[별첨19-1] 결과보고서

2025학년도 국립한밭대학교
SW중심대학 산학협력프로젝트 결과보고서

<table>
<tr><td rowspan="2">과제명</td><td colspan="4">(국문) Zero Trust 기반 원격 근무 보안 강화 프로젝트</td></tr>
<tr><td colspan="4">(영문) Zero Trust-Based Remote Work Security Enhancement Project</td></tr>
<tr><td rowspan="2">과제책임자
(교수)</td><td>소속 학과</td><td>컴퓨터공학과</td><td>성 명</td><td>김태훈</td></tr>
<tr><td>E-mail</td><td>thkim@hanbat.ac.kr</td><td>연락처</td><td>1151</td></tr>
<tr><td rowspan="3">참여기업</td><td>기 업 명</td><td>㈜하얀마인드</td><td>대표자 성명</td><td>오정민</td></tr>
<tr><td>주 소</td><td>대전광역시 유성구 대학로 157
대전스타트업파크D1</td><td>연락처</td><td>010-2711-1357</td></tr>
<tr><td>주 업 종</td><td>서비스업</td><td>E-mail</td><td>jmoh@hayanmind.com</td></tr>
<tr><td>수행기간</td><td colspan="4">2025년 4월 1일 ~ 2025년 11월 30일 (8개월)</td></tr>
<tr><td>신청 과제비</td><td colspan="4">총 20,000 천원
(SW중심대학 지원금 : 최대 15,000 천원, 기업 현물 : 5,000 천원)</td></tr>
<tr><td>참여학생</td><td colspan="4">유용상(컴퓨터공학과), 장예나(컴퓨터공학과)</td></tr>
<tr><td>Github주소</td><td colspan="4">https://github.com/youyongsang/ZeroTrust_new_version.git</td></tr>
<tr><td rowspan="3">수행 결과 요약
(결과 사진 또는 화면 캡쳐 이미지 포함)</td><td>개요</td><td colspan="3">기존의 내부망 중심 보안 모델은 원격 근무 환경에서 보안 위협에 노출될 확률이 높다. 원격 근무 환경에서 발생할 수 있는 보안 문제를 해결하기 위해 Zero trust 모델 적용 방안을 연구한다. 이를 위하여 기존 보안 모델의 한계를 분석하고, Zero Trust 모델의 장점과 실무적 적용 방안을 제시하여 기업 데이터 보호와 보안 강화를 모색한다. 이를 위해 Zero Trust 아키텍처 설계, MFA 적용, 사용자 행동 기반 이상 탐지 시스템 구축, 보안 테스트를 진행하며 기업 환경에 최적화된 보안 정책과 기술적 접근법을 제시한다.</td></tr>
<tr><td>국립한밭대</td><td colspan="3">Zero Trust 모델 적용, 원격 근무 환경에서의 보안 강화 방안 모색</td></tr>
<tr><td>기업</td><td colspan="3">서버 구축 환경 공유 / 원격 근무 환경에서 발생하는 보안 문제 제공 / 요구 사항에 대한 데이터 제공</td></tr>
</table>

<프로젝트 결과 보고서 본문>

1. 과제 필요성 및 목표

1) 과제 개요

현대에 들어서면서 다양한 분야에서 원격근무가 이루어지며 그중에서도 IT관련 직종에서 대중화된 근무 방식으로 자리잡고 있다. 영어 쉐도잉 앱 '미미킹'부터 유튜브 기반 영어듣기 어플 '레드키위'를 중심으로 하여 현재 중소기업의 범용 고객관리 솔루션 AI인 '모네타이' 개발까지 손 뻗고 있는 기업 하얀마인드 또한 위와 같은 근무 방식을 적극적으로 도입하고 있다. 장소에 구애받지 않으면서 사용되는 화상회의(Zoom, MS Teams), 메신저(Slack, Discord), 클라우드 저장소(Google Drive, OneDrive) 등의 디지털 협업 도구가 이러한 환경에서 사용된다. 개발자들에게 유용한 개발 환경이 갖춰짐에도 불구하고 기존의 네트워크 중심 보안 모델은 사내 네트워크 내부환경은 안정성을 갖추지만, 외부 네트워트의 환경에서는 보안 문제에 대해서 노출 될 위험이 있으며 아래와 같은 문제점을 가진다.

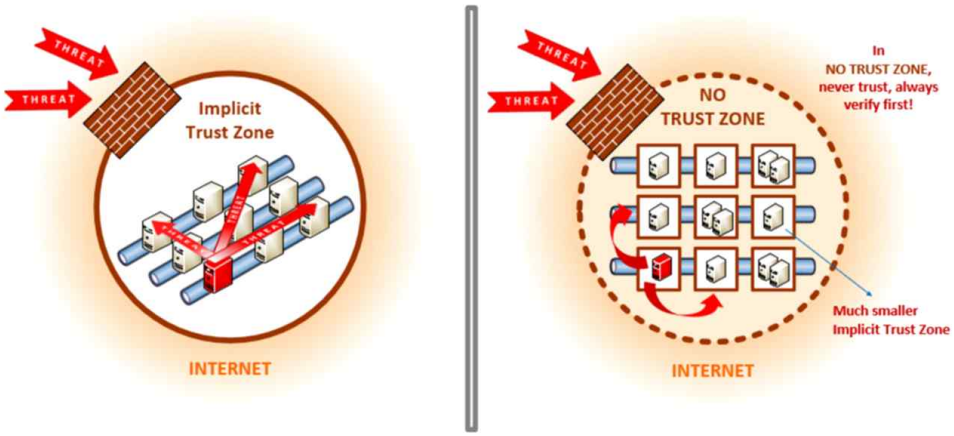
- 경계 기반 보안의 붕괴
기존 모델은 방화벽과 VPN을 통해 기업 내부망을 보호하는 구조이지만, 원격 근무자는 기업 네트워크 바깥에서 접속하기 때문에 보안 경계가 모호해진다.

- VPN의 보안 및 성능 문제
VPN을 통해 원격으로 사내 네트워크에 접속하지만, 계정이 탈취되거나 VPN 서버가 공격받을 경우 기업 전체가 보안 위협에 노출된다.

- 다양한 디바이스 및 접속 환경
직원들은 개인 노트북, 태블릿, 스마트폰 등 다양한 기기를 사용하며, 기업이 보안 설정을 완벽하게 통제할 수 없다.

- 내부자 위협과 계정 탈취 위험
기존 모델에서는 한 번 인증되면 내부 사용자를 신뢰하는 구조이므로, 계정 유출 또는 내부자의 악의적인 접근 시 보안이 취약하다.

이러한 문제점를 통해서 원격 근무 환경에서도 안정성을 높일 수 있는 Zero Trust라는 보안 모델을 적용하고자 한다. Zero Trust는 **"기본적으로 아무도 신뢰하지 않는다"** 라는 원칙을 기반으로 모든 접근 요청을 지속적으로 검증하는 보안 모델이다. 단순히 기업 내부망에 접속했다고 해서 신뢰하는 것이 아니라, 사용자, 디바이스, 네트워크 환경을 실시간으로 평가하고, 정책을 기반으로 접근을 허용 또는 차단한다.



(a) 전통적인 단일 경계 방어 (b) 제로 트러스트 방어 모델

전통적인 경계방어와 Zero Trust 모델 비교
(출처: Zero Trust Cybersecurity: 'Never Trust, Always Verify', NIST, Oct. 2020)

미국 국립표준기술연구소(NIST)는 2020년 8월 11일 Zero Trust Architecture(ZTA)를 다루는 특별 간행물 800-207(Special Publication 800-27)을 발표 하였으며 모델을 정의하고 원칙 아키텍쳐 설계, 구현을 위해 필요한 7가지 원칙을 발표했다.

① Resources : 모든 데이터, 컴퓨팅 서비스는 보호 대상이다.

② Communication : 내부망/외부망 관계없이 동일한 보안 요구를 충족해야 한다.

③ Per-session Access : 접근 요청을 세션 단위로 검증하여 권한을 부여한다.

④ Dynamic Policy : 클라이언트 ID, 소프트웨어 버전, 네트워크 위치, 요청 시간 등의 동적 정보를 활용하여 정책을 결정한다.

⑤ Monitoring : 보안 상태를 지속적으로 모니터링하고 평가한다.

⑥ Authentication & Authorization : 모든 인증 및 권한 승인은 동적으로 수행해야 한다.

⑦ Continuous Improvement : 환경을 지속적으로 개선, 데이터를 수집해 보안 전략 최적화한다.

Zero Trust 구현의 7원칙
(출처 : https://www.techtarget.com)

마이크로소프트는 이 7가지 원칙 중에서도 핵심 원칙 3가지를 강조했다.

① 명확한 검증 : 모든 접근 요청을 검증하며, 계정 탈취 여부를 지속적으로 점검한다.

② 최소한의 권한 액세스 : 필요할 때만 최소 권한을 부여한다.

③ 침해 가정 : 내부 시스템도 안전하지 않다고 가정하고 보안 설계를 수립한다.

위와 같이 발표된 Zero Trust보안 모델은 2011년부터 Google에서 BeyondCorp 프로젝트를 통해 VPN 없는 Zero Trust 보안 환경 구축을 시작으로 Netflix의 Zero Trust 기반 접근 제어 및 내부 서비스 보호, Microsoft의 Azure AD와 Endpoint Manager를 결합하여 Zero Trust 원칙을 적용한 클라우드 보안 모델 운영 등 현장에서도 직접적으로 활용되는 기술로 자리 잡고 있다.

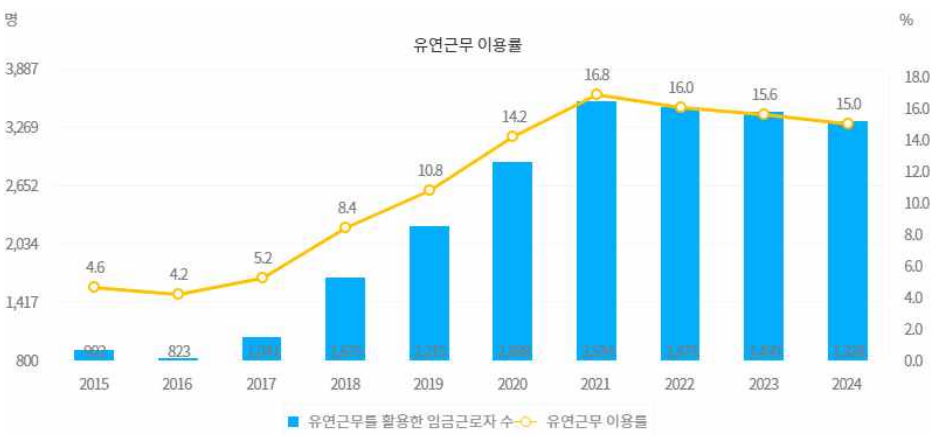
위와 같은 사례를 통해서 Zero Trust라는 보안 모델을 기반으로 개발 환경에서 늘어나는 근무형태인 원격 근무에서의 발생하는 사용자의 안정성에 대한 문제점을 해결하고자 한다.

2) 과제 필요성

1. 원격 근무 환경의 확산과 보안 변화의 필요성

2019년에 시작된 코로나19 팬데믹 이후, 원격 근무는 단기적인 대책이 아니라 장기적인 근무 방식의 변화로 자리 잡았다. 이후로 공무원 근무 혁신 지침을 통해 공공기관에서도 비대면 근무를 장려했고, 이후 기업에서도 원격 및 유연 근무 방식을 채택하는 사례가 늘어났다.

통계청 「경제활동인구조사 근로형태별 부가조사」에 따르면, 2020년 이후 원격 및 유연한 근무를 도입한 기업의 수가 지속적으로 증가하였으며 코로나 팬데믹 시기가 지난 이후 최근에 들어서도 유연근무 이용률은 코로나시기의 연도들과 비교시 크게 감소하지 않는 상황이다. 이처럼 업무 환경이 변화함에 따라 기업의 보안 환경도 변화할 필요성이 존재하는 상황이지만 기존의 내부망 중심 보안 모델은 원격 근무 환경을 갖추기에 기술적으로 한계를 가지고 내부망을 사용하지 않을 시 보안 위협에 노출될 확률이 높다.



(출처: 통계청, 경제활동인구조사 근로형태별 부가조사)

2. 기존 보안 모델의 한계와 새로운 위협 요소

회사 내부에서 근무하는 것을 가정한 기존 보안 모델은 원격 근무 환경에서 보안 위협으로 돌아올 수 있다.

유형	의미	예시
물리적 위협	회사가 제공하는 수준의 안전한 근무환경 (단말기 유실, 위변조 대책 등)이 보장되지 않아 생기는 위협	- 불특정 다수가 모이는 카페, 도서관 등의 장소에서 장비 도난 - 이동 중 업무용 전산 장비 분실 또는 도난 등
인적 위협	비대면 방식으로 업무를 수행할 때 노출될 수 있는 각종 사회공학적 공격, 또는 의도하지 않은 비정상적인 작업(행위)으로 인한 위협	- 회사의 중요 자료가 원격근무에 사용되는 사용자 단말기를 통해 외부로 유출 - 재택근무 시 가족 및 방문자, 또는 아이들이 업무용 전산장비에 접근하여 자료 수정, 삭제 등
기술적 위협	보안에 취약한 사용자 단말기가 악성코드에 감염되어 인가되지 않은 사용자(해커)가 회사 내부망 침투해 생기는 위협	- 원격 근무에 사용되는 네트워크 환경 (와이파이 장비 등)이 안전하지 않아 통신 내용 또는 데이터가 유출 - 업무 처리 시스템의 접속 인증 절차가 부실하여 허가받지 않은 단말기 등이 사내 네트워크에 접속

이 중 네트워크 관점에서의 보안 문제를 정리하면 아래와 같다.

① VPN 보안 취약점

VPN은 한 번 로그인하면 내부 시스템에 자유롭게 접근 가능하지만 계정이 탈취되거나 악성코드에 감염된 경우, 기업 전체가 보안 위협에 노출될 수 있다. 트래픽이 암호화되어 있더라도, 내부 네트워크가 안전하지 않으면 공격자가 내부 시스템에 쉽게 침투할 수 있다.

② 사용자 검증 취약성

기존 보안 모델에서는 ID/PW 정보만 검증하여 접근을 허용하는 경우가 많다. 도난당한 계정이 사용될 경우 쉽게 기업 내부 시스템에 접근 가능하며, 개인 기기를 사용하는 경우 기기의 보안 상태를 알 수 없어 기업 데이터 유출 가능성이 증가한다.

③ 보안이 취약한 네트워크 환경

원격근무자는 내무망 이외의 네트워크 환경에서 근무한다. 이때 외부에서의 공공 Wi-Fi를 사용할 가능성이 있다. 이러한 행위는 중간자 공격(Man-in-the-Middle Attack), 네트워크 도청 등 문제점이 발생하고 더 나아가 협업 도구(Slack, Zoom, Google Drive)의 악용의 가능성을 가진다.

3) 과제 목표

본 연구에서는 원격 근무 환경에서 Zero Trust 모델을 효과적으로 구현하는 방안을 탐색하고, 기업이 실제 업무 환경에서 적용할 수 있는 보안 정책과 기술적 접근법을 분석한다. 추가로, 원격 근무 확산에 따른 기존 경계 기반 보안 모델의 한계를 극복하고자 한다. 이를 위해 VPN 모델과 Zero Trust 모델을 비교 분석하고, 원격 근무 환경에 적합한 실무적인 Zero Trust 적용법을 탐구

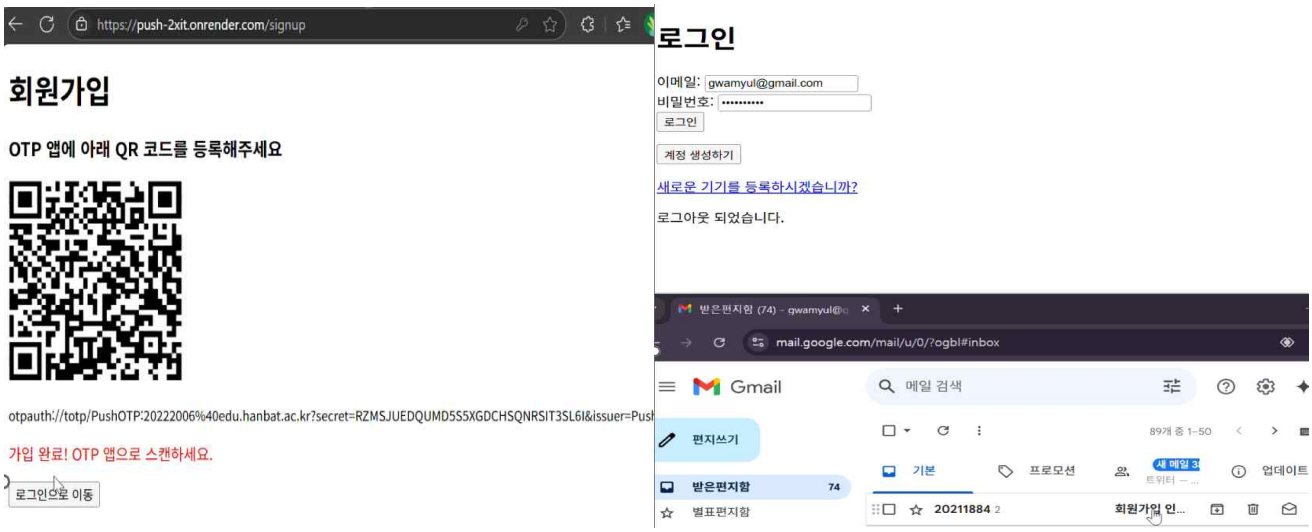
한다. 최종적으로 기업이 도입 가능한 구체적인 보안 정책과 기술 솔루션(SDP, SASE)을 제안한다. 이를 통해 계정 탈취 후 내부망 확산을 방지하고, 모든 기기와 접속 환경을 검증하여 데이터 유출을 차단한다. 궁극적으로 장소에 구애받지 않는 높은 수준의 원격 근무 보안 태세를 확립한다.

2. 과제 수행 결과

1) React 및 FastAPI를 이용한 시스템 환경 개발

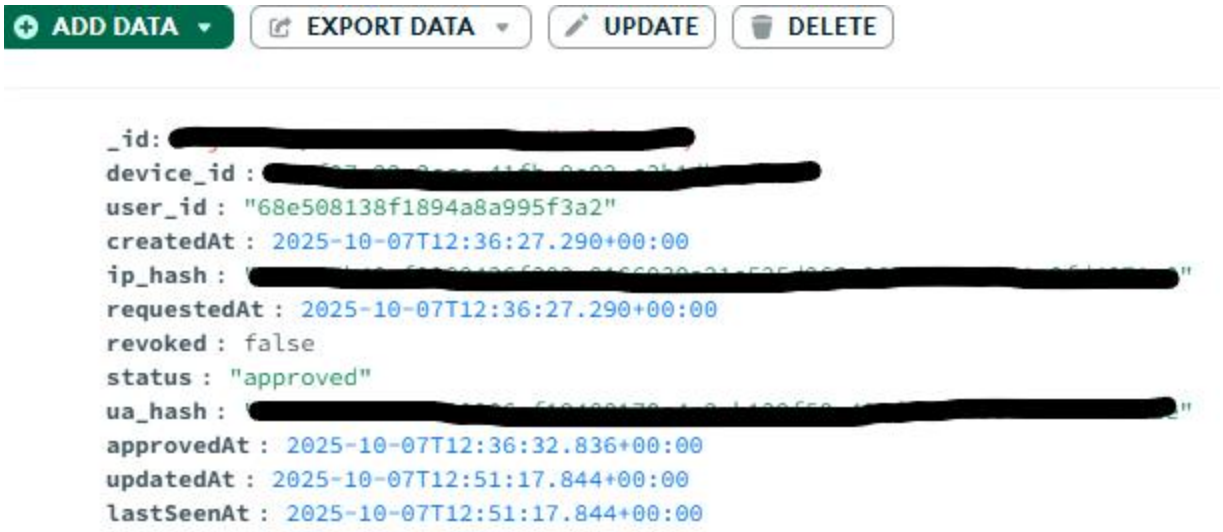
본 과제는 Zero Trust 환경에서의 다중 인증 방식을 비교 분석하기 위해 3-Tier 아키텍처를 기반으로 구축하였다. 전체 시스템은 로컬 개발 환경에서 구성되었다.

프레젠테이션 계층인 사용자 인터페이스는 React 18을 사용하여 구현하였으며, 사용자는 이 클라이언트를 통해 로그인, TOTP 코드 입력, Push 알림 수신 및 승인/거부 등의 상호작용을 수행한다.
애플리케이션 계층인 핵심 인증 로직 및 API 서버는 FastAPI를 기반으로 구축하였고, FastAPI의 비동기 처리 방식을 활용하여 인증 요청 및 응답 시간을 정밀하게 측정하고 빠른 성능을 확보하였다. 또한, 데이터 계층은 MongoDB 데이터베이스를 사용하여 사용자 정보, 암호화된 비밀번호, MFA 설정, 기기 정보를 저장 및 관리하도록 하였다. 마지막으로 Push 인증 기능의 구현을 위해 Google의 Firebase Cloud Messaging을 연동하여, 서버가 인증 요청 시 FCM을 통해 사용자의 모바일 기기로 알림을 전송하도록 설계하였다.



OTP 앱 인증 과정과 기기 인증 과정

admin 페이지에서 확인한 로그



3) Zero Trust 원칙 기반 다중 인증 기능 개발

본 과제 수행을 통해 Zero Trust의 “Never Trust, Always Verify“ 원칙을 반영한 다중 인증 시스템의 백엔드 로직과 클라이언트 UI를 개발하였다. 개발된 시스템은 1차 인증 성공 후, 사용자별로 사전에 설정된 2차 인증 방식을 동적으로 분기하여 처리하는 핵심 인증 플로우를 구현하였다.

첫 번째로 ’Device 인증’ 기능을 구현하였다. 클라이언트는 최초 실행 시 자체적으로 랜덤 고유 식별자를 생성하여 서버로 전송한다. 서버는 전달된 dev_id를 MongoDB에 저장된 등록 기기 목록과 비교하며, 미등록 기기일 경우 서버 관리자 화면에 승인 요청 팝업이 즉시 표시된다. 관리자가 수락을 선택하면 해당 dev_id가 MongoDB의 기기 등록 컬렉션에 저장되어 정식 등록 기기로 관리되며, 이후 동일한 디바이스는 추가 승인 없이 다음 인증 단계로 이동한다. 이 구조를 통해 미등록・위조 기기의 비인가 접근을 근본적으로 차단한다.

두 번째로 ’TOTP 인증’ 기능을 구현하였다. 사용자는 회원가입 시 발급된 QR 코드를 ’Google Authenticator’ 등 표준 OTP 인증 앱에 등록하며, 이후 앱에서 생성되는 6자리 TOTP 코드를 로그인 과정에서 입력한다. 서버는 pyotp 라이브러리 기반 검증 알고리즘을 통해 유효성을 실시간 확인한다. 또한, 서버는 OTP 실패 횟수를 관리하여 3회 이상 연속 실패 시 일시적으로 로그인 시도를 제한함으로써 무차별 대입 공격을 효과적으로 차단한다.

세 번째로 'Push 인증' 기능은 Firebase API를 연동하여 구현하였다. 1차 인증 성공 시, 서버는 FCM을 통해 해당 사용자의 등록된 모바일 기기로 승인 혹은 거부 알림을 전송하며, 사용자의 응답을 콜백 받아 인증을 완료한다. 이는 가장 빠르고 편리한 UX를 제공한다.

이와 같은 구조는 기기, 사용자 지식, 사용자 편의성을 결합한 이중 다중 검증 체계로, 제로 트러스트 모델의 핵심 요소인 지속적 검증을 구현한다.

3) 논문 및 보고서 작성

연구 결과를 기반으로 논문을 작성하여 학회 발표(2025년도 한국통신학회 하계학술대회)를 수행하고 개발 프로그램을 Github에 공개(깃헙링크넣기) 하였으며, 최종적으로 프로젝트 결과 보고서를 작성하였다.

Zero Trust 환경에서의 다중 인증 기술 설계 및 분석 연구

장예나, 유용상, 방인규*, 김태훈*

국립한밭대학교 컴퓨터공학과, *국립한밭대학교 지능미디어공학과

{20222006, 20211884}@edu.hanbat.ac.kr, {ikbang, thkim}@hanbat.ac.kr*

Analysis of Multi-Factor Authentication Technologies in a Zero Trust Environment

Yena Jang, Yongsamg Yu, Inkyu Bang*, Taehoon Kim*

Department of Computer Engineering, Hanbat National University

*Department of Intelligence Media Engineering, Hanbat National University

요 약

최근 클라우드 서비스와 원격 근무 확산으로 인해 기존 경계 기반 보안 모델의 한계가 두드러지면서, 사용자의 신원을 지속적으로 검증하는 MFA의 필요성이 강조되고 있다. 이를 위해 본 연구에서는 FastAPI와 MongoDB를 이용한 서버 환경과 React 및 Firebase 기반 클라이언트 환경을 구성하여 MFA 인증 절차를 실험적으로 구현한다. 이를 통하여 제로 트러스트 환경에서 동적 위험 평가 기반의 다중 인증 정책이 보안성과 사용성의 균형을 달성할 수 있는 효과적인 접근임을 확인한다.

I. 서론

로로나 팬데믹 이후 원격근무와 클라우드 기반 업무 환경이 급격히 확산되면서, 기업은 내부망 중심의 전통적 보안 모델만으로는 다양한 접속 환경을 안전하게 관리하기 어려워졌다. 이에 따라 Zero Trust 보안 패러다임이 새로운 대안으로 주목받고 있으며, 기업의 네트워크 접근 통제 및 인증 체계 전반에 변화를 가져오고 있다 [1]. 국내 보안 환경에서도 이러한 흐름에 발맞추어 제로 트러스트 기반 접근제어, 인증 구조 개선, 기업 보안 체계 강화를 다루는 연구가 활발히 진행되고 있다 [2].

본 논문에서는 Zero Trust 환경에서 적용 가능한 MFA 방식을 직접 구현하는 데 중점을 두고, 보안성과 사용자 편화성을 기반으로 TOTP 방식과 Push 인증 각 방식의 장단점을 분석한다. 나아가서, 실제 구현을 통해 얻어진 비교 결과는 Zero Trust 적용 환경에서의 MFA 선택과 설계에 실질적인 가이드라인을 구축할 수 있는 기반을 제안한다. 추가로, 이상 행위 탐지나 리스크 기반 인증 모듈의 초기 단계 설계로 확장할 수 있는 기초를 마련하는 데이터를 확인하고자 한다.

II. MFA 개념

Zero Trust의 핵심 원칙 중 하나는 항상 검증하는 것이며, 이를 실현하는 대표적 수단이 다중 인증 (Multi - Factor Authentication, MFA)에 해당한다. 전통적인 비밀번호 기반 인증은 피싱, 크리덴셜 스터핑, 재사용 공격 등 다양한 위협에 취약하므로, 단일 인증 수단에 의존하는 모델로는 Zero Trust의 요구사항을 충족하기 어렵다. 이를 보완하기 위해서 최근의 보안 시스템들은 TOTP (Time - based One - Time Password), Push Notification 인증, 디바이스 바인딩(Device-bound Token) 등의 방식을 조합하여 보안을 강화한다.

MFA는 지식(비밀번호), 소유(휴대기기, 인증 토큰), 내재(생체정보)의 서로 독립적인 인증 요소를 결합하면서 단일 요소로 탈취하는 것만으로는 접근 권한을 얻을 수 없도록 설계한다. 특히 이러한 다중 인증 과정에서 생성되는 로그 정보(IP, User-Agent, 타임스탬프, 디바이스 ID 등) 는 사후 분석 및 이상 행위 탐지 (Anomaly Detection)의 기반 데이터로 활용될 수 있다. 이러한 원리는 NIST SP 800-207(Zero Trust Architecture) 와 SP 800-63B(Digital Identity Guidelines)에 명시된 표준적 권고사항이다 [3].

본 논문에서는 기술적으로 MFA를 크게 다음 세 방향로 구분하여 수행하였다. OTP 기반 인증은 TOTP 알고리즘(RFC 6238)을 기반으로 주기적으로 새로운 코드를 생성하는 방식으로, 서버 상태를 최소화하면서 재사용 공격에 대한 저항성을 확보한다. 다만 사용자의 직접 입력이 필요하고, 코드 만료나 시간 동기 오차로 인한 UX 저하가 발생할 수 있다. 다음으로, Push Notification 인증은 서버가 모바일 기기로 승인 요청을 전송하고, 사용자가 승인/거부만 수행하면 인증이 완료되는 방식이다. 이 방식은 평균 응답 시간이 짧고 직관적인 UX을 제공하지만, 네트워크 중계 서버스(FCM, APNs)에 대한 의존성과 기기 탈취·루팅 시 승인 권한이 노출될 위험이 존재한다. 또한 디바이스 기반 인증은 특정 장치에 토큰을 바인딩하거나 승인된 디바이스만 접근하도록 제한함으로써, 계정 탈취 이후의 비인가 접근을 방지한다.

이러한 세 가지 범주의 MFA 중에서도 TOTP와 Push 기반 인증을 중심으로 구현·비교하여, 이에 디바이스 신뢰 기반 접근(device_login)을 결합하여 Zero Trust 환경에서의 다단계 검증 구조를 실험적으로 제시한다. 이 비교는 단순히 보안성을 평가하는 것을 넘어, 재현성, 운영 부담, 사용자 경험이라는 실무적 관점에서 MFA의 적용 가능성을 검증하기 위한 것이다.

■ Corresponding Authors: Inkyu Bang (ikbang@hanbat.ac.kr), Taehoon Kim (thkim@hanbat.ac.kr)

Ⅲ. MFA 구현 전략 및 실험

본 연구의 구현은 단순한 개념 검증이 아닌, 실제 서비스 환경을 모사한 엔드투엔드(End-to-End) 실험 구조로 설계되었다. 서버는 FastAPI 프레임워크를 기반으로 하며, MongoDB를 사용자·기기·세션 로그의 저장소로 활용한다. 클라이언트는 React 기반 웹 UI와 Firebase Cloud Messaging(FCM) 을 연동한 모바일 앱으로 구성하여 푸시 승인 플로우를 구현하였다. 각 구현은 basic_login, otp_login, device_login 세 브랜치로 분리되어 기능적 확장 과정을 체계적으로 검증할 수 있도록 설계되었다.

1. 구현 환경 및 진행 과정

basic_login 단계에서는 이메일/비밀번호 기반 인증과 이메일 링크 검증 절차를 포함하였다. 비밀번호는 Argon2 해시를 적용하여 고강도 해싱을 보장하고, 이메일 검증 토큰은 itsdangerous 시리얼라이저로 서명 및 만료 시간 검증을 수행한다. otp_login 단계에서, RFC 6238 기반 TOTP 알고리즘(pyotp) 을 도입하였다. 가입 과정은 이메일 코드 검증, 시크릿 발급, 최초 TOTP 검증으로 구성되며, 백업 코드(10자리 영숫자)를 생성해 해시로 DB에 저장한다. device_login 단계에서 디바이스 승인 토큰(devtk) 을 쿠키 기반으로 발급하여, 승인된 기기에서의 자동 로그인을 허용하였다. 미승인 기기는 이메일을 통해 1회 승인 과정을 거치며, 승인된 기기는 MongoDB의 devices 컬렉션에 해시화된 기기 정보를 저장한다. 이 구조는 Zero Trust의 '기기 신뢰도 기반 접근 제어'를 실제로 구현하였다. 최종적으로 실험 결과는 반복 시뮬레이션을 진행하여 수집된 인증 요청·응답의 타임스탬프, 결증 결과 코드, 응답 지연(ms) 로그 데이터에 기반하여 분석하였다.

표 1 구현 환경

항목	구성 내용
서버	python 3.11 / FastAPI 0.110 / Uvicorn 0.29
데이터베이스	MongoDB 7.0 (Atlas Cluster)
인증 라이브러리	pyotp (RFC 6238 TOTP 지원), argon2-cffi (비밀번호 해싱)
클라이언트	React 18 / Firebase Cloud Messaging (FCM)
실험 환경	Windows 11 / Chrome 브라우저 v120 / 테스트용 Android FCM 앱
측정 지표	인증 성공률, 응답 지연 (ms), 재사용 공격 차단율, 기기 승인 성공률

2. 실험 결과 및 분석

세 가지 구현은 보안성, 편의성, 운영성 측면에서 상호 보완적 특성을 보였다. TOTP는 시간 기반 일회용성(Time-based One-Time Password)으로 인해 재사용 공격에 강한 저항성을 보였으며, 서버 상태 유지가 불필요해 우수한 확장성을 보였다. 다만 사용자의 직접 입력과 시간 동기 오차로 인한 인증 실패 가능성이 존재했다. 이에 비해 Device 기반 인증은 승인된 기기 외 접근을 원천적으로 차단하였고, Push 인증은 평균 응답 시간이 1초 미만으로 가장 빠르며, 승인/거부만 수행하는 단순한 UX로는 우수한 편의성을 가졌다. 그러나 중계 서버스(Firebase) 의 존성과 기기 분실 시 승인 탈취 위험이 확인되었다. 운영 측면에서 TOTP는 독립적으로 동작해 유지 비용이 낮았고, Push 인증은 네트워크 상태에 따라 가용성이 영향을 받았다. Device 기반 인증은 초기 승인 과정이 필요하지만, 이후 로그인 절차가 단축되어 장기적으로 효율성이 높았다. 이상의 결과를 종합하면, TOTP는 보안성, Push는 편의성, Device는 신뢰도에서 각각 강점을 보였으며, Zero Trust 환경에서는 이들 상황 기반으로 혼합 적용하는 Adaptive MFA 정책이 가장 합리적임을 확인하였다.

표 2 MFA 방식별 기능적 비교

구분	보안성	편의성	평균 응답 시간	관리 복잡도
Basic	낮음	보통	350 ± 20	낮음
TOTP	높음	중간	420 ± 25	낮음
Push	중간	높음	210 ± 15	중간
Device	높음	중간	300 ± 18	중간

Ⅳ. 결론

본 연구는 Zero Trust 환경에서 MFA 기술을 실제로 구현하고, 보안성과 사용자 경험(UX)의 상충 관계를 실험적으로 분석하였다. 단일 MFA 방식의 한계를 넘어, 리스크 기반 동적 정책(Risk-Based Adaptive MFA) 을 결합하여 한다. 향후 연구에서는 AI 기반 이상 행위 탐지 모델과의 결합을 통해, 본 연구에서 수집된 인증 로그(TOTP 실패 패턴, Push 승인 지연, Device 재등록 빈도 등)를 입력 피처로 활용하여 MFA 기반 행위 분석형 보안 시스템으로 확장할 예정이다. 이를 통해 Zero Trust 기반 디바이스 인증 체계와 이상 행위 탐지 시스템의 교차 검증 체지탑으로써, 향후 실무 보안 아키텍처 설계의 기초적 도대를 제공할 것으로 기대된다.

ACKNOWLEDGMENT

본 연구는 국립한밭대학교 공학교육혁신센터의 「창의융합형공학인재양 성지원사업」 의 지원 및 2025년도 교육부 및 대전광역시의 지원으로 대전RISE센터의 지원을 받아 수행되었음 (2025-RISE-06-002)

참 고 문 헌

[1] 장인규 외, 「제로트러스트 동향 분석 및 기업 보안 강화 연구」, 정보보호학회논문지, 2023.

[2] 이상훈 외, 「제로트러스트 기반 접근제어를 위한 기업 보안 강화 연구」, 디지털융복합연구, 2023.

[3] 미국 국립표준기술연구소(NIST), 「SP 800-207: 제로 트러스트 아키텍처(Zero Trust Architecture)」, 2020 / 미국 국립표준기술연구소(NIST), 「SP 800-63B: 디지털 신원 가이드라인 (Digital Identity Guidelines)」, 2020.

2025년도 한국통신학회 추계학술대회 발표논문:
Zero Trust 환경에서의 다중 인증 기술 설계 및 분석 연구