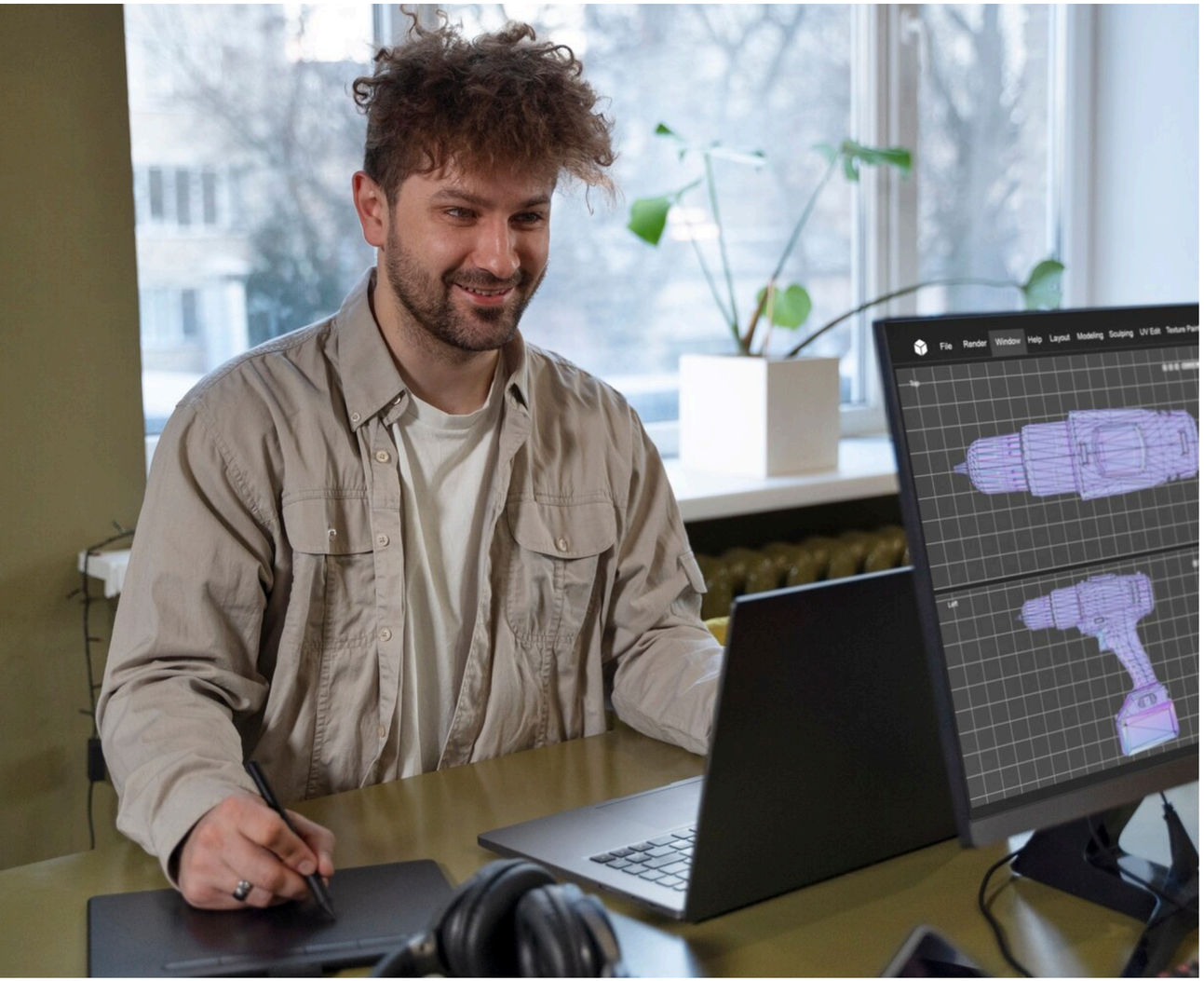
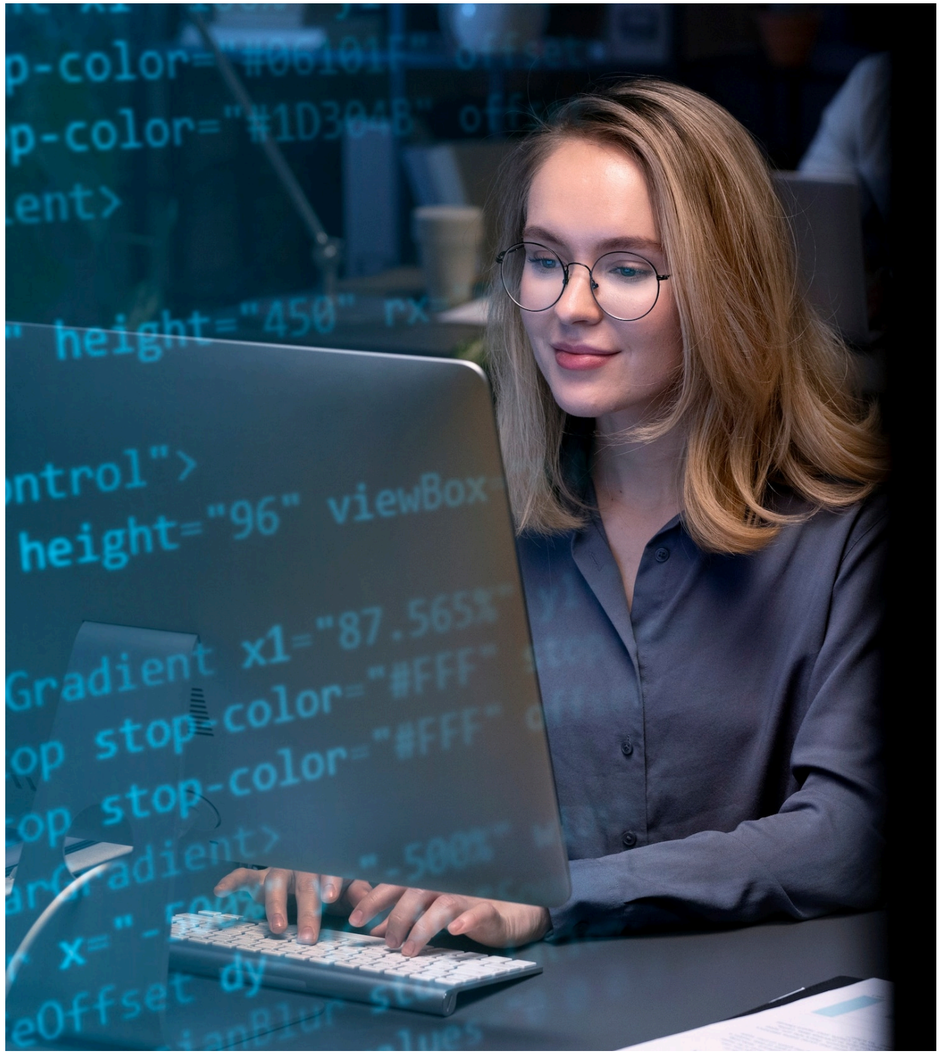


NODE.JS 보안: JWT를 활용한 취약점 분석 및 예방 전략

소개

Node.js는 웹 애플리케이션 개발에 널리 사용되는 플랫폼입니다. 그러나 **보안** 취약점이 존재할 수 있으며, 이를 예방하기 위한 전략이 필요합니다. 이 발표에서는 **JWT**를 활용한 취약점 분석과 예방 전략을 다룰 것입니다.



JWT란 무엇인가?

JWT(JSON Web Token)는 클라이언트와 서버 간의 **인증** 정보를 안전하게 전송하기 위한 방법입니다. 이 토큰은 **서명**되어 있어 변조를 방지하며, 다양한 **정보**를 포함할 수 있습니다. JWT의 구조와 동작 방식에 대해 알아보겠습니다.

NODE.JS의 보안 취약점

Node.js 애플리케이션은 **코드 취약점**, **의존성 문제**, 그리고 **잘못된 구성**으로 인해 보안 위협에 노출될 수 있습니다. 이러한 취약점을 이해하고, 이를 통해 발생할 수 있는 **위험**을 최소화하는 것이 중요합니다.

JWT를 활용한 예방 전략

JWT를 사용하여 **인증** 및 **인가**를 강화할 수 있습니다. 유효성 검사, **만료 시간** 설정 및 **비밀키** 관리와 같은 전략을 통해 보안을 향상시킬 수 있습니다. 이러한 방법을 통해 애플리케이션을 보다 안전하게 보호할 수 있습니다.

결론

Node.js 애플리케이션의 보안을 강화하기 위해서는 JWT를 활용한 접근 방식이 효과적입니다. 취약점을 이해하고, 예방 전략을 수립함으로써 보다 안전한 웹 애플리케이션을 구축할 수 있습니다. 지속적인 보안 점검이 필요합니다.

Thanks!