 안전하지 않다면,  
여러 사람들이 영향을 받을 수 있습니다.

- Impact on users

사용자에게 끼치는 영향

- Sensitive information, such as personal data, could be leaked or stolen.

개인정보와 같은 민감한 정보가 노출되거나 도둑맞을 수 있습니다.

- Content could be tampered with.

컨텐츠가 함부로 변경될 수 있습니다.

A tampered site could direct users to a malicious site.

누군가에게 함부로 손대진 사이트는 사용자들을 악성 사이트로 유도할 수 있습니다.

- Impact on the application

어플리케이션에 끼치는 영향

- User trust may be lost

사용자의 신뢰를 잃게 될 것입니다.

- Business could be lost due to downtime or loss of confidence as a result of tampering or system shortage.

시스템의 문제나 누군가의 악의적 행동으로 인해

당신의 사업이 중단되거나 사람들로부터 신뢰를 잃어

피해를 입을 수 있습니다.

- Impact on other systems

다른 시스템들에 끼치는 영향

- A hijacked application could be used to attack other systems, such as with a denial-of-service attack using a botnet.

누군가에 의해 강탈된 어플리케이션은

봇넷을 이용한 서비스 거부 공격에 사용되는 것처럼

다른 시스템들을 공격할 수 있습니다.

Actively securing your application is not only crucial for you and your business, but also for your users, protecting them and other systems from attacks launched from your site.

당신의 어플리케이션을 보다 적극적으로 안전하게 만드려는 노력은

당신과 당신의 사업뿐만 아니라

당신의 사이트를 이용한 공격으로부터 당신의 사용자와 다른 시스템들을 보호하는데 굉장히 중요합니다.

---

## Wrap up

### 마무리

Congratulations! You are halfway through this introduction.

축하합니다. 당신은 이제 도입부를 반쯤 지나오셨습니다.

Now you know the difference between security vulnerabilities and features, and you are aware that not only you but everyone else gets affected when your application is not secure.

이제 당신은 보안 취약점과 보안 관련 기능의 차이를 알고,  
당신의 어플리케이션이 안전하지 않는다면,  
당신뿐만 아니라 모든사람들이 영향을 받는다는 것을 알게 되었습니다.

The next guide covers the types of attacks in depth to make security even less scary.

다음 가이드에서는 당신이 좀더 보안에 친숙해지도록 어떤 유형의 공격이 있는지 알아보겠습니다.