최신 디지털 인증 기술 분석 보고서

2025년09월01

(사) 한국디지털인증협회

1. 최신 인증 기술 분석

가. 서론

현대의 디지털 환경에서 **이용자 인증(authentication)** 기술은 사이버 보안의 핵심 요소로 부각되고 있습니다. 기존에는 ID/비밀번호 조합에 의존한 1차 인증이 주류였지만, 오늘날 이러한 방식을 노린 피싱 및 크리덴셜 스터핑(대량 도난된 비밀번호 조합을 이용한 공격)이 끊이지 않아 막대한 피해를 야기하고 있습니다[1]. 이를 보완하기 위해 이중 인증, 일회용 코드 등도 도입되었으나 여전히 근본적인 한계가 드러나고 있어, **무자각성**(transparent)과 **높은 보안성**을 동시에 추구하는 새로운 인증 기술들이 활발히 개발되고 있습니다.

본 보고서는 보안 전문가 관점에서 최신 인증 기술들을 심층 분석합니다. 특히 **생체인증**, **패스워드리스 인증(패스키 포함)**, **블록체인 기반 분산 인증**, **위험 기반 인증**, **연속 인증**, **프라이버시 보호 인증**, **양자 안전 인증** 등 주요 현대 인증 기술의 원리와 구현, 보안성, 활용 사례, 표준 동향, 장단점, 그리고 향후 과제까지 폭넓게 다룹니다. 각 기술별로 기술적 배경과 작동 방식(1), 구현 프로토콜과 표준(2), 알려진 보안 강점과 취약점(3), 산업 분야별 활용 사례(4), 기술적·운영상 이점과 한계(5), 글로벌 표준화 및 정책 동향(6), 향후 발전 방향과 남은 과제(7)를 순차적으로 분석합니다.

나. 생체인증 (Biometric Authentication)

1) 기술 개요 및 작동 원리

생체인증은 사용자의 고유한 **신체적 또는 행동적 특성**을 분석하여 신원을 확인하는 인증 방식입니다. 지문, 안면, 홍채, 정맥 등 **신체적 생체정보**와 걸음걸이, 타이핑 습관, 음성 및 서명 패턴 같은 **행동적 생체정보**를 모두 포괄합니다[2]. 일반적인 생체인증 시스템은 사용자 등록 단계에서 생체 특성을 센서로 측정하여 특징 벡터(템플릿)를 생성·저장하고, 인증 시 **실시간으로 입력된 생체 데이터**와 등록 템플릿을 비교하여 **일치 정도**를 판정하는 방식으로 작동합니다

[3][4]. 예를 들어 지문 인증의 경우 사용자의 지문 이미지를 받아 특징 점들을 추출한 뒤, 미리 저장된 템플릿과 **매치율(match score)**을 계산하여 사전에 정한 임계치 이상이면 동일인으로 간주하는 식입니다.

물리적 특성이 변하지 않는 한 사용자가 매번 기억해야 할 비밀정보가 필요 없다는 점에서 생체인증은 사용자 편의성과 보안성을 모두 높일 잠재력이 있습니다.

2) 주요 구현 방식 및 프로토콜

생체인증은 **현장(local) 인증**과 **원격(remote) 인증** 방식으로 구현될 수 있습니다. 현장 생체인증은 스마트폰이나 PC 등 사용자의 단말 기기에서 지문 센서, 얼굴 인식 카메라 등을 활용해 **단말 내부**에서 인증이 이뤄지는 방식입니다.

예를 들어 스마트폰의 지문/얼굴 잠금해제나 Windows Hello의 얼굴인증은 센서와 인증 프로세스가 기기 내에서 수행되고, 성공 시 기기 내 안전영역에 저장된 키가 풀리면서 로그인이 이루어집니다. 특히 FIDO2/WebAuthn 기반 **패스워드리스 인증**에서는 사용자 디바이스가 **생체인증을 통해 잠금 해제**되는 보안 토큰 역할을 하여, 서버와 주고받는 것은 공개키 인증서와 서명뿐이고 지문 등 생체정보는 외부로 유출되지 않습니다[5][6]. 한편 *원격 생체인증*은 사용자가 원격 서버에 자신의 생체정보(예: 음성, 얼굴 영상)를 보내어 서버가 이를 검증하는 방식으로, 일부 금융기관의 화상 신원확인이나 콜센터의 목소리 인증 등에 활용됩니다. 이 경우 **바이오메트릭 템플릿**을 안전하게 저장하고 비교하는 프로토콜이 중요합니다.

국제 표준 ISO/IEC 19794, 30107 등이 지문·얼굴 등 모달리티별 데이터 형식과 **발표 공격 탐지(PAD)** 절차를 정의하고 있으며, FIDO 얼라이언스도 기기 내 생체인증 모듈에 대한 **Biometric Component 인증 프로그램**으로 성능과 보안 요건을 제시하고 있습니다. 예를 들어 FIDO 인증 기기는 지문인식 오인식률(FAR) 0.001 이하 등 일정 기준을 충족해야 하며, 센서 무결성 및 템플릿 보호에 대한 **HMAC 기반 기기 검증(attestation)** 정보를 원격 서버에 제공하여

신뢰성을 담보합니다[7][8]. 이러한 다층적인 표준과 프로토콜을 통해 생체인증은 단말 내 로컬 인증부터 서버 연계 인증까지 다양한 방식으로 구현되고 있습니다.

3) 보안성 분석 (공격 벡터 및 취약점)

생체인증은 **사용자 고유 특성**에 기반하므로 비밀번호 대비 추측 공격에 강하지만, 그 보안 강도는 **매칭 알고리즘 정확도**와 **센서 안전성**에 달려 있습니다. **오인식률(FMR)**과 **오거부률(FNMR)**은 성능의 핵심 지표로, NIST 가이드라인 등은 보안 용도로 사용하려면 생체인증 장치가 **0.1% 이하**의 오인식률(1:1000)을 가져야 함을 명시합니다[9]. 다만 이런 수치는 **정상 사용자의 임의 오인증**만 따진 것으로, 지문 위조나 얼굴 사진 사용 같은 **위장 공격(spoofing)**에는 취약할 수 있습니다[10]. 실제로 얼굴인식은 사진이나 딥페이크 영상으로, 지문인식은 실리콘 지문이나 **지문 초박막 필름** 등으로 속인 사례들이 보고되었습니다. 이러한 **프레젠테이션 공격**에 대응하기 위해 **활성도 검증(liveness detection)** 기술이 필수적으로 도입되고 있습니다.

예컨대 얼굴인식 시 눈 깜빡임이나 3D 얼굴 깊이센서를 통해 실존 여부를 확인하고, 지문인식은 손가락의 정전기 반응이나 혈류를 감지해 위조 지문을 가려냅니다. 표준 ISO/IEC 30107-3은 이러한 PAD 기술 도입 시 **공격 탐지 성공율 90% 이상**을 권고하고 있으며, NIST 역시 향후 PAD 구현을 **의무화**할 계획을 밝히고 있습니다[11][12].

근본적인 보안 취약점으로는 **생체정보의 불변성**이 있습니다. 비밀번호와 달리 지문이나 홍채는 유출되거나 도용되더라도 사용자가 바꿀 수 없다는 치명적인 한계가 있습니다[13]. 공격자가 고해상도 사진으로 사용자의 홍채 패턴을 획득하거나, 사용자가 만진 물건에서 지문을 채취하는 등으로 **생체 특성을 탈취**할 수 있으며[14], 일단 탈취된 생체정보는 그 사용자를 평생 노출시키는 위험 요소가 됩니다. 따라서 안전한 저장을 위해 **템플릿 보호 기법(Template Protection)**이 연구되고 있지만, 현재까지는 해시나 암호화 형태로 저장하여 역추적을 어렵게 만드는 정도이며 완벽한 대처법은 제한적입니다[15]. 이외에

도 **센서 자체에 대한 공격**(예: 지문센서 교체, 영상 스트림 조작) 가능성이 있어, FIDO2 등에서는 센서/장치 인증과 통신 채널 암호화를 요구함으로써 센서 신뢰성을 검증하도록 하고 있습니다[5][16].

요약하면 생체인증은 사용자 편의성과 높은 난이도의 위장 필요성 등으로 **일반적인 비밀번호 공격에는 강하지만**, 생체정보 유출이나 위조 공격, 센서 취약점에 대응하는 추가 보안 통제가 필수적입니다. Gartner도 "생체인증은 특징의 비밀성에 기대기보다는, **실제 사용자만이 통과할 수 있도록 센서+검출 기술로 위장 어려움을 확보**하는 데 달렸다"고 지적합니다[17].

4) 활용 사례 (산업별 적용 현황)

생체인증은 최근 스마트폰과 금융 분야를 중심으로 **폭넓게 상용화**되고 있습니다. 스마트폰의 지문인식과 3D 얼굴인식은 수억 명의 사용자가 일상적으로 기기 잠금해제나 모바일 앱 인증에 활용하는 대표 사례입니다. 특히 Apple Face ID나 Samsung, Google의 지문 인증은 FIDO2 표준을 통해 모바일 뱅킹, 전자상거래 앱 로그인 등에도 연동되어 비밀번호 없이 **생체로 앱 로그인**하는 경험을 제공하고 있습니다.

금융권에서는 ATM 기기에 지문인증을 도입하거나, 콜센터에서 고객의 **목소리 특징(Voiceprint)**으로 본인 확인을 하는 솔루션도 확산 중입니다. 예를 들어 일부 은행은 고객이 전화를 걸면 통화 중 목소리 패턴을 분석해 추가 질문 없이 신원을 확인하는 시스템을 운용합니다. 물리적 보안 측면에서는 기업 건물 출입에 홍채스캐너나 얼굴인식 게이트를 두어 직원 인증에 활용하고, 공항 출입국 심사에서는 여권의 ICAO 생체정보(얼굴·지문)와 실시간 촬영 영상을 대조하는 **e-게이트**가 널리 쓰입니다.

공공 분야에선 인도의 Aadhaar 같은 **범국가 생체인증 신원체계**가 수억 명에 대한 지문·홍채 정보를 관리하며 복지 지급 등 본인확인에 쓰이고 있습니다. 이처럼 금융, 통신, 출입통제, 전자정부 등 여러 산업에서 생체인증은 **편의성과 보안 강화**를 이유로 적극 도입되고 있습니다. 다만 분야별로 요구 보안수준과

사용자 경험이 달라, **멀티모달 생체인증**(두 가지 이상 생체 결합)이나 **생체+기타 요소 결합** 등으로 맞춤 설계되는 경우도 있습니다. 예컨대 최고보안을 요하는 군사 시설은 **홍채+안면 복합인증**을 쓰고, 반면 스마트폰 앱은 지문 하나로 충분히 사용자 편의를 중시하는 식입니다.

5) 기술적/운영상 장단점

생체인증의 가장 큰 **장점**은 사용의 편리성과 **높은 사용자 식별 정확도**입니다. 사용자는 자신의 몸이 곧 "패스워드"이므로 기억하거나 따로 소지할 것이 없으며, 센서 앞에 손가락이나 얼굴을 대는 **짧은 순간**에 인증이 완료돼 UX 측면에서 뛰어납니다. 또한 생체정보는 개인마다 고유하고 위조가 어렵기 때문에(충분한 PAD 적용을 전제로) 비밀번호처럼 추측되거나 무차별 대입될 위험이 작고, **공유**나 **재사용**이 사실상 불가능하여 인증 강도가 높습니다[18].

여러 서비스에서 동일한 지문을 쓴다고 해서 위험이 증가하지도 않으므로, 한 번 등록해두면 여러 용도로 활용할 수 있다는 장점도 있습니다. 반면 **단점과 한계**도 분명합니다. 앞서 언급했듯 **생체정보 유출 시 변경 불가**하다는 치명적인 단점이 있고, 높은 정확도를 유지하려면 센서 및 알고리즘 품질이 담보되어야 하므로 **전용 하드웨어 비용**이 수반됩니다. 또한 **오인식/오거부 문제**로 인해 일정 확률로 정당한 사용자가 거부되거나 타인이 통과하는 오류가 발생할 수 있으며, 이로 인해 **긴급 상황**(예: 지문 손상 시 대체 인증) 대응을 위한 백업 수단이 필요합니다. 운영 측면에서는 사용자의 **프라이버시 우려**도 단점으로 지적됩니다.

예컨대 지문과 얼굴 데이터는 민감정보로 분류되는데, 이를 기관이나 기업이 수집·보관하는 것에 대한 거부감과 규제 요건(GDPR 등)이 존재합니다. 이러한 이유로 FIDO2와 같이 **생체 템플릿을 클라우드에 절대 보내지 않는 로컬 매칭** 구조가 권장되고 있지만, 일부 원격 얼굴인증 서비스 등은 개인정보 처리 이슈를 안고 있습니다. 끝으로, **환경 요인**에 따라 성능 저하가 단점으로 작용하기도 합니다. 어두운 곳에서의 얼굴인식 오류, 건조한 겨울철 지문센서 오류 등은 사용자 불편을 초래할 수 있습니다.

요약하면, 생체인증은 **편의성과 보안** 측면에서 혁신적이지만 **변경 불가능한 민감정보 사용**이라는 태생적 한계를 가지고 있어, **다중요소 인증의 일부**로 활용하거나, **보조 수단**을 반드시 갖추는 식으로 단점을 보완해야 합니다[13]. 실제로 NIST도 생체인증은 **별도의 물리적 인증요소**(토큰 등)와 결합된 MFA의 일부로만 사용할 것을 권고하고 있습니다[19].

6) 글로벌 주요 표준 및 정책 흐름

생체인증 분야에서는 **국제 표준화 기구(ISO/IEC)**와 각국 정부 기관, 그리고 **FIDO 얼라이언스** 등이 중심이 되어 표준과 정책을 주도하고 있습니다. ISO/IEC JTC 1/SC 37 위원회는 지문, 얼굴, 홍채 등 생체 데이터의 저장 포맷(ISO 19794 시리즈), 성능 시험 기준(ISO 19795), 위조 공격 검출(ISO 30107) 등의 표준을 제정하여 글로벌 상호 운용성과 안전기준을 마련했습니다.

예를 들어 ISO/IEC 30107-3은 지문, 얼굴인식 장치가 갖춰야 할 **활성 감지 성능 지표**와 테스트 방법을 규정하고 있으며 최신 2023년 개정판에서는 **딥페이크 검출 기준** 등 새로운 공격 유형에 대응하는 지침도 추가되었습니다. 미국 NIST 역시 **디지털 인증 가이드라인(NIST SP 800-63B)**에서 생체인증 활용 시 지켜야 할 요건을 상세히 명시합니다. 해당 지침에 따르면 연방기관에서 생체인증을 사용할 때 **오인식률 1/1000 이하**를 충족해야 하고, 반드시 **다른 요소와 결합된 다중요소**로만 활용하며, 센서와 검증 서버 간 통신은 상호 인증된 암호 채널로 보호해야 합니다[5][9]. 또한 생체 인증 실패가 **5회 연속** 발생하면 지연 혹은 대체인증으로 전환하여 무작위 시도를 어렵게 만드는 등 **시도 제한 요구사항**도 존재합니다[20]. 한편 **FIDO 얼라이언스**는 웹과 모바일에서 생체인증 활용을 촉진하기 위해 FIDO UAF, FIDO2 등의 기술 표준을 수립하고 있습니다. 이 표준에서는 지문이나 얼굴인식으로 **사용자 기기 잠금해제 → 기기가 서버와 공개키 챌린지 인증**을 수행하도록 규정하고, 각 기기의 생체인증 모듈에 대해 **Biometric Certification** 프로그램으로 성능과 보안성(예: 위조 공격 탐지 능력)을 인증합니다.

정책 측면에서는, EU 일반개인정보보호법(GDPR)이 **생체정보를 민감정보로 분류**하여 엄격한 동의 및 보호조치를 요구함에 따라 기업들이 생체인증 도입 시 **템플릿 암호화, 로컬 저장** 등을 채택하도록 유도하고 있습니다. 일부 국가는 공공 신원확인에 생체인증을 의무화하거나(예: 인도 Aadhaar의 지문 인증), 여행객 출입국 심사에 전자여권 얼굴인식을 도입하는 등 **정부 주도 활용**도 활발합니다. 전반적으로 표준화 기구는 **상호운용성과 보안강화 프레임워크**를 제공하고, 정책입안자들은 **프라이버시와 보안 간 균형**을 맞추는 규제 환경을 조성하는 방향으로 움직이고 있습니다.

7) 향후 발전 전망 및 과제

생체인증 기술은 향후 **정확도 향상**과 **공격 대응력 강화**, **신규 바이오 지표 발굴** 측면에서 지속 발전할 것으로 전망됩니다. 딥러닝 기반의 지문/얼굴 판별 알고리즘 등 AI 기술 도입으로 매칭 정확도가 갈수록 높아지고 있어 **오인식률(FAR) 0.0001%** 수준의 고성능도 장기적으로 기대됩니다. 동시에, 정교해지는 위조 공격(예: 딥페이크 영상으로 얼굴인증 우회)에 대응하기 위해 **동작 기반 활성검증**이나 **3D 센싱** 등 **반(反)스푸핑 기술**이 고도화될 것입니다[14]. 예컨대 영상으로 눈동자 미세 떨림이나 맥박 신호를 포착해 진짜 사람인지 판별하는 기술, 지문인식에서 **혈류 패턴**을 감지하는 기술 등이 연구되고 있습니다. 이와 함께 **새로운 생체 특징**들도 인증에 활용될 전망입니다. 심전도(ECG)나 걸음걸이, 향후 뇌파나 근전도 등 각종 바이오리듬이 개인식별에 유용한 특징을 보여 학계 연구가 활발하며, 이미 일부 웨어러블 기기는 워치의 ECG로 사용자 인증을 시도하고 있습니다.

멀티모달 인증은 특히 중요한 방향성으로, 단일 생체정보의 한계를 극복하기 위해 두 가지 이상의 생체인증을 조합해 **정확도와 내성을 동시에 높이는** 방식이 각광받고 있습니다. 한편 **프라이버시 보호**와 **데이터 보안**에 대한 요구도 높아져, 템플릿을 암호화한 상태로 비교하는 **동형암호 기반 매칭**, 연합학습으로 중앙에 원본 데이터 없이 모델 개선, **취소가능(cancelable) 바이오메트릭스**처럼 유출 시 재발급할 수 있는 변형 기법 등도 연구 과제입니다[21]. 정책적으로는 개인 생체정보 남용을 막기 위한 법규들이 강화될 것이며, 기업들은 **최소**

한의 생체정보만 활용하고 **온디바이스 처리**를 지향하는 방향으로 기술을 발전시킬 것입니다[22]. 예를 들어, FIDO2의 **사이트별 키 분리** 설계처럼 생체인증도 각 서비스 간 사용자 식별자가 공유되지 않게 하는 등 프라이버시 고려가 필수가 될 것입니다. 향후 **연속 인증**이나 **위험 기반 인증**과 생체인증이 결합하여, 로그인 이후에도 사용자의 타이핑 습관이나 얼굴 확인으로 **동일 사용자 세션 지속 여부**를 감시하는 **지속적 보증** 모델도 현실화되고 있습니다. 궁극적으로 생체인증은 **편리하면서도 사용자를 안전하게 식별**하는 수단으로서 자리잡겠지만, **지속적인 공격 대비책 개발**, **사용자 동의와 신뢰 확보**, **윤리적 이슈 대응** 등이 지속 해결해야 할 과제로 남아있습니다.

다. 패스워드리스 인증 (Passwordless Authentication, 패스키)

1) 기술 개요 및 작동 원리

패스워드리스 인증은 말 그대로 사용자에게 **비밀번호 입력을 요구하지 않는** 인증 방식들의 총칭으로, 최근 FIDO2 **패스키(passkey)** 기술의 부상으로 주목받고 있습니다. 기존 비밀번호 기반 인증은 기억 또는 저장된 공유 비밀을 이용하므로 피싱, 비밀번호 재사용 등으로 인한 보안 취약점이 큰 반면, 패스워드리스 방식은 **공개키 암호 기반** 등의 기법을 활용해 **서버와 비밀 공유를 없애고**도 사용자 확인을 달성합니다[6].

가장 대표적인 패스워드리스 인증은 FIDO 얼라이언스가 주도한 **FIDO2/WebAuthn 표준**으로, 여기서는 사용자 각각의 디바이스(스마트폰, 보안키 등)가 서비스마다 고유한 **공개키-개인키 쌍**을 생성하여 인증에 이용됩니다[23][24]. **원리**는 간단합니다. 가입(Register) 시 사용자 기기가 해당 서비스 도메인에 연동된 **새 키 쌍**을 만들고 공개키를 서버에 전달해 계정에 연계시킵니다. 이후 로그인 시(server authentication), 서버는 기기(클라이언트)에게 난수 챌린지를 보내고, 사용자는 기기에 **생체인증이나 PIN 입력** 등 **로컬 언락**을 한 뒤 자신의 **개인키**로 챌린지에 서명하여 돌려줍니다[6].

서버는 미리 저장된 공개키로 서명을 검증함으로써 기기 보유자(사용자)임을

확인하고 로그인 세션을 부여합니다[25]. 이 과정에서 사용자 비밀(개인키)은 절대로 기기 밖으로 유출되지 않고, 서버도 공개키만 보유하므로 설령 서버 DB가 탈취되더라도 공격자는 공개키만으론 사용자를 가장할 수 없습니다[6]. 또한 서명 데이터에는 해당 사이트의 도메인 정보가 포함되어 **피싱 사이트**에서는 올바른 서명이 생성되지 않으므로 패스워드리스 인증은 **피싱 저항성**을 지닙니다[6].

Apple, Google, Microsoft 등이 도입한 **패스키(passkey)**란 이러한 FIDO2 공개키 인증을 **멀티 디바이스 간 동기화**까지 지원하도록 구현한 것으로, 사용자는 한 기기에서 생성한 패스키를 다른 자신의 기기들(예: 휴대폰과 노트북)에서 공유하여 이용할 수 있습니다[26]. 즉 패스키 환경에서는 하나의 계정에 대해 여러 기기에 분산된 개인키 백업을 가질 수 있어 기기 분실 시 복구를 도우며, **QR코드나 BLE 통신**으로 주변 기기의 패스키를 이용해 새로운 기기에 로그인하는 사용자 경험도 제공됩니다. 이처럼 패스워드리스 인증은 암호학적 키쌍과 **사용자 디바이스 신뢰**를 기반으로 작동하며, 궁극적으로 "기기가 곧 사용자 신원 증명" 역할을 수행하게 합니다.

2) 주요 구현 방식 및 프로토콜

패스워드리스 인증의 구현은 **웹표준과 플랫폼 연동** 측면에서 주로 FIDO2(WebAuthn + CTAP) 프로토콜을 따르고 있으나, 그 외에도 다양한 형태가 존재합니다. **FIDO2/WebAuthn**은 현재 월드와이드웹 컨소시엄(W3C) 표준으로 확정되어 모든 주요 브라우저와 OS에서 지원되며, 자산(device)을 소유한 사용자만이 응답할 수 있는 **도전-응답식(public-key challenge)** 인증을 웹 생태계에 통합한 것입니다[24][27]. FIDO2는 크게 **웹인증(WebAuthn) API**와 **클라이언트-인증기 프로토콜(CTAP)**로 구성되는데, WebAuthn은 브라우저와 웹서버 간 상호작용 표준이고, CTAP은 PC/모바일과 외장 보안키(예: YubiKey) 또는 플랫폼 내 안전모듈 간 통신을 담당합니다[28]. 이를 통해 사용자는 노트북에서 웹사이트에 로그인할 때, 노트북 내장 TPM/보안칩의 키나 USB/NFC 보안키의 키를 활용할 수 있습니다.

FIDO2 이외에 **전통적 패스워드리스 방식**도 존재합니다. 예를 들어 일회성 로그인 링크를 이메일로 보내는 **매직 링크(Magic Link)** 방식이나, 인증 앱 푸시 승인 방식, 인증코드(OTP)를 이용한 방식 등도 사용자에게 비밀번호 입력을 요구하지 않으므로 광의의 패스워드리스에 속합니다. 다만 이러한 방식들은 여전히 **사용자에게 공유형 코드를 전달**한다는 점에서 피싱에 취약할 수 있어, 궁극적인 방향은 FIDO2와 같은 공개키 기반으로 수렴되고 있습니다.

Apple, Google, Microsoft는 2022년 공동 발표를 통해 **패스키 상호운용**을 선언하고, iCloud 키체인이나 구글 계정으로 패스키를 자동 백업/동기화하여 **생태계 전반**에서 패스워드 없는 인증 경험을 구현하고 있습니다. 실제로 오늘날 Microsoft 계정이나 Google 계정은 사용자가 본인 기기의 화면 잠금(지문/얼굴)만 풀어도 로그인되며, 이는 내부적으로 FIDO2 패스키가 동작한 결과입니다. 한편 기업용 환경에선 Okta나 Duo 등 IAM 솔루션에서 **푸시 인증**(로그인 시 모바일로 승인을 보내 OK 누르면 로그인)이나 PKI 스마트카드 인증 등을 제공하여 비밀번호를 배제하고 있습니다. 이렇듯 프로토콜 관점에서 패스워드리스 구현은 **FIDO2 표준**이 사실상 자리잡았고, 추가로 각 서비스 시나리오에 맞게 **푸시 메시지, 일회용 링크** 등이 병용되고 있습니다.

3) 보안성 분석

패스워드리스 인증은 **비밀번호 기반 공격면**을 획기적으로 감소시킨다는 점에서 보안성이 매우 높습니다. 특히 **공개키 인증**을 활용하는 FIDO2 패스키의 경우 사용자 계정에는 더 이상 크랙 가능한 해시나 재사용 우려가 있는 비밀번호가 전혀 존재하지 않으므로, 데이터베이스 해킹이 발생해도 공격자는 공개키만 얻을 뿐 사용자 사칭에 쓸 수 있는 정보는 없습니다[6]. 또한 패스워드리스 방식은 본질적으로 **피싱 공격에 탁월한 저항성**을 가집니다. 예컨대 사용자가 가짜 피싱 사이트에 속아 들어가더라도, WebAuthn 인증기는 브라우저가 제공하는 **오리진 식별**을 통해 **서버 도메인 불일치 시 서명생성을 거부**하므로 공격자가 인증토큰을 가로채거나 위조할 수 없습니다[6].

실제 연구에서도 FIDO2 패스워드리스는 기존 비밀번호 대비 **보안성과 사용편**

의성 둘 다 우수함이 검증되었습니다[27]. Passkey 구현에서 서버와 주고받는 서명 데이터는 1회용 챌린지에 대한 응답으로만 쓰이므로 재전송 공격이 불가능하며, 서명에는 **사용자 인증기 장치의 인증서(attestation)**를 포함할 수 있어 서버가 **인증기 신뢰도**를 검증(예: 보안칩 탑재 여부 확인)하는 것도 가능합니다. 요컨대 패스워드리스는 구조적으로 **중간자 공격(MitM)**이나 **피싱**, **크리덴셜 유출**을 막는 강인한 설계를 지닙니다[6].

물론 안전한 구현과 사용을 위한 고려사항도 있습니다. 패스워드리스를 구현하는 핵심은 **사용자 단말의 안전성**입니다. 만약 사용자의 패스키가 저장된 스마트폰이 탈취되었을 경우, 공격자는 추가 인증(예: 화면 잠금 PIN)을 우회해 기기 잠금을 해제할 수 있다면 그 기기의 패스키로 서비스 로그인을 시도할 수 있습니다. 이에 대비해 대부분의 구현은 **기기 잠금 해제**를 패스키 사용의 **전제조건**으로 요구합니다. 예를 들어 WebAuthn은 기본적으로 **사용자 검증(User Verification)** 절차(지문 등)를 거쳐야 서명이 완료되도록 규정하고 있습니다[29]. 이는 분실 기기가 공격에 악용될 위험을 낮춰주지만, 반대로 사용자가 기기와 연동된 인증 수단을 모두 잃어버렸을 때 **계정 복구 문제**가 발생하는 단점도 있습니다. 계정 복구를 위해 서비스별로 **별도 백업 코드**나 **비상 연락처**를 등록하도록 하는데, 이 과정에서 다시금 사회공학 공격의 여지가 생길 수 있습니다.

최근 연구에서는 패스키 도입시 **계정 복구 및 기기변경 절차**가 가장 큰 보안 취약점 겸 채택 장벽으로 지목되기도 했습니다[30]. 이 밖에 패스워드리스 인증은 사용자 단말에서 동작하므로 **멀웨어에 대한 방어**도 중요한 이슈입니다. 만약 사용자 PC나 스마트폰이 루팅되어 악성코드가 패스키 서명 과정을 가로챌 수 있다면 위험할 수 있으므로, 최신 운영체제는 안전실행환경(TEE)이나 보안칩(e.g. Android Keystore, Apple Secure Enclave)을 통해 키를 격리 관리하고 서명 연산을 보호합니다.

4) 활용 사례 (산업별 적용 현황)

패스워드리스 인증은 최근 대형 IT 서비스와 **모바일·PC 운영체제 차원**에서 본

격 도입되기 시작했습니다. **일반 소비자 서비스** 중에서는 마이크로소프트(Microsoft)가 자사 계정 로그인에 패스워드리스 옵션을 2021년 도입하여, 사용자들이 **MS Authenticator 앱 승인, Windows Hello 얼굴인증, FIDO2 보안키, SMS 코드** 중 하나로만 로그인하고 비밀번호는 삭제할 수 있도록 하였습니다. 구글도 2023년 자사 구글 계정에 패스키 지원을 추가하여, 안드로이드폰 화면잠금이나 iOS Face ID로 구글 로그인을 완료할 수 있게 되었습니다. 애플은 iOS/macOS의 **아이클라우드 키체인**을 통해 패스키를 기기 간 동기화하고, 사파리 브라우저에서 **"패스키로 로그인"** 기능을 제공하면서, Dropbox, PayPal, eBay, 서비스들이 잇따라 패스키 로그인을 지원하도록 유도했습니다. 이처럼 B2C 영역에서 주요 웹 플랫폼들이 패스워드리스 생태계를 형성 중이며, 2023년 FIDO 얼라이언스 조사에 따르면 일반 사용자들도 패스키 방식에 대한 선호도가 높아지는 추세입니다[31].

기업 및 산업 분야에서도 패스워드리스 도입이 활발합니다. 예를 들어 은행 및 금융기관들은 내부 직원들이 스마트카드나 FIDO2 보안키로 로그인하도록 전환하여 피싱으로부터 **업무망 계정**을 보호하고 있습니다. 클라우드 업무환경의 증가로, Okta, Duo, Azure AD 같은 기업용 SSO 솔루션들은 모두 FIDO2 인증을 지원하며 패스워드 없는 사용자 포털 로그인을 구현했습니다. 개발자 커뮤니티(GitHub 등)나 협업툴(Slack, Atlassian) 등도 패스키 지원을 발표하여, 점차 **전 산업에서 패스워드 제거** 움직임이 보입니다.

다만 **레거시 시스템**이 많은 제조, 의료 등 일부 분야는 기존 애플리케이션이 패스워드 전제하에 설계된 경우가 많아 전환이 더뎠습니다. 하지만 Windows 도메인 로그인 등에 패스워드리스 기술이 적용되고(Windows Hello 기업배포) FIDO2를 활용한 **레거시 랩탑 로그온** 솔루션도 등장하면서 이런 장벽도 해소되는 추세입니다.

요약하면, 현재 패스워드리스 인증은 **개인사용자용 빅테크 서비스와 기업 IAM 영역 모두에서 빠르게 확산** 중이며, FIDO 얼라이언스의 2023년 설문에서는 응답 기업의 58%가 향후 패스워드리스 기술을 도입·확장할 계획이라고 밝혔습니다[32]. 이는 보안 강화뿐 아니라 사용자 지원 비용(비밀번호 재설정 등) 절

감 효과도 크기에 금융, IT, 공공 등 다방면에서 채택이 늘어나는 것으로 분석됩니다.

5) 기술적/운영상 장단점

패스워드리스 인증의 **장점**은 명확합니다.

첫째, **뛰어난 보안성**입니다. 패스워드가 없으므로 피싱이나 크리덴셜 탈취, 무차별 대입 등의 기존 계정 공격이 거의 원천 차단됩니다[6]. 특히 패스키는 **도메인 바인딩된 공개키**만 서버에 있어 유출되어도 악용될 수 없고, 재사용 위험도 없으므로 하나의 사고로 다수 서비스 계정이 동시에 털리는 **도미노 해킹** 가능성이 사라집니다. 둘째, **사용 편의성 향상**입니다. 한번 기기에 패스키(또는 인증앱)가 설정되면 사용자는 비밀번호를 기억하거나 주기적으로 변경할 필요 없이 **간편인증(생체, PIN, 푸시)**으로 로그인하므로 로그인 과정이 빨라지고 스트레스가 줄어듭니다. 실제 FIDO2 사용자 연구에서 **로그인 성공률과 만족도가 비밀번호 대비 높다**는 결과가 보고되었습니다[27]. 셋째, **운영 효율성** 측면에서도 헬프데스크의 비밀번호 초기화 업무 부담을 크게 덜고, 비밀번호 정책 관리에 들던 노력이 감소합니다.

그러나 패스워드리스의 **단점과 과제**도 있습니다. 우선 **호환성과 전환 비용 문제**가 있습니다. 기존 수많은 시스템들이 비밀번호 체계를 전제로 구축되어 있어 이를 패스워드리스로 전환하려면 상당한 개발·통합 작업이 필요합니다. 웹 서비스들은 FIDO2 지원을 위해 웹Authn API 구현 및 UI 변경이 필요하고, 사용자의 브라우저/OS 환경도 최신이어야 합니다. 둘째, **계정 복구 및 기기 분실 시 처리**가 어렵습니다. 패스워드가 없으므로, 사용자가 새 기기로 바꾸거나 기존 기기를 모두 분실하면 본인임을 증명할 마스터 비밀번호 같은 게 없습니다. 서비스들은 이를 위해 **별도 백업 인증 수단**(예: 이메일 OTP, 백업 코드)을 준비하지만, 이는 다시 피싱 위험이 있는 임시 비밀번호 역할을 하므로 전체 보안성을 저해하거나 사용자 경험을 복잡하게 만들 수 있습니다[33]. 셋째, **멀티 기기/플랫폼 간 상호운용** 이슈입니다. 예컨대 사용자가 iOS에서 만든 패스키를 안드로이드 기기에서도 쓰려면 QR코드 촬영 등 교차 인증 절차가 필요해 다

소 번거롭습니다.

패스키 동기화는 일반적으로 **동일 플랫폼 계정 생태계 내**에서만 원활하므로 (예: iCloud, Google 계정별), 서로 다른 생태계 간 이동 시 **전송 절차의 표준화**가 더 요구됩니다. 넷째, **사용자 습관 및 문화적 장벽**입니다. 비밀번호에 익숙한 많은 사용자들에게 패스워드리스 개념은 낯설 수 있고, 일부는 오히려 "로그인에 아무 것도 안 치니 불안하다"는 심리도 있습니다. 또한 IT 관리자 입장에서도 초기에는 **새 기술에 대한 신뢰성** 우려와 배포 부담이 있을 수 있습니다[33].

마지막으로, **장치 의존성** 문제도 있습니다. 패스워드리스는 결국 **사용자 소유 디바이스**가 필요하므로, 만약 사용자가 해당 디바이스를 사용할 수 없는 환경(공용PC 등)에서는 번거로운 대체 절차가 필요합니다. 예컨대 공공 PC에서 작업하려는데 FIDO2 보안키도 없고 폰도 배터리가 없다면 로그인에 어려움이 생길 수 있습니다. 이러한 시나리오를 위해 FIDO2는 **기기 연결 패스키(필요시 휴대폰으로 QR인증)** 등의 방안을 제공하지만 여전히 제약이 존재합니다.

정리하면 패스워드리스 인증은 **보안과 사용자 경험 양면에서 큰 개선**을 이루는 기술이나, **계정 복구 문제, 이질적인 플랫폼 환경, 사용자 교육 및 신뢰 확보** 등의 과제를 안고 있습니다. **다중 계층 보안** 측면에서도, 패스워드리스 도입 시에도 관리콘솔 접속 등 일부 매우 민감한 액션에는 추가 MFA를 요구하는 등 **리스크 기반 보완**이 권고되고 있습니다.

6) 글로벌 주요 표준 및 정책 흐름

패스워드리스 인증 분야에서는 **FIDO 얼라이언스**와 **W3C**가 핵심 표준화 역할을 수행하고 있으며, 각국 정부의 보안 지침도 이를 수용하는 방향입니다. FIDO 얼라이언스는 2014년 U2F(2단계 인증용)를 시작으로 2015년 UAF(패스워드 대체)를 제정했고, 2018년 **FIDO2**를 완성하여 웹 표준화 기구 W3C에서 **WebAuthn Recommendation**으로 공식 승인받았습니다. 그 결과 2020년대 들어 Chrome, Firefox, Safari, Edge 등 모든 주요 브라우저가 WebAuthn을

구현하고, Windows, Android, iOS, Linux 등 OS도 FIDO2 호환 API를 시스템에 통합하였습니다[27]. 현재 FIDO2(WebAuthn) 인증은 **Level AAL3** 고급 인증 수단으로 미국 연방에서 인정되고 있으며, NIST SP 800-63B는 피싱 저항성과 인증기 기기 인증 특성을 갖춘 FIDO U2F/2를 **권장 구현**으로 언급하고 있습니다[34].

미국 백악관은 2022년 「제로 트러스트 전략」에서 정부기관들에 **MFA 구현**을 지시하면서, 특히 **피싱에 안전한 MFA**를 도입하라고 강조했는데 이는 사실상 FIDO2 패스워드리스를 의미합니다. 금융 분야에서도, 미 연방금융기관검사협의회(FFIEC)는 2021년 인증 가이드라인에서 **리스크 기반의 강인한 MFA**를 권고하며 전통적 비밀번호+OTP 대신 **크립토그래픽 MFA**로의 이행을 촉구했습니다[35]. 유럽연합의 PSD2(지불결제서비스 지침)도 전자결제에 **강력고객인증(SCA)**를 의무화하면서 생체인증+보안키 등 비밀번호 미사용 방식을 사실상 요구하고 있고[36], FIDO 얼라이언스도 유럽은행협회(EBA)에 패스워드리스 도입 권고 의견을 제출한 바 있습니다. 한편 표준 상호운용을 위해 2022년 FIDO 얼라이언스, Apple, Google, Microsoft는 **다중 기기 패스키 통합** 협력을 발표하여 플랫폼 간 패스키 이동성 향상을 약속했습니다. 이러한 업계 주도의 움직임에 발맞춰 국내외 정책들도 패스워드리스 기술을 **인증수단 선택지**에 포함시키고 있습니다.

국내 전자서명법 등에서도 FIDO 생체인증을 공인인증서 대체 수단으로 활용할 수 있게 되었고, 일본 정부도 공공 웹사이트 로그인에 FIDO2를 도입하는 시범을 진행 중입니다. 전 세계적으로 "패스워드 종말"에 대한 공감대가 커지면서, 표준과 정책은 **FIDO2 패스키**를 중심으로 사용자계정 보호 수준을 높이는 방향으로 수렴하고 있다고 볼 수 있습니다[27].

7) 향후 발전 전망 및 과제

패스워드리스 인증은 향후 **디지털 인증의 주류**로 자리잡을 가능성이 높습니다. 이미 애플, 구글, MS의 생태계 통합이 이루어짐에 따라 수억 사용자가 패스키를 사용할 기반이 마련되었고, 앞으로 더 많은 웹서비스들이 지원을 늘려갈 것

입니다. 향후 전망으로는 **패스워드 완전 퇴출**이 현실화되어, 일반 사용자들이 비밀번호를 아예 취급하지 않게 되는 변화를 기대할 수 있습니다. 이를 위해 남은 과제는 **플랫폼 간 매끄러운 사용자 경험**입니다. 예컨대 업무용 PC에 개인 스마트폰 패스키로 로그인하거나, 가족 간 계정 공유 시 권한 위임 등 아직 풀리지 않은 UX 시나리오들이 있습니다. FIDO 얼라이언스는 이러한 문제를 해결하기 위해 BLE 및 QR코드 기반 **교차 단말 패스키 사용**, **보조 사용자 인증 공유** 등의 확장 기능을 개발 중입니다. 또한 **계정 복구 문제**에 대해서는, 여러 장치를 소유한 사용자의 경우 자동으로 **여러 장치에 키를 분산**저장하거나 클라우드에 **암호화 백업** 후 신원증명으로 복원하는 방법 등이 논의되고 있습니다[37].

보안 측면에서는, 현재 선택지로 남아있는 SMS OTP, 이메일 링크 같은 취약한 패스워드리스 방식들을 **장기적으로 퇴출**하고 모두 공개키 기반으로 전환하는 것이 목표입니다. 이를 위해 중소 서비스나 내부 시스템에서도 손쉽게 FIDO2를 활용하도록 **오픈 소스 라이브러리**와 **서비스형 인증(API)**가 확대될 것입니다. 또한 **하드웨어 보안모듈**의 발전으로, 저가 기기에도 TPM/보안칩이 탑재돼 누구나 안전한 패스키를 보관할 수 있게 되고, 장차 **양자 내성 알고리즘**이 필요한 시점에는 패스키 알고리즘도 Kyber 등의 PQC로 업그레이드될 것입니다. **과제**로 남는 부분은 **사용자 인식 제고**와 **레거시 시스템 호환**입니다. 많은 일반인들이 여전히 비밀번호에 익숙하기에 패스워드리스의 편리함과 안전함에 대한 교육이 필요합니다. 기업 측면에서도 오래된 시스템과의 연동, 법규상의 전자서명 인정 여부 등 풀어야 할 정책적 이슈가 있습니다. 하지만 전반적인 흐름은 패스워드리스로 기울었고, FIDO 얼라이언스가 매년 **대규모 상호운용 테스트 행사**를 개최하며 기술 성숙도를 높이고 있는 만큼, 향후 5~10년 내에 패스워드 기반 로그인은 급격히 감소하고 패스키 등 **차세대 인증**이 사실상 표준이 될 전망입니다[27].

라. 블록체인 기반 인증 (Decentralized Identity)

1) 기술 개요 및 작동 원리

블록체인 기반 인증은 **신원 관리와 인증과정에 블록체인/분산원장 기술을 활용**하여 중앙 기관 없이 사용자 스스로 신원 증명을 제어할 수 있게 하는 접근 방식입니다. 종종 **탈중앙 신원(DID, Decentralized ID)** 또는 **자기주권 신원(SSI, Self-Sovereign Identity)**이라는 용어로도 불리며, 핵심 개념은 사용자에게 발급된 디지털 신원 정보를 중앙 서버가 아닌 **분산 네트워크에 등록**하고, 검증 시 분산원장에 기록된 신뢰 루트를 활용한다는 것입니다.

예를 들어 사용자는 정부로부터 발급받은 디지털 자격증명을 자신의 디지털 지갑에 보관하고, 서비스에 로그인할 때 해당 증명서에 서명된 **증명(assertion)**만 제시하여 본인을 증명할 수 있습니다[38]. 이때 서비스 측은 블록체인에 공개된 **발급자(정부)의 공개키**와 **신원 식별자(DID)**를 조회하여 증명의 유효성을 검증하므로, 중앙집중 DB나 중개 신원 제공자 없이도 신뢰 검증이 가능합니다[39]. 구체적으로, W3C가 정립한 **탈중앙식별자(DID)** 표준에 따르면 DID는 예컨대 did:example:123456 형태의 전역 고유 식별자이며, 해당 DID에 대한 공개키, 서비스 엔드포인트 등의 메타데이터가 **블록체인** 또는 분산 네트워크에 **DID Document**로 저장됩니다[40]. 사용자는 자신의 개인키로 해당 DID에 속하는 메시지에 서명함으로써 **자신이 그 DID의 주체임**을 증명할 수 있고, 검증자는 체인 상의 DID Document로 서명을 검증합니다. 이러한 구조에서 블록체인은 일종의 **공개 키 디렉토리(PKI)** 역할을 수행하지만, 전통 PKI와 달리 **루트 인증기관**이 아니라 **분산 합의**로 무결성을 보장한다는 차이가 있습니다[41].

요약하면 블록체인 기반 인증에서는 **사용자 → 증명서 발급기관(신뢰주체) → 서비스 제공자** 간에 **분산원장**을 통한 신뢰 연결 고리가 형성됩니다[42]. 기술적으로는 **블록체인**에는 개인정보 자체가 저장되지 않고, 주로 DID와 공개키, 해시값, 폐기목록 등의 **메타데이터**만 올라갑니다[43].

실제 인증 흐름에서는 사용자가 자신의 **디지털 자격 증명(Verifiable Credential)**을 들고 있다가, 서비스에 **제로지식증명(ZKP)** 등으로 필요한 속성만 증명하거나, **디지털 서명**으로 로그인 요청을 승인하는 식입니다[44].

블록체인은 이러한 증명서의 진본 여부(발행자의 서명 검증), 폐기 여부, 사용자 DID의 유효성 등을 **탈중앙 신뢰 원장**으로 보장해주는 역할을 합니다[42]. 대표적인 예가 **로그인 위드 이더리움**과 같은 방식으로, 사용자가 소유한 블록체인 계정(이더리움 주소)으로 서비스에 **메시지 서명**하여 로그인하면, 서비스는 이더리움 네트워크에 해당 주소의 존재와 거래내역(또는 DID Document)을 조회함으로써 별도 중앙 ID 없이도 인증을 수행합니다. 정리하면 블록체인 기반 인증은 중앙 ID 제공자(OAuth나 SSO IdP 등)를 없애고 **분산 합의 네트워크가 신뢰의 앵커(anchor)**가 되는 모델로, 사용자에게 **신원 정보에 대한 자율권**을 부여하고자 하는 패러다임입니다[39].

2) 주요 구현 방식 및 프로토콜

블록체인 기반 인증의 구현에는 여러 오픈 표준과 플랫폼들이 존재합니다. **W3C Decentralized Identifiers(DID) 표준**은 다양한 블록체인/원장에서 DID를 구현할 수 있도록 추상 모델을 제공합니다[45]. 현재 **이더리움, Hyperledger Indy, 솔리드 등** 여러 원장에서 DID 메소드 규격들이 나와 있습니다 (예: did:ethr:…, did:sov:… 등). DID는 **공개키 쌍**과 연결되며 DID Document에는 해당 공개키와 인증/암호화 방법, 및 **서비스 엔드포인트**가 명시됩니다[46]. 이를 등록/갱신하는 행위는 각 원장의 트랜잭션으로 수행되어 **변조 불가능한 형태**로 기록됩니다. 사용자 측에서는 **디지털 신원 지갑(identity wallet)** 앱이나 프로그램을 이용해 자신의 DID와 자격증명을 관리합니다[38].

W3C의 또 다른 표준인 **Verifiable Credentials(VC)**은 발행자(Issuer)가 특정 주체(Subject)에 대해 서명한 **속성 주장**들을 표현하는 JSON-LD 기반 데이터 모델로, 예컨대 대학교가 "홍길동은 컴퓨터공학 석사학위 취득"이라는 VC를 발행해주면 홍길동은 이를 지갑에 저장합니다. 인증을 요구하는 서비스(Verifier)는 사용자가 제시하는 VC와 그 서명을 **발행자의 DID 공개키로 검증**

하고, 필요시 블록체인에 등록된 **철회 목록(Revocation Registry)**도 확인하여 해당 VC가 취소되지 않았는지 검사합니다[47][48]. 이 모든 과정을 거칠 때도 개인정보는 사용자가 직접 제시하는 VC에만 포함되고, 블록체인에는 해시값이나 폐기 토큰만 기록되므로 프라이버시를 지킬 수 있습니다[42].

프로토콜 측면에서는 **DID Comm**(Decentralized ID Communication) 같은 표준을 통해 지갑과 서비스 간 **P2P 메시지 교환**이 이뤄집니다[49]. 예컨대 사용자가 웹사이트에 로그인하려 하면, 해당 사이트가 사용자의 DID 엔드포인트로 인증 요청 메시지를 보내고, 사용자는 지갑앱에서 이를 승인하면 서명 토큰을 되돌려주는 식입니다. 또한 **JSON 웹 토큰(JWT)**이나 **JSON-LD 기반 ZKP (BBS+ 서명)** 등이 VC 전달 및 증명에 활용되기도 합니다[44].

플랫폼 구현으로는 Sovrin/Indy 기반 **Hyperledger Aries/Ursa** 프레임워크, Microsoft의 ION (Bitcoin 기반 DID), Ethereum 기반의 uPort (현재는 DID:ethr) 등이 대표적입니다. Hyperledger Indy에서는 특정 퍼미션드 블록체인에 DID와 public DID(Document)를 올리고, 익명 증명 기술(클라메 증명)로 속성 증명을 수행하는 **Hyperledger Aries protocols**이 제공됩니다.

이더리움계 DID는 퍼블릭 이더리움 체인에 DID Document 해시를 올리고 외부 저장소(IPFS 등)에 실데이터를 저장하는 방식 등으로 구현됩니다. 또한 Web3 영역에선 **지갑 서명 로그인**이 널리 쓰이는데, 예컨대 **EIP-4361 표준 ("Sign-In with Ethereum")**은 서비스가 사용자에게 nonce가 포함된 메시지를 보내면 사용자가 자신의 이더리움 주소로 서명해 응답하고, 서비스는 **해당 주소가 예상된 사용자 DID와 일치하는지** 확인하는 절차를 정의합니다. 이처럼 여러 구현이 존재하지만, 모두 **분산원장에 등재된 공개키**와 **사용자 개인키 서명**이라는 큰 맥락은 동일합니다.

3) 보안성 분석 (공격 벡터, 취약점 대응)

블록체인 기반 인증은 **중앙 신원정보 DB가 없다는 점**에서 전통적 ID 관리보다 **단일 실패점(single point of failure)**을 줄이고, 보안성을 높일 잠재력을 지

닙니다[41].

중앙 IdP나 데이터베이스가 없으므로 그 자체가 해킹당해 수백만 사용자 비밀이 유출되는 사고를 예방할 수 있습니다[50]. 대신 각각의 DID와 자격증명은 **분산 네트워크의 합의**로 무결성이 유지되고, 조작이나 삭제가 어렵기 때문에 공격자가 신원 정보를 위조하거나 임의 폐기하는 것이 거의 불가능합니다[39]. 또한 사용자가 자신의 증명서를 제시할 때 **선택적 공개(Selective Disclosure)**나 **영지식 증명(ZKP)** 등을 활용하면, 검증자에게 최소한의 정보만 넘겨도 인증이 가능하므로 **불필요한 개인정보 노출을 크게 줄이는** 보안상 이점이 있습니다[44]. 예컨대, 블록체인 기반 인증 시스템에서는 "이 사람이 성인이다"라는 사실만 증명하고 실제 생년월일은 공개하지 않을 수 있습니다. 이러한 **프라이버시 강화 인증**은 기존 중앙 인증에서는 어려웠던 부분입니다[51].

한편 보안성 확보를 위해 해결해야 할 부분도 있습니다.

첫째, **사용자 개인키 관리** 문제가 있습니다. 중앙 서버가 없어 **각 사용자가 자신의 키를 온전히 책임**져야 하므로, 만약 사용자가 DID 개인키를 분실하거나 도난당하면 그 계정을 제어할 방법이 없습니다. 이를 보완하기 위해 사회적 복구(social recovery)나 다중서명 지갑 등 기법이 논의되지만, 여전히 **키 분실=신원 상실**이라는 위험이 상존합니다[52]. 둘째, **프라이버시의 역설**이 있습니다. 원장에 기록된 거래는 공개되는 경우가 많아, DID 사용 내역이 고의치 않게 **사용자 행동 추적**에 악용될 수 있습니다. 이를 막기 위해 **비식별 DID**(pairwise DID: 상대마다 다른 DID 사용)나 **영지식 증명**으로 최대한 링크성을 없애도록 설계하지만, 실사용에서 완벽히 추적 불가를 담보하기는 어렵습니다. 셋째, 블록체인의 **합의 안정성**과 **스마트컨트랙트 보안** 이슈입니다. 신원 증명에 쓰이는 블록체인 자체가 51% 공격 등에 흔들리면, 공격자가 DID Document를 위조 등록하거나 삭제하는 것도 이론적으로는 가능합니다(특히 소규모 퍼블릭체인일 경우). 또한 스마트 컨트랙트 기반으로 DID 레지스트리를 운용할 경우 컨트랙트 버그가 보안 위험이 될 수 있습니다. 따라서 주요 DID 네트워크는 오디팅과 거버넌스로 안정성을 담보하려 노력합니다. 넷째, **인증 지연 및 성능** 문제입니다.

블록체인은 트랜잭션 합의에 시간이 걸리므로, 실시간 대량 인증에 직접적으로 체인을 매번 사용하면 지연이 발생할 수 있습니다. 이를 해결하려고 대부분 **오프체인 캐싱** 또는 **1회 등록 후 조회만**으로 동작하게 디자인되어, 실시간 속도는 크게 문제되지 않도록 하지만, 여전히 전통 DB보다 조회 지연이 있는 편입니다.

공격 벡터로는 사용자 지갑/키 탈취, 가장 공격 등이 있을 수 있습니다. 예컨대 피싱이나 악성앱으로 사용자의 신원 지갑(모바일 앱)을 탈취하면, 중앙 통제가 없으므로 공격자는 사용자 DID로 마음대로 서명해서 **신원 사칭**이 가능합니다. 이런 점에서 사용자 측 단말 보안이 중요하고, 하드웨어 지갑이나 안전한 저장소를 사용하는 것이 권장됩니다. 또한 **발급자 자체의 신뢰성** 문제가 있습니다. 분산 신원에서도 결국 어떤 권위자가 VC를 발급해줘야 하는데, 만약 발급자가 잘못된 정보를 담은 VC를 서명해주면 검증자는 속을 수밖에 없습니다. 즉 **"쓰레기를 분산해봤자 쓰레기"**라는 지적이 있으며[53], 아무리 기술이 좋아도 **초기 신원 증명 단계(예: 정부가 사용자를 대면 확인하고 DID를 발급)**의 신뢰가 확보되지 않으면 무의미합니다. 따라서 블록체인 인증 도입 시 **신뢰 프레임워크**(어떤 기관이 어떤 정보에 권위를 갖는지 협약)가 병행되어야 합니다.

4) 활용 사례 (산업별 적용 현황)

블록체인 기반 인증은 아직 신흥 기술이지만, 전 세계적으로 다양한 **파일럿 프로젝트와 특화 분야 활용**이 진행되고 있습니다. **공공 부문**에서는 유럽연합이 주도하는 **EBSI(European Blockchain Services Infrastructure)**에서 **유럽 디지털 신원 지갑(EUDI Wallet)**을 개발하면서, 학력증명, 의료면허 등 **검증 가능한 자격증명(VC)**을 블록체인 상에서 유통하는 실험이 이뤄지고 있습니다. 특히 벨기에, 스페인 등 일부 국가에서는 주민ID를 DID 형태로 발급하여 행정 서비스 로그인에 활용하는 시범사례가 보고되었습니다. **금융권**에서도 **KYC 간소화**나 **고객 신원 공유** 목적으로 SSI 개념을 도입하는 움직임이 있습니다. 예를 들어 캐나다의 Verified.Me 시스템이나 UAE의 KYC 블록체인 콘소시엄은 은행들이 고객의 신원 정보를 분산 원장에 올려 공동 활용하는 구조로, 고객이

일일이 본인확인을 반복하지 않아도 되게 합니다. **기업 및 산업 분야**에서는, LG CNS, 삼성 SDS 등 IT기업들이 DID/VC를 임직원 출입 및 문서 인증에 활용하는 사례가 있습니다. 예컨대 LG CNS는 사내 DID 플랫폼으로 출장 시 호텔 체크인, 건물 출입 등을 한 번 발급한 VC로 처리하는 등 **사내 인증 간소화**를 시연했습니다. **교육 분야**에서는 대학 졸업장이나 성적 증명서를 VC로 발급해 주는 일이 늘고 있습니다.

MIT, 동국대 등 몇몇 대학은 졸업증을 블록체인에 저장해, 취업 시 졸업생이 간단히 증명서를 제출하면 기업이 블록체인 검증으로 확인하도록 했습니다[54]. **의료 분야**에서도 의료인 면허증, 환자 동의서 등을 VC로 발행해 무단 위변조를 방지하는 파일럿이 진행되었습니다.

한편 **웹3 및 탈중앙 웹** 분야에서는 블록체인 계정 자체를 아이덴티티로 사용하는 일이 흔합니다. 암호화폐 지갑 주소로 디앱(DApp)에 로그인하고, NFT 토큰 소유 여부로 특정 커뮤니티 접근을 허용하는 등, **지갑 서명 기반 인증**이 DeFi, NFT 마켓플레이스 등에서 표준으로 자리잡았습니다. 예컨대 **Ethereum Name Service(ENS)**는 사용자가 alex.eth 같은 닉네임을 이더리움 주소에 연결해 디지털ID로 쓰게 하며, 이것을 소셜 프로필로 활용하는 사례도 있습니다. 또한 **SOUL 바운드 토큰(SBT)**처럼 양도 불가능한 인증서 토큰을 개인 지갑에 발행하여, 학위증서나 경력 증명을 나타내는 시도도 있습니다. **물류/산업 IoT 분야**에서도 부품이나 제품에 DID를 부여해 이력 추적 및 인증에 활용하는 연구가 있습니다.

이처럼 아직 대중화 초기 단계지만, **정부 ID, 금융 KYC, 교육 인증, 기업 내부 인증, 웹3 로그인** 등에서 블록체인 기반 분산ID 개념이 점진적으로 도입되고 있습니다. 마이크로소프트, IBM, 에버님 등 기술 기업들과 각국 정부가 컨소시엄을 만들어 **역할별 표준 인터페이스**를 정의하고 상호운용을 테스트하는 등, 미래의 신원 인프라로 발전시키기 위한 노력이 이어지고 있습니다[55].

5) 기술적/운영상 장단점

블록체인 기반 인증의 **장점**은 **자율성과 확장성, 보안성** 측면에서 찾아볼 수 있습니다. 먼저 **사용자 자율성(Self-Sovereignty)**이 가장 큰 강점입니다. 사용자는 자신의 신원 정보와 인증 수단을 스스로 소유·통제하며, 어떤 정보가 누구에게 가는지 **세밀하게 제어**할 수 있습니다[56][22]. 이는 중앙기관이 모든 신원 정보를 관리하는 기존 체계에 비해 프라이버시 보호와 이용자 주권 측면에서 혁신적입니다. 둘째, **보안 및 신뢰 분산**입니다. 한 곳의 DB가 뚫려도 전체 신원체계가 무너지는 일이 없고, 여러 노드가 합의하여 신뢰를 검증하므로 **단일 실패점이 제거**됩니다[41]. 또한 암호화된 원장 기록과 서명 검증으로 데이터 위변조를 방지해 **무결성 보장**이 탁월합니다. 셋째, **범용 상호운용성**입니다. 표준에 따르면 DID/VC는 국경이나 조직 경계를 넘어 쓸 수 있으므로, 한 번 발급받은 디지털 신분증을 다양한 서비스에서 재활용할 수 있습니다[57]. 예컨대 운전면허 VC 하나로 공항 체크인, 렌터카 대여 등 여러 곳에서 증명 가능하며, 대학 졸업 VC 하나로 국내외 모든 채용사이트에 증빙하는 식입니다. 이는 **중복 인증 절차 제거**와 **프로세스 효율화**로 이어집니다[38]. 넷째, **데이터 주권과 투명성**입니다. 원장에 기록된 신원 확인 이벤트는 필요시 감사 가능하고, 사용자의 동의 하에만 데이터 제공이 이뤄지므로 GDPR 등의 **프라이버시 규제 준수**에도 용이합니다[22]. 마지막으로, **경제적 효율성** 측면에서 중장기적으로 ID 중개 비용을 낮출 잠재력도 있습니다. 기존에는 각 서비스가 중복으로 본인 확인을 수행하던 것을 한 번의 검증으로 대신하거나, 거대 IdP에 지불하던 비용을 절감할 수 있기 때문입니다.

반면, 아직 해결해야 할 **단점과 과제**도 존재합니다.

첫째는 **복잡성과 초기 비용**입니다. 기술이 상당히 복잡하여 일반 사용자에게 이해시키기 어렵고, 지갑 관리나 키 복구 등의 UX가 다소 난이도가 있습니다. 조직 입장에서도 현 시스템에 통합하려면 러닝커브가 있으며, 관련 인프라(노드 운영 등) 구축 비용이 듭니다. 둘째, **성숙도 부족과 표준 난립**입니다. DID/VC 표준은 나왔지만 구현 메서드가 여러 개라 상호운용성이 100% 확립되지 않았습니다. 어떤 네트워크의 DID를 다른 네트워크 검증자가 신뢰할지

거버넌스 이슈도 있습니다. 아직 **하나의 통일된 글로벌 신뢰망**이 아니므로 과도기적 혼란이 단점입니다[52]. 셋째, 앞서 언급한 **키 분실 위험**과 **사용자 책임 증가**입니다. 중앙기관이 없으니 편리한 비밀번호 재설정 같은 것도 없고, 전적으로 개인이 보안에 신경써야 하므로 사용자 측에 부담이 커집니다. 이는 **보편적 채택의 걸림돌**이 될 수 있습니다. 넷째, **규제와 법적 승인** 문제입니다. 국가 신원 체계는 법률과 강하게 연동되는데, 탈중앙 ID의 법적 효력이나 책임소재가 명확치 않습니다. 예컨대 누가 발급자이고 잘못된 VC 발급 시 누가 책임지는지 등이 정립되어야 합니다. 각국 정부는 중앙통제력이 약화되는 것에 우려를 가질 수도 있습니다. 다섯째, **성능 및 확장성**입니다. 퍼블릭 체인을 사용하는 경우 처리속도와 비용 이슈가 있고, 프라이빗 체인은 신뢰자 수가 제한되는 딜레마가 있습니다. 트랜잭션 수수료(gas fee)가 발생할 경우 채택에 장애가 되므로, L2 솔루션이나 오프체인 프로토콜 등 추가 기술이 필요합니다. 여섯째, **기존 시스템과 연계**하는 데 시간이 걸립니다. 예를 들어 모든 서비스가 DID를 지원하기 전까지는 기존 계정 체계와 병행 운영해야 해서 복잡성이 늘어납니다.

결국 블록체인 기반 인증은 **"개인에게 권한을, 신뢰는 네트워크가"**라는 장점이 있지만, **미성숙한 생태계, UX 과제, 법제도 이슈** 등의 단점을 극복해야 합니다[53]. 특히 NIST 등은 블록체인 IDMS가 해결책이 될 잠재력은 인정하면서도, 현재로서는 전통 ID 관리의 문제를 완전히 치유하진 못하며 **추가 연구 및 프레임워크 정비**가 필요함을 지적합니다[58].

6) 글로벌 주요 표준 및 정책 흐름

이 분야의 표준화는 주로 **W3C**와 **DID Alliance, Decentralized Identity Foundation(DIF)** 등이 주도하고, 각국 정부와 기업 컨소시엄이 참여하고 있습니다. **W3C**는 2020년 **DID Core 1.0** 표준을 발표하여 탈중앙 식별자의 문법, 해석 방법 등을 국제 표준으로 규정했습니다[55]. 이어 2019년 제정된 **Verifiable Credentials Data Model 1.0**은 앞서 설명한 VC의 구조와 서명, 검증 절차를 표준화했습니다[59][60].

W3C 산하 **Credentials Community Group**과 **DID Working Group**은 현재도 2.0 버전(예: BBS+ Signatures 기반 ZKP 지원 등)을 작업 중입니다[55]. **DIF**(Decentralized Identity Foundation)는 마이크로소프트, IBM, 메타 등의 회원사로 구성된 단체로 DID해결, 신원허브, DIDComm 등 구현 지침을 제시하고 있습니다[61]. 예를 들어 **DIDComm v2**는 안전한 메시징 프로토콜로 채택되어, 다양한 지갑과 검증자 간 통신에 쓰입니다[49]. 또한 **OASIS** 재단에서는 분산 신원 프레임워크 표준(XDI 등)을 논의했고, **ISO/IEC**에서도 최근 분산 ID 관련 표준 (ISO/IEC 18013-7 모바일 운전면허증에 VC 활용 등)을 개발 중입니다.

정책 측면에서 유럽연합은 **eIDAS 2.0** 규정을 통해 **유럽 디지털 신원** 체계를 만들고, 회원국이 인정하는 **EU 지갑**에 VC를 담아 공공/민간 서비스에서 통용시키려는 법안을 추진 중입니다. 이는 탈중앙 신원 기술을 정부 차원에서 채택하는 흐름으로서 전 세계 이목을 끌고 있습니다. 미국 국토안보부(DHS)도 2020년대 초반 수백만 달러 규모의 SSI 기술 연구 지원을 하여, 운전면허증, 출입경 검문 등에 활용 가능성을 검증했습니다. 또한 각국 규제 기관들은 **개인정보 보호**를 강조하며 **프라이버시 보호 인증**을 권장하는 추세입니다. 블록체인 신원은 이러한 요구에 부합할 수 있어 긍정적으로 검토됩니다. 다만 동시에 정부 입장에서는 신원 체계의 통제권이 분산되는 것에 우려도 있으므로, EU처럼 **정부가 주도적으로 표준지갑 관리**에 나서거나, 한국처럼 **공동분산ID 협의체**를 구성하여 민관이 함께 표준을 만드는 방식이 나타나고 있습니다.

요약하면 표준은 **W3C DID/VC**가 중심축이며, 정책은 **EU eIDAS2**가 선도적인 역할을 하고 있습니다[62]. 앞으로 기술 성숙과 함께 정책도 **상호 인정 프레임워크** (어느 발행자의 VC를 신뢰할지, 어떤 속성은 법적으로 유효한지 등)를 정하는 쪽으로 발전할 것입니다. 또한 웹브라우저 표준에도 DID 통합 가능성이 논의되고 있고, OAuth/OIDC 등 기존 인증 프로토콜에도 **탈중앙 ID 확장** (OIDC4SSI 등)이 접목되는 등, 점진적인 융합이 진행 중입니다. 각국 정부들도 **신원주권**과 **디지털경제** 전략 차원에서 분산ID를 면밀히 검토하여, 향후 5~10년 내에 국제적으로 통용되는 분산 신원 체계의 윤곽이 드러날 것으로 전망됩

니다.

7) 향후 발전 전망 및 과제

블록체인 기반 분산 인증은 아직 초기이지만, **미래의 디지털 신원 인프라**로 성장할 잠재력이 높다는 평가를 받습니다. 향후 발전 방향으로는 첫째, **광범위한 상호운용성 확립**이 있습니다. 현재 다양한 DID 메소드와 네트워크가 난립해있지만, 향후에는 주요 표준으로 수렴하고 서로 신뢰를 연계하는 **루트 신뢰 네트워크**가 구축될 것입니다[52]. 이를 위해 각국 정부, 국제기구, 표준단체가 협력하여 **거버넌스 프레임워크**(예: 유엔 산하 또는 글로벌 컨소시엄에서 신뢰 등급 매기기 등)를 만들 가능성이 있습니다. 둘째, **사용자 경험 개선**입니다. 복잡한 키 관리와 지갑 UX를 대중이 쉽게 사용할 수 있도록 추상화하는 기술이 중요합니다. 예를 들어 **스마트폰 기본 기능으로 DID 지갑 내장**, 클라우드 기반 백업과 **소셜 복구**를 통해 분실 시 손쉽게 계정 복원, 그리고 인증시 **백그라운드에서 자동 증명**이 일어나도록 하는 등 UX 혁신이 이루어질 것입니다. 셋째, **프라이버시 강화 기술의 통합**입니다. 현재도 ZKP 등 기술을 쓰지만, 앞으로 **영지식 범위**를 넓혀 더 복잡한 주장도 노출 없이 검증하거나, **동형암호**로 원장에 민감 데이터가 있어도 해독 불가하게 하는 등의 연구가 실용화될 것입니다[63]. 이는 분산 신원의 프라이버시 우려를 완화하고 법규 준수를 높일 것입니다. 넷째, **양자 내성 및 보안 향상**입니다. 블록체인과 DID 시스템도 장기적으로 양자컴퓨터 등장에 대비해야 하므로, 향후 PQC 알고리즘(Dilithium 등)으로 전환하거나 **후기 양자 서명** 방식으로 업데이트가 필요합니다. 또한 스마트 컨트랙트의 포멀 검증, 체인 간 상호운용 보안강화 등 인프라 레벨 보안이 개선될 것입니다. 다섯째, **새로운 활용 영역 개척**입니다. 메타버스나 IoT 분야 등에서 수십억 개 기기와 가상 객체의 신원 관리에 DID가 적용될 수 있습니다. 예컨대 **사물 DID**를 통한 IoT 디바이스 인증, 디지털 콘텐츠 NFT와 DID 결합으로 소유 증명 및 접근제어, DAO(탈중앙조직) 멤버십 인증 등 신흥 분야에 응용될 전망입니다. 이는 **Machine Identity** 영역 확장과 맞물립니다.

이러한 발전을 위해 해결해야 할 **과제**들도 있습니다. 가장 큰 과제는 **생태계 채택**입니다. 기술이 좋아도 서비스 제공자들이 받아들이지 않으면 무용지물입

니다. 따라서 **양쪽 시장 문제**(발급자와 검증자 모두 일정수 이상 참여해야 효용 발생)를 넘기 위한 전략이 필요합니다. 정부 주도의 국민ID 사업이나 거대 플랫폼의 채택 등이 마중물이 될 수 있습니다. 또 하나는 **표준 경쟁과 단일화** 이슈입니다. 수많은 DID/VC 솔루션이 경쟁 중인데, 자칫 **또 다른 파편화**를 가져올 수 있습니다. 따라서 국제 협력을 통해 핵심 표준을 단일화하고 레퍼런스 구현을 공유하는 노력이 중요합니다. 법제도 측면에서는 각국이 **분산신원 법적 효력**을 인정하고, 책임 소재 (예: 잘못된 VC 발급 시 배상)는 어떻게 할지 정비해야 합니다. 기업들도 규제 눈치를 보며 도입하므로, 정부가 테스트베드 마련과 가이드 발행을 해줄 필요가 있습니다. 끝으로, **보안/프라이버시 트레이드오프**를 지속 관리해야 합니다.

분산 환경이지만 여전히 프라이버시 침해 가능성이 있고, 반대로 너무 익명성이 강하면 범죄 악용 우려도 있습니다. 예컨대 운영자금 세탁이나 인증 우회 등에 악용되지 않도록 **정책적 견제**와 기술적 수단(예: 필요시 법적 승인 하 특정 DID 연결성 밝히는 메커니즘 등)이 마련되어야 할 것입니다.

전망컨대, **웹3.0 시대**의 아이덴티티 인프라는 기존 빅테크 또는 국가가 독점한 모델에서 사용자/주체들이 **연합하고 분산 합의**하는 모델로 점진적 전환을 이룰 것으로 보입니다. 이는 **오래 걸리는 여정**이겠지만, 이미 많은 글로벌 표준화 활동과 시범사업이 진행 중이며, EU의 주도 아래 2030년경엔 **범유럽 분산신원 체계**가 상용화될 것으로 목표가 설정되어 있습니다. 궁극적으로 블록체인 기반 인증은 완전 대체라기보다 **기존 중앙 인증과 상호보완**하며, 이용자에게 **더 많은 통제권과 상호운용성**을 부여하는 방향으로 발전할 것으로 기대됩니다 [64][58].

마. 위험 기반 인증 (Risk-Based Authentication, RBA)

1) 기술 개요 및 작동 원리

위험 기반 인증(Risk-Based Authentication)은 인증 시도마다 **맥락(Context)과 행위**를 실시간 분석하여 **접근 위험도를 평가**하고, 그 위험 수준에 따라 추

가 인증이나 접근 차단 등의 **동적 대응**을 하는 방법론입니다. 전통적 이분법적 인증(성공/실패)과 달리, RBA는 매 로그인 시도에 대해 **계정 탈취 가능성**을 확률로 산출하고[65], 만약 **평소와 다른 의심스러운 상황**이면 사용자의 인증 절차를 강화합니다[65]. 예컨대 어떤 사용자가 평소 서울에서 로그인하던 계정에 갑자기 해외 IP에서 접근하면, RBA 엔진은 이를 **높은 위험 시나리오**로 분류하여 비밀번호 입력만으로는 허용하지 않고 추가로 OTP 입력이나 생체인증을 요구할 수 있습니다[66]. 반면 평소와 동일한 기기·위치에서의 정상적인 시도라면 별도 장애 없이 통과시킵니다. 이러한 방식으로 RBA는 **보안과 편의의 균형**을 상황별로 맞추어주는 스마트 인증 체계라고 할 수 있습니다.

작동 원리를 구체화하면, 위험 기반 인증 솔루션은 **로그인 요청의 다양한 맥락 정보**를 수집·분석합니다[67]. 주요 고려 요소로는 **장치(Device)** 정보(등록된 기기인가, 브라우저 지문, OS 등), **네트워크/위치(Location)** 정보(IP주소, 지리적 위치, VPN/프록시 여부), **사용자 행위 패턴(Behavior)**(로그인 시각과 평소 패턴 비교, 현재 요청 리소스의 민감도 등) 등이 있습니다[68]. 예를 들어 "새로운 기기 + 평소와 다른 국가 + 새벽 시간대 + 민감 자원 접근"과 같은 조합이면 위험 점수가 높게 산정됩니다[66][68]. RBA 시스템은 이러한 요소를 바탕으로 **룰 기반** 또는 **머신러닝 기반 모델**로 위험도를 계산합니다[69][70]. 그리고 미리 정해둔 정책에 따라 **Low Risk**면 추가 조치 없이 통과, **Medium Risk**면 2FA 요구, **High Risk**면 차단 또는 관리자 승인 요구 등의 **다단계 대응**을 합니다[71]. 이때 "위험"의 정의와 임계치는 해당 시스템의 보안 요구에 맞게 설정됩니다. 일반적으로 **계정 탈취 공격**(도난된 크리덴셜 사용, 봇넷 로그인 시도 등)을 탐지하는 것이 1차 목표이지만, **내부자 오용**이나 **세션 중간 탈취** 등도 위험도로 반영할 수 있습니다. 예컨대 세션 중에 갑자기 사용자의 행동 패턴이 달라지면(예: 키입력 속도나 마우스 움직임이 평소와 현저히 다르거나, 동일 세션에서 IP가 바뀌는 등) **연속적 위험 모니터링**을 통해 추가 인증을 요구하는 식입니다.

결국 RBA의 핵심은 **"항상 모니터링하고 의심스러우면 신원재확인"**이라고 요약할 수 있습니다. NIST 등에서는 이를 **어댑티브 인증(adaptive auth)** 또는 **연속 인증**과 연계하여 설명하며, 제로트러스트 보안 모델에서도 **항시 검증**

(Continuous verification) 원칙 아래 위험 기반 접근제어를 강조합니다[72]. 예를 들어 Zero Trust 구현에서 **사용자 디바이스 신뢰 점수**나 **행동 이상징후**를 토대로 세션을 끊거나 재인증시키는 것이 이에 해당합니다. RBA는 사용자 측에 거의 **투명하게 작동**할 수 있다는 것이 특징입니다. 위험이 낮으면 사용자 경험에 변화가 없고, 위험이 높을 때만 추가 MFA 등의 **프롬프트가 트리거**되므로, 전체적으로는 **필요 최소한의 마찰**만 발생시킵니다[73]. 이러한 작동 방식 덕분에 RBA는 **"좋은 트래픽에는 친절하게, 나쁜 트래픽에는 엄격하게"**라는 원칙을 구현하는 기술이라 할 수 있습니다.

2) 주요 구현 방식 및 프로토콜

위험 기반 인증은 구체적인 **솔루션/제품** 형태로는 IAM(Identity & Access Management) 제품들의 한 기능으로 구현되거나, 개별 서비스에 **정책 엔진**으로 내장됩니다. 기술적으로 표준화된 단일 프로토콜이 있는 것은 아니지만, 구성 요소를 크게 나누면 **(1) 위험 평가 엔진**, **(2) 정책 룰셋**, **(3) 액션(추가 인증 수단)**으로 볼 수 있습니다. **위험 평가 엔진**은 다양한 신호(signals)를 입력 받아 **실시간 스코어링**을 수행합니다. 단순한 구현은 정해진 규칙에 따라 점수를 매기는 룰 엔진 형태입니다. 예컨대 "새 기기이면 +30점, 해외 IP이면 +50점, 블랙리스트 IP이면 +100점" 등의 규칙을 운영자가 설정해두고 합산하는 방식입니다[74]. 고급 구현은 머신러닝/통계 모델을 사용합니다.

많은 서비스들이 사용자 로그인 패턴(시계열) 데이터를 학습하여 **평균 프로파일과의 편차**로 이상 여부를 판정합니다[75]. 구글, 마이크로소프트 등의 대규모 서비스는 수억 건의 로그인 데이터를 기반으로 **이상탐지 AI**를 돌려, 수상한 로그인(예: IP대역, 유저 에이전트, 시각의 조합)이면 위험 플래그를 세우는 것으로 알려져 있습니다. 또한 **디바이스 지문(Device fingerprint)** 기술도 활용됩니다. 브라우저 쿠키, WebGL 정보, 캔버스 해시 등으로 기기를 식별하여 인지된 기기인지 판별합니다[68].

정책 룰셋은 위험 점수 구간에 따른 대응을 정의합니다. 예: 0~30점=정상 통과, 31~60점=MFA 요구, 61점 이상=차단 같은 식입니다[71]. 이 정책은 관리자

콘솔에서 설정 가능하도록 하거나, 또는 위험 엔진이 점수 대신 **High/Medium/Low** 레이블을 내부적으로 반환하는 식으로 하기도 합니다. RBA 솔루션은 일반적으로 **관리자가 조정 가능한 임계치와 예외 규칙** 인터페이스를 제공합니다. 예를 들어 VIP 계정은 항상 MFA 요구하도록 하거나, 회사 내부망 IP는 위험도를 낮추는 등 튜닝이 가능합니다.

액션(대응 수단)으로는 보통 **스텝업(step-up) 인증**과 **블록(block)**, **알림(notification)** 등이 있습니다. 가장 흔한 것이 **MFA 추가 인증** 요구입니다[76]. 현재 로그인 시도에 등록된 2차 인증 수단(OTP, 푸시, 지문 등)을 추가로 입력하도록 하고, 성공하면 통과시킵니다. 또 다른 액션은 아예 로그인 시도를 **차단**하고 계정을 잠그는 것입니다. 위험이 매우 높거나 사기 의심이 확실할 경우 보안상 차단이 이루어지며, 계정주에게 **알림**을 보내기도 합니다. 예를 들어 "본인 계정에 의심스러운 로그인이 차단되었으니 비밀번호를 변경하라"는 이메일을 발송하는 식입니다. 경우에 따라서는 **비밀정보 추가 질문**(KBA)처럼 비표준적 액션을 취하기도 하지만, 이는 요즘 권장되지 않습니다.

RBA 구현은 전용 솔루션으로는 Okta Adaptive MFA, Microsoft Azure AD Conditional Access, Cisco Duo Trust Monitor, RSA Adaptive Auth 등으로 제공됩니다. 이들은 REST API나 SAML/OIDC 연동으로 기존 시스템과 통합됩니다. 서비스 자체 개발 사례로는 각종 대형 웹서비스들이 자체 RBA를 운영하는데, 예를 들어 구글은 **Risky Login Challenge**로 의심스러운 로그인에 자동 보안질문/전화 확인을 거는 것으로 알려졌고, 페이스북도 **Login Approvals**라 하여 낯선 환경 로그인 시 추가 확인을 요구합니다. 프로토콜 표준은 없지만, OAuth2.0/OIDC에서는 acr이나 amr 클레임을 사용해 인증 강도와 방법을 표현하는 규약이 있어, RBA 엔진이 "높은 보안 레벨 요구" 신호를 Identity Provider에 보내면 IdP가 추가 인증 수행 후 acr 값을 높여 다시 인증해주는 식으로 활용되기도 합니다.

또한 **Continuous Authentication Protocols**이 연구되고 있는데, 이는 세션 중 지속적으로 위험 평가를 실시하여 필요 시 재인증하는 개념입니다. 업계동향으로는 FIDO 얼라이언스도 **Device Authentication** 작업 그룹에서 위험 기

반 접근의 표준화를 논의하고 있습니다(예: FIDO Device Metadata를 활용한 신뢰 점수 교환). 전반적으로 RBA는 개별 기업 구현에 머물러 있으나, **제로트러스트** 열풍과 맞물려 **지속적 맥락 인증**이 새로운 표준으로 자리잡아가는 추세입니다.

3) 보안성 분석

위험 기반 인증은 **정적 정책으로 잡아내기 어려운 공격 시나리오**를 탐지하고 차단해준다는 점이 가장 큰 보안상 장점입니다. 특히 **크리덴셜 탈취 공격**에 효과적인데, 설령 공격자가 올바른 비밀번호를 알아내도, RBA 엔진이 **맥락의 차이**를 포착하여 추가 장치를 가함으로써 차단 확률을 높입니다[66]. 실제 통계에 따르면 RBA 도입 시 계정 탈취 시도의 **80% 이상을 자동 저지**할 수 있다는 보고도 있습니다[77]. 또한 RBA는 기존 인증 수단의 약점을 보완합니다. 예를 들어 SMS OTP가 가로채기 당해도, 만약 공격자의 접속 환경이 매우 이례적이라면 RBA로 의심을 가져 추가 음성인증 등 다른 수단으로 재확인하게 할 수 있습니다. 이처럼 **다층 방어** 효과가 있습니다.

RBA의 또 다른 장점은 **사용자 경험 측면의 보안 강화**입니다. 보안을 높이기 위해 모든 로그인에 복잡한 MFA를 강제하면 사용자 불편이 커지지만, RBA는 **"정상 사용엔 간편하게, 공격 의심엔 엄격히"**라는 **어댑티브** 접근으로 보안과 편의의 균형을 맞춥니다[73]. 이를 통해 이용자 업무 흐름을 크게 방해하지 않으면서도 위험할 땐 경각심을 주고 대응하게 하므로, **보안 인식 제고**에도 도움이 됩니다.

물론 한계와 취약점도 존재합니다. 첫째로 **오탐(False Positive)** 문제입니다. RBA는 이상탐지에 기반하므로, 정상 사용이더라도 상황 변화로 위험도가 올라갈 수 있습니다. 예컨대 사용자가 실제로 출장 가서 해외에서 로그인하면 RBA가 잘못 차단할 수 있습니다. 이로 인해 사용자는 추가 인증 번거로움을 겪거나 서비스 이용이 지연될 수 있습니다. 오탐을 줄이는 튜닝이 중요하지만, 완전히 없앨 수는 없습니다. 둘째로 **우회 가능성**입니다. 영리한 공격자는 RBA를 속이려고 **정상 패턴을 모방**할 수 있습니다. 예를 들어 피싱으로 크리덴셜을 탈

취한 공격자가 처음엔 IP 프록시를 써서 피해자 국가의 IP로 로그인 시도할 수 있고, 사용자 에이전트, 사용 시간대 등도 맞출 수 있습니다. 어느 정도까지는 시나리오 연기로 위험 점수를 낮출 수 있다는 연구 결과도 있습니다. 특히 **지속적인 계정 탈취자**는 한 계정 정보를 빼내 여러 번 로그인 시도하면서 RBA 임계치를 파악해볼 수도 있습니다. 따라서 RBA를 맹신하기보다는 기본적인 **MFA 도입**과 병행해야 합니다. 셋째, **보안 신호 수집의 한계**입니다. 프라이버시 이슈로 인해 모든 신호를 다 수집할 수 없는 경우도 있습니다. 예컨대 EU GDPR에 따라 사용자의 위치정보나 행동로그를 마음대로 써 위험 분석에 활용하기 어렵다는 지적이 있습니다. 브라우저의 추적방지 기능도 디바이스 지문 수집을 제한합니다. 이런 제약은 RBA 정확도를 떨어뜨릴 수 있습니다. 넷째, **적응형 모델 학습**의 위험입니다. 만약 공격이 점진적으로 이뤄져서 RBA 엔진이 그것을 정상으로 학습해버리면 탐지하지 못할 수도 있습니다. 즉 **"끓는 물 속 개구리"** 식으로 조금씩 다른 IP, 조금씩 다른 시간대를 늘려가며 계정에 접근하면 엔진이 변화로 인식하지 못하는 한계도 있을 수 있습니다. 이를 막으려면 주기적으로 모델 기준을 재조정해야 합니다.

결론적으로, 위험 기반 인증은 **동적인 맥락 정보를 활용해 보안을 한층 강화**하는 효과적인 방안이지만, **완벽하지 않으며** 어디까지나 추가 방어층으로 이해해야 합니다. 특히 **RBA를 우회하는 사회공학**(예: 사용자가 위험 알림을 무시하도록 유도) 가능성에도 대비해야 합니다. RBA 경고가 자주 뜨면 사용자들이 이에 둔감해질 수 있고, 공격자는 오히려 이를 이용해 사용자를 속일 수도 있습니다. 따라서 RBA 도입시 경고 빈도와 메시지 내용을 잘 디자인해야 합니다.

4) 활용 사례 (산업별 적용 현황)

위험 기반 인증은 **전자상거래, 금융, 클라우드 서비스** 등 온라인 계정 보안이 중요한 산업에서 폭넓게 채택되고 있습니다. **전자상거래 및 게임 분야**에서는 계정 도용으로 아이템이나 포인트가 탈취되는 문제가 커서, 많은 기업들이 RBA 솔루션을 도입했습니다. 예를 들어 국내 한 게임사는 해외 IP에서의 로그인 시 **거의 100% 추가 인증**을 거치도록 하여 중국발 해킹을 크게 줄였다고

합니다. **온라인 뱅킹/결제** 분야도 오래전부터 RBA 개념을 활용해 왔습니다. Visa나 Mastercard의 **3D Secure 2.0** 프로토콜에는 **Risk-Based Authentication** 단계가 있어, 거래 정보를 분석해 의심스러운 카드 결제만 발행은행 OTP 인증을 추가 요구하도록 합니다[35]. 또한 은행들은 웹 로그인 시 IP, PC 환경 등을 대조하여 비정상 패턴이면 **전화 인증**을 더 요구하거나 아예 로그인 차단 후 고객에게 경고합니다. **기업용 SaaS와 업무 VPN** 환경에서도 RBA는 필수 요소가 됐습니다. Okta, Azure AD 등은 **조건부 접근 정책**으로 위치, 디바이스 상태에 따라 MFA 적용을 달리하며, Google Workspace도 관리자들이 **Login Challenge** 설정으로 의심스러운 로그인에 **보안키** 등 높은 보안요소 요구를 설정할 수 있습니다. **클라우드 인프라**(AWS, GCP) 콘솔도 사용자 IP 화이트리스트, 새로운 지역 접근시 이메일 알림 등을 제공해 RBA 효과를 내고 있습니다.

일반 소비자 서비스 중 Google은 오랫동안 **위험한 계정 액세스 차단** 메커니즘을 운영해왔습니다. 사용자는 "새 기기/새 위치 접근 차단됨" 이메일을 받아본 경험이 있을 것입니다. Microsoft 계정도 마찬가지로 **"우리가 인지하지 못한 로그인 시도가 있었음"**을 통보하고 차단합니다. 페이스북은 **로그인 알림** 및 **이중 승인** 기능으로 비정상 로그인 시도시 유저에게 알리고 본인 확인을 거칩니다. 이러한 RBA 적용으로 대형 서비스들의 계정 도용률이 많이 낮아졌습니다. 예컨대 Gmail은 피싱된 비밀번호로 로그인 시도되는 경우 대다수를 추가 질문 또는 기기 알림으로 걸러낸다고 발표한 바 있습니다. **모바일 앱** 쪽에서도 RBA 활용이 늘고 있습니다. 모바일 뱅킹 앱들은 기기 정보와 통신사 정보 등을 종합해 단말 변경을 탐지하고, 변칙이면 추가 본인확인을 하거나 사용을 제한합니다. **결제 승인** 시에도 과거 패턴과 다른 큰 금액 결제면 앱에서 재로그인하게 하거나, 위치 불일치 결제면 차단하는 등 시나리오 기반 통제가 이뤄지고 있습니다.

이외에 **중요 인프라 시스템** (예: 정부 행정망)에서도 RBA 개념을 도입하는 사례가 늘고 있습니다. 내부 직원이 평소와 다른 PC에서 접속하거나, 일반 직원이 평소 안 다루던 대량 데이터를 빼내려고 하면, 시스템이 이를 탐지해 계정 잠금이나 관리자 경고를 주는 식입니다. 이는 내부자 이상행위 탐지와 IAM를

결합한 형태입니다.

전반적으로, 위험 기반 인증은 **광범위한 서비스에 사실상 기본 탑재**되어 가고 있습니다. 특히 MFA 적용이 어려운 환경에서는 RBA라도 적용해 보안을 높이는 식으로 활용됩니다. 한 조사에서는 90% 이상의 대형 금융기관이 RBA 기법을 일정 부분 사용하고 있다는 결과도 있습니다. 더불어 **규제 준수**에도 활용되는데, 유럽 PSD2에서 **거래위험분석(Transaction Risk Analysis)**을 통해 저위험 거래는 SCA 면제 가능토록 한 것이 대표 예입니다. 이는 일종의 RBA를 법적으로 인정한 셈입니다. 미국 FFIEC 가이드라인도 **계층적 보안과 위험 기반 접근**을 권고하여 사실상 RBA 활용을 업계 표준으로 정착시켰습니다[35][78].

5) 기술적/운영상 장단점

위험 기반 인증의 **주요 장점**은 앞서 보안 분석에서 언급한 바와 같이 **보안성과 사용자 편의성의 최적 절충**이라는 점입니다. **보안 측면**에서 RBA는 정교한 공격 탐지로 계정탈취 방어율을 높이고, **편의 측면**에서 정상 이용자에게는 불필요한 장벽을 치우는 효과가 있습니다[73]. 결과적으로 **사용자 경험을 거의 해치지 않으면서 보안을 강화**할 수 있는 수단이기에, 고객 서비스 만족도에 민감한 업계에서 선호됩니다. 두 번째 장점은 **적응성(adaptability)**입니다. 새로운 공격 기법이나 위협 트렌드에 따라 RBA 룰이나 모델을 업데이트함으로써 **유연한 대응**이 가능합니다. 예컨대 특정 기간 피싱 공격이 유행하면 "동시간대 여러 계정 로그인 시도" 같은 규칙을 일시 강화하는 등의 대응이 쉬워집니다. 세 번째로 **관리 효율성**입니다. RBA를 도입하면 보안팀이 모든 로그인 로그를 일일이 살피는 대신 시스템이 **자동 분류**해주므로, 경고에만 집중하면 되어 업무 효율이 올라갑니다. 또한 위험 징후별 통계가 쌓여 **위협 인텔리전스**로 활용될 수도 있습니다. 네 번째, **규제 준수와 고객 보호**입니다. 금융기관 등에서는 강력인증이 필수인데, RBA 기반으로 필요 시에만 MFA를 요구하면 규제는 지키면서 고객 불편은 최소화할 수 있습니다. 고객 보호 측면에서도, 계정이 이상징후가 있을 때 바로 알려주어 사전에 비밀번호 변경 등 대응하게 해주므로 신뢰도를 높입니다.

반면 **단점과 한계**도 뚜렷합니다. 우선 **False Positive/False Negative 문제**입니다. RBA 엔진은 완벽하지 않아, 때로 정상 유저를 의심하거나(오탐), 공격자를 평범하다고 놓칠 수 있습니다(미탐). 오탐이 많은 경우 사용자가 불편을 느끼고 보안을 무시하게 되어 오히려 역효과가 날 수 있습니다. 예컨대 자꾸 추가 인증을 시키면 편법으로 보안 기능을 끄려 할 수도 있습니다. 반대로 미탐이 생기면 보안 사고로 이어지겠지요. 둘째, **초기 구축과 튜닝의 복잡성**입니다. RBA를 효과적으로 쓰려면 어떤 신호를 볼지, 가중치는 어떻게 둘지 등 많은 시행착오와 전문지식이 필요합니다. 잘못 설정하면 아예 작동을 안 하거나, 무의미한 경보만 연발할 수 있습니다. 특히 ML 기반으로 갈 경우 데이터 준비와 모델 검증이 어려울 수 있습니다. 셋째, **개인정보 및 윤리 이슈**입니다. RBA는 사용자의 위치, 기기정보, 행동로그 등을 수집하므로 프라이버시 침해 우려가 있습니다. 이를 위해 정책적으로 해당 데이터의 사용 목적을 제한하고, 익명화 등을 해야 하나, 여전히 "사용자 동의 없이 행동을 감시한다"는 비판이 있을 수 있습니다. 넷째, **전문가 개입 필요성**입니다.

RBA는 완전 자동화가 이상적이지만, 현실에서는 보안팀이 룰을 관리하고 경보를 조사해야 합니다. 너무 엔진에만 의존하면 놓칠 수도 있고, 그렇다고 모든 결정을 사람에게 맡기자니 느려집니다. 그래서 **인력 운영**의 마지노선을 정하는 것이 어렵습니다. 다섯째, **공격자의 역이용 가능성**입니다. RBA는 공격자를 필터링하지만, 공격자가 RBA 동작을 역이용할 가능성도 있습니다. 예를 들어, 의도적으로 여러 정상 사용자의 로그인을 계속 실패시켜 RBA가 그 사용자들을 잠그게 만들거나, RBA 경고 메시지를 가장하여 피싱하는 등입니다. 이러한 사회공학적 리스크도 고려해야 합니다. 마지막으로, **잔여 공격 벡터**입니다. RBA가 잡을 수 없는 공격도 있습니다. 예를 들어 사용자의 세션 쿠키를 탈취해 그대로 사용하면, RBA는 새로운 로그인 시도가 없으니 감지하지 못할 수 있습니다. 또는 사용자 기기 자체가 악성에 감염되어 있으면 RBA는 정상으로 보일 것입니다. 따라서 **RBA 만능주의는 경계**해야 합니다.

운영 측면에서는, RBA 도입 후 **성공률 측정과 지속 개선**이 필수입니다. 어떤 비율로 추가 인증이 발생하며 그 중 실제 공격이 얼만지(Precision/Recall 측

정)가 필요합니다. 또한 사용자 불만을 모니터링하고, 고객지원과 연계하여 "왜 차단되었는지" 해명해줄 체계도 있어야 합니다. RBA 결정에 불복하는 합법 사용자에게 대체 인증수단을 주거나, 일정 시간 후 다시 시도해보라고 안내하는 UX도 중요합니다.

6) 글로벌 주요 표준 및 정책 흐름

위험 기반 인증 자체는 특정 기술표준이기보다는 **보안 모범사례(Best Practice)**에 가깝지만, 관련하여 간접적으로 영향 있는 표준과 규제가 있습니다. **미국 NIST**의 디지털 인증 가이드라인(SP 800-63)에는 RBA를 직접적으로 언급하진 않지만, **"장치 및 세션 특성에 따른 리스크 대응"**을 권장하는 내용이 포함되어 있습니다[79]. 또한 **NIST SP 800-207 (제로트러스트 아키텍처)**는 **"모든 접속은 지속적으로 평가되어야"** 한다고 명시하여, 사실상 RBA/연속인증을 요구하고 있습니다[72]. **유럽 PSD2** 규정에서는 거래 위험 분석(TRA)을 기반으로 **SCA (Strong Customer Authentication)** 면제를 허용하는데, 이 면제를 적용하려면 은행들은 정교한 RBA 엔진을 갖춰야 합니다.

EBA 가이드라인에는 사기율 통계와 신뢰도 기준까지 제시되어 있어, 금융기관이 **RBA 성능**을 일정 수준 이상 유지하도록 압력을 가하고 있습니다. **FFIEC** (미국 연방금융검사협의회) 2021년 가이드라인도 "단일요소 인증은 고위험 거래에 불충분하며, **리스크 기반 접근관리**가 필요"하다고 강조합니다[35]. 이러한 규제들은 사실상 RBA 도입을 간접 의무화하는 효과를 냅니다. **산업 표준** 측면에서는, **PCI-DSS(v4.0)**도 관리자 접근 등에서 **동적 인증**을 요구하고 있고, **CIS 컨트롤**에서도 계정 모니터링 및 제어를 권장합니다.

제품 표준은 아니지만, **FIDO 얼라이언스**는 2020년대 들어 **"수준높은 인증자(FIDO) + RBA"** 조합을 많이 홍보하고 있습니다[80]. FIDO의 공개키 인증은 피싱에는 강하나 기기 분실에는 약한데, RBA와 결합하면 기기위험 감지 시 FIDO 키도 사용 제한하는 식으로 보완이 가능합니다. 따라서 FIDO2 기술 문서에 RBA 활용에 대한 언급이 추가될 움직임도 있습니다. **IETF**에서도 Continuous Authentication 관련 연구 그룹이 있으며, OAUTH WG에서

"Step-up Authentication Challenge" 등에 대한 논의가 이루어지고 있습니다. 하지만 아직 정식 RFC는 없습니다.

요즘 **제로트러스트**가 사이버보안 정책의 화두가 되면서, 각국 정부 조달 규격에도 RBA와 연속 검증 요구사항이 들어가기 시작했습니다. 예컨대 미국 DoD의 Zero Trust Reference Architecture는 사용자 단에서 **시너지 기반 신원 점수**를 계산해 접근 제어하는 그림을 담고 있습니다. 국내에서도 2022년 정부 발표 제로트러스트 전략에 "동적 인증 및 접근통제"가 핵심 원리로 포함되었습니다.

결국 정책적 흐름은 **"항상 맥락을 평가하라"**는 방향으로 확고합니다. 이것은 RBA 기술의 중요성을 뒷받침하므로, 앞으로 RBA는 개별 구현 차원을 넘어 **플랫폼 기본기능**이나 **SDK** 형태로도 제공될 전망입니다. 예컨대 브라우저가 자동으로 위치이상 로그인을 감지해 OIDC 공급자에게 신호를 주는 식의 표준이 나타날 수 있습니다. 또한 **프라이버시 강화 RBA**(edge에서 위험평가, 서버엔 최소 정보 제공 등) 같은 연구도 정책 필요에 의해 진행될 것입니다[81].

7) 향후 발전 전망 및 과제

위험 기반 인증은 향후 인증 체계의 필수 요소로 더욱 고도화될 전망입니다. 몇 가지 발전 방향을 살펴보면, 첫째로 **머신러닝/AI의 더 깊은 접목**입니다. 이미 ML을 쓰고 있지만, 향후엔 **딥러닝**을 통한 사용자 행위 분석으로 **정밀한 개별 프로파일링**이 가능해질 것입니다[82]. 예컨대 키스트로크, 마우스 움직임, 터치 패턴 등의 **행동 생체정보**까지 통합 분석하여 **사용자 고유의 "디지털 습관 지문"**을 학습하고, 조금만 벗어나도 감지하는 수준으로 나아갈 수 있습니다[83][84]. 둘째, **실시간성**과 **연속성 강화**입니다. 현재는 로그인 시점 위주로 적용되지만, 향후에는 **세션 중 지속적으로** 위험을 평가해 **"항상 검증"**하는 형태가 늘어날 것입니다[72]. 특히 제로트러스트 모델에서 사용자와 디바이스 신뢰는 **지속 재평가**가 원칙이므로, 일정 시간마다나 이벤트 발생 시 **자동 재인증**이 일반화될 수 있습니다. 이를 위해 UX 저항을 최소화하는 투명한 재인증 기술이 함께 발전할 것입니다. 셋째, **데이터 공유**

와 협력입니다. 개별 서비스의 RBA도 한계가 있으므로, **업계 간 위험정보 공유**가 활성화될 수 있습니다. 예컨대 구글 계정에서 의심된 IP를 다른 서비스와 공유하거나, FIDO2 인증기 제조사가 기기 신뢰평가를 실시간 제공하는 식입니다. 이런 **공유 인텔리전스**는 공격자를 한 곳에서 잡으면 전체에서 막는 협업 효과를 냅니다. 다만 개인정보 이슈로 민감하므로, **익명화된 피드**나 **스코어 교환** 방식 연구가 필요합니다.

넷째, **사용자 컨트롤과 투명성**입니다. RBA가 자동화되다 보면 사용자 입장에서 "왜 추가 인증이 나왔는지" 알기 어려울 수 있습니다. 향후에는 사용자에게 **위험 알림 대시보드**를 제공해, 최근 로그인 활동과 차단 이력을 보여주고 필요시 여행 모드 등 예외 설정을 허용하는 기능이 도입될 수 있습니다. 이는 사용자 신뢰를 높이고 불편을 줄일 것입니다. 다섯째, **프라이버시 보호 기술 결합**입니다. 위험평가에 필요한 데이터 중 민감한 것은, **연합학습(Federated Learning)**이나 **로컬 처리** 방식으로 수집 없이 활용하도록 발전할 수 있습니다 [81]. 브라우저나 OS가 사용자 패턴을 로컬에서 학습해 해시만 서버로 보내는 등 프라이버시 보호형 RBA 연구가 진행될 가능성이 있습니다. 여섯째, **공격 회피 대응**입니다. 공격자들이 RBA를 속이려는 시도가 증가할 것이므로, RBA 엔진도 이에 대응하는 반대 전략이 필요합니다. 예컨대 "IP, UA만 같다고 낮은 점수 주지 않고, 미묘한 패턴 차이까지 본다"거나, "공격 흔적이 의도적으로 없는 경우 오히려 의심 점수 부여" 같은 **대항적 모형**을 만들어야 합니다.

RBA 발전에 따라 해결 과제들도 있습니다. **표준화** 측면에서, 현재 제각각인 RBA 구현에 대한 **상호운용 프로토콜** (예: 위험 점수/레벨을 IdP와 RP간 주고받는 OIDC 확장 등)이 필요할 수 있습니다. 이는 연합인증 환경에서 일관된 RBA 적용을 가능케 할 것입니다. **윤리적 이슈**로는, RBA가 특정 사용자 집단에 불리하게 작용하지 않도록 해야 합니다. 예를 들어 해외 자주 나가는 직업인이나, TOR 같은 프라이버시 툴을 쓰는 사용자가 과도한 불이익을 받지 않게 **편향 제거** 노력이 필요합니다. AI를 쓴다면 **AI 투명성**과 **설명가능성**도 중요한 과제입니다.

결론적으로, 위험 기반 인증은 **정적 암호 시대**에서 **동적 맥락 인텔리전스 시대**로 가는 다리 역할을 하고 있으며, 향후 점점 **모든 접근제어 결정에 기본 요소**로 자리잡을 것입니다[85][71]. 완벽하진 않지만, 다른 보안 기술들과 **심층 방어**의 한 축으로 결합되어 계정 및 시스템 보안을 한층 강화하는 것이 기대됩니다.

바. 연속 인증 (Continuous Authentication)

1) 기술 개요 및 작동 원리

연속 인증(Continuous Authentication)은 사용자가 시스템에 로그인한 이후에도 **세션 내내 지속적으로 사용자의 신원을 확인**하는 개념입니다[72]. 전통적으로 인증은 로그인 순간에 1회 이뤄지고 이후 세션 전체를 신뢰하지만, 연속 인증에서는 **사용 중인 사람이 계속 같은 본인인지**를 **끊임없이 모니터링**하고, 의심스러울 경우 세션을 종료하거나 재인증을 요구합니다. 이를 통해 공격자가 로그인 후 세션을 가로채거나, 혹은 로그인 직후 합법 사용자가 자리를 떠났을 때 제3자가 그 세션을 악용하는 상황 등을 탐지·차단하는 것이 목표입니다.

연속 인증은 다양한 **사용자 행동 특성(behavioral biometrics)**과 환경 신호를 활용합니다. 예를 들어, **타이핑 패턴**(키 스트로크의 속도와 압력, 오류율 등), **마우스/터치 사용 패턴**(움직임 경로, 클릭 간격), **스마트폰 잡는 자세나 걷는 걸음걸이**, **사용자의 얼굴/목소리** 등을 수시로 분석하여 현재 조작자가 원래 사용자와 동일한지 점검합니다[83][84]. 이때 미리 사용자의 **행동 프로필**을 학습해두거나, 사전에 정의된 인간 행위 특징들을 비교합니다.

예컨대 어떤 사용자는 항상 100~200ms 사이 간격으로 타자를 치는데, 갑자기 그 패턴이 300ms 이상으로 느려지고 오타율이 치솟았다면 다른 사람이 대신 입력하는 것으로 의심할 수 있습니다. 혹은 PC 앞 웹캠으로 주기적으로 얼굴을 찍어 얼굴인식으로 동일인 여부를 확인할 수도 있습니다(일부 보안 프로그램은 10분 간격으로 사용자 얼굴 재확인). **이러한 정보들은 실시간 위험 점수로 환산**되어, 기준을 벗어나면 연속 인증 시스템이 개입합니다.

연속 인증의 **동작 사이클**은 일반적으로 다음과 같습니다: **(1) 초기 등록 단계**에서 사용자의 행동 데이터(예: 타이핑 패턴)를 일정 기간 수집하여 **참조 모델**을 구축합니다[86]. **(2) 인증 후 감시 단계**에서 사용자가 세션을 진행하는 동안 연속적으로 행동 데이터를 포착합니다. 웹 브라우저의 자바스크립트, OS 에이전트, 모바일 앱 SDK 등 다양한 수단으로 비침습적으로 데이터가 수집됩니다. **(3) 실시간 비교 단계**에서 수집된 데이터와 참조 모델을 **알고리즘이 비교**하여 **유사도/일치도 점수**를 산출합니다. 예컨대 사용자의 평균 키 입력 비율과 현재 비율 차이를 통계량으로 나타냅니다. **(4) 결정 단계**에서 그 점수가 사전에 정의한 임계치보다 낮으면(즉 일치하지 않으면) **의심 이벤트**가 발생합니다[87]. 그러면 **(5) 대응 단계**로, 시스템이 자동으로 조치를 취합니다. 주로 **세션 잠금/로그아웃**, **재인증 요구(예: 비밀번호 다시 묻기)**, 또는 **경고 알림** 등이 가능합니다. 만약 점수가 정상 범위라면 별다른 조치 없이 모니터링만 계속 이어집니다. 이 사이클이 **지속적으로 반복**되어, 사용자가 상호작용하는 내내 인증이 *"현재 진행형"*으로 유지되는 셈입니다[72].

연속 인증은 구현에 따라 **투명 인증(invisible)**과 **명시적 재인증(explicit)**을 구분할 수 있습니다. 투명 모드에서는 사용자가 전혀 인식 못하는 배경에서만 점검이 이뤄지고(정상 시 아무런 프롬프트 없음), 명시 모드에서는 일정 주기나 의심 시점마다 사용자에게 "본인인지 확인" 질문이 뜹니다. 이상적인 연속 인증은 투명 모드로, 사용자는 자연스럽게 일하면 되고 시스템이 알아서 실시간 보증해주는 것입니다. 최근 연구와 산업계 모두 **부드러운 연속 인증**을 지향하며, 이를 위해 **행동 생체** 등의 소극적 신호를 최대한 활용하려 합니다[88].

연속 인증의 개념은 제로트러스트의 "**절대 신뢰하지 말고 항상 검증**" 원칙과도 맥락을 같이 합니다[72]. 특히 **"지속적 사용자 확인"**은 고위험 환경(예: 군사망, 공장 제어 시스템)에서 인간요소 보안을 강화하는 방법으로 주목받고 있습니다. 나아가 미래에는 사람뿐 아니라 **기계 프로세스**에 대해서도 연속 검증 개념이 확장되어, **항시 신뢰 평가**가 표준이 될 것이라는 전망도 있습니다.

2) 주요 구현 방식 및 프로토콜

연속 인증을 구현하는 방법은 대개 **클라이언트 측 에이전트** 또는 **어플리케이션 내 SDK** 형태로 동작합니다. **클라이언트 에이전트** 방식은 OS 수준에서 동작해 사용자의 전체 행위(여러 앱에서의 입력 패턴 등)를 감지하고 분석합니다. 예컨대 군사용 보안 솔루션에서는 PC에 드라이버/에이전트를 설치해 키 입력, 마우스 움직임, 얼굴 웹캠을 항상 모니터링하고 이상 시 OS 세션을 잠그기도 합니다. **응용 프로그램 내 SDK** 방식은 특정 애플리케이션(예: 모바일 뱅킹 앱)에 라이브러리를 삽입하여, 사용자의 터치 압력, 기울기 센서, 스크롤 습관 등을 측정합니다. 보안 회사 BioCatch 등이 이러한 모바일 행동 생체 정보를 수집해 고객사 앱에 연속 인증 기능을 부여합니다[83][84]. 웹 환경에서는 자바스크립트를 통해, 예컨대 금융 웹사이트 폼에 입력 시 입력 패턴(속도, 제자리 타이핑 등)을 재고, 세션 중 일정 주기로 hidden AJAX call로 현재 윈도우 포커스, 마우스 움직임량 등을 전송하는 방식이 연구되었습니다.

분석 알고리즘 측면에서는, 연속 인증은 **행동 생체(Behavioral Biometrics)** 기술과 긴밀히 연결됩니다. 주로 사용되는 특징들에는 **키스트로크 다이내믹스** (Keystroke dynamics) – 키 누르는 시점과 시간 간격, 누르고 떼는 시간 등, **포인터 이동 패턴** – 곡선/직선 이동 선호, 가속/감속 곡선, **터치 제스처** – 스마트폰에서 드래그할 때의 속도와 압력, **사용 루틴** – 특정 작업 순서나 앱 전환 패턴 등이 있습니다[89][83]. 이러한 특징 벡터를 머신러닝 분류기나 통계 모델로 학습시켜 **사용자 식별 모델**을 만듭니다[90]. 일부 연구는 **심장박동 패턴**(예: 키보드 터치시마다 손의 PPG 신호)이나 **ECG 센서**로 인체 신호를 추가 활용하기도 합니다. 최근에는 **딥러닝**(CNN, RNN 등)을 적용해 키입력/마우스의 시계열 신호에서 개개인 특징을 추출하는 시도가 있습니다[82]. 정확도 향상을 위해 여러 모달리티를 결합한 **다중특징 융합 모델**도 쓰입니다.

임계치 설정은 연속 인증의 중요 요소입니다. 즉 언제 "다른 사람"으로 간주할지 결정하는 것이죠. False Reject (본인을 틀리게 거부)와 False Accept (타인을 잘못 승인) 비율 간 트레이드오프가 존재합니다. 일반 생체인증처럼, 보안 요구에 따라 임계치를 조정합니다. 엄격 모드에서는 조금만 달라도 즉시 세션

종료, 관대 모드에서는 꽤 차이나도 일정 시간 더 관찰하는 식입니다. 때론 **시간 가중 적용**: 초반엔 관대하게 보다가 시간이 지날수록 엄격히 보는 전략도 씁니다.

연속 인증 표준 프로토콜은 아직 초기입니다. 다만 FIDO 얼라이언스에서 **"Continuous Authentication"** 개념을 2018년 whitepaper로 다룬 바 있고, ISO/IEC JTC1 SC37 (생체인증) 쪽에서도 Behavioral biometrics 표준화 연구 그룹이 활성화되어 있습니다. OASIS에도 과거 연속 인증 TC가 있었으나 표준 출판까진 이르지 못했습니다. 연속 인증 관련 특허/논문들은 다수 존재하지만, 상호운용 프로토콜보다는 **알고리즘 정확도 개선**이 주된 관심사입니다[86][91].

상용 제품으로는, Citrix, Evidian 등 몇몇 IAM 솔루션이 "세션 동안 사용자의 활동이 중단되면 재인증" 정도의 간단한 연속 검증을 제공하고 있고, BioCatch, TypingDNA 같은 스타트업들이 API로 행동 인증을 제공하기도 합니다. OS 차원에선, Windows 10의 **Dynamic Lock** 기능이 폰 Bluetooth 신호 강도를 모니터링해 사용자가 자리 뜨면 PC 잠그는 기본 기능을 제공하고, Mac OS도 Apple Watch 착용자가 멀어지면 Mac을 잠그는 Continuity 기능이 있습니다. 이들은 행동보다는 환경 변화 기반이지만 연속성의 한 형태라 볼 수 있습니다.

결과적으로 연속 인증 구현은 다소 **커스터마이징**된 형태가 주류이며, 향후 표준 인터페이스나 API (예: "연속 인증 결과 점수"를 OS가 앱에게 제공하는 표준) 등이 등장할 가능성도 있습니다.

3) 보안성 분석

연속 인증의 **보안상 장점**은, **초기 인증만으로는 방어 못하는 시나리오**까지 커버할 수 있다는 데 있습니다. 사용자가 로그인했다고 해서 그 세션이 끝날 때까지 안전하리라는 보장이 없는데, 연속 인증은 **"로그인 후"**의 보안 공백을 메워줍니다. 대표적으로 **세션 하이재킹**(Session Hijacking) 공격을 생각할 수 있습니다. 공격자가 사용자의 세션 쿠키를 탈취했거나, 네트워크상 가로챘다고

해도, 연속 인증 시스템이 **세션 내 행동이 달라진 것**을 눈치채면 즉각 차단할 수 있습니다. 예를 들어 SNS에서 평소 한국어로 메시지 쓰던 계정이 갑자기 영어로 메시지를 대량 전송하면, 행동 프로필을 벗어나므로 자동 로그아웃시키는 식입니다. 이로써 세션 쿠키 탈취나 **중간자 공격**을 어느 정도 보완하는 효과를 기대할 수 있습니다.

또한 **내부자 위협**이나 **권한 도용** 방지에도 유용합니다. 예컨대 사무실에서 자리를 비운 사이 동료가 와서 PC를 조작하려 하면, 연속 인증이 키보드 사용자 변화를 감지해 화면을 잠가버릴 수 있습니다. 실제 은행들에선 직원 PC에 이러한 **자리이탈 감지** 시스템을 설치하기도 했습니다. 그리고 **사용자 스푸핑** 공격 - 공격자가 유저의 디바이스/비밀번호를 확보해 들어와도, 연속 인증은 보다 **근본적인 사용자의 행동 특징**을 보기 때문에 추가 검증층이 됩니다. 일종의 **행동 기반 2차 인증**인 셈이죠.

보안성 면에서 연속 인증의 **성과 지표**로 흔히 **EER(Equal Error Rate)**을 봅니다. 이는 거짓거부율과 거짓허용율이 교차하는 지점으로, 작을수록 좋은 건데, 최신 연구들은 키스트로크+마우스 융합으로 EER 5% 미만까지 달성했다고 보고합니다[92]. 이는 단독 생체(얼굴 지문)만큼은 아니어도 꽤 신뢰성 있는 수준입니다. 연속 인증을 **다른 인증과 병용**하면 보안 시너지 효과가 큽니다. 예컨대 초기 로그인에 비밀번호+OTP를 쓰고, 세션 중엔 행동 인증을 돌리면, 설령 악당이 운좋게 OTP까지 통과해도 행동에서 들통날 확률이 높습니다.

한편 **제약과 단점**도 존재합니다. 첫째, **완벽 정확도 부재**입니다. 행동 기반 인증은 생체인증 중에서도 변동성이 높습니다. 사람의 타이핑이나 마우스 사용은 **컨디션, 업무 내용, 도구(키보드 종류)** 등에 따라 달라질 수 있습니다. 예컨대 오늘은 피곤해서 평소보다 타이핑이 느리다면 오탐이 날 수 있습니다. 혹은 사용자가 손 다쳐 깁스했다면 패턴이 확연히 변하겠지요. 이러한 **정상 상황 변화**를 수용 못 하면 성가신 보안으로 여겨질 것입니다. 반대로 공격자가 운 좋게도 피해자와 비슷한 타이핑 습관을 가졌다면 (가능성은 낮지만) 탐지 못할 수 있습니다. 둘째, **초기 학습 필요**입니다. 어느 정도 사용자의 충분한 행동 데이터가 누적되어야 모델이 안정화됩니다[86]. 초기 며칠은 false alarm이 많을

수 있고, 개개인별 튜닝도 필요할 수 있습니다. 또한 사용자가 **다양한 환경**에서 사용할 경우 (집, 회사 등 키보드 다름) 각각 프로파일이 필요합니다. 셋째, **하드웨어 센서 및 성능 이슈**입니다. 연속 인증은 지속 데이터 수집/분석이므로 **시스템 부하**나 **전력 소모** 문제가 있을 수 있습니다. PC에선 부담 적지만, 모바일에선 센서 상시 구동이 배터리를 잡아먹고 CPU 점유도 할 수 있습니다. 최적화 없이는 사용자 불만이 생길 수 있습니다. 넷째, **프라이버시 및 사용자 반감**입니다. 연속 인증은 사용자 행동을 **감시**하는 것이므로, 오해를 사거나 심리적 불쾌감을 줄 위험이 있습니다. 특히 근로자 PC에 연속 인증을 달면, 직원들은 "사측이 나를 상시 감시한다"고 느낄 수 있습니다. 따라서 이를 도입하려면 목적과 범위를 명확히 공지하고 동의 받아야 하며, 수집 데이터 저장도 최소화해야 합니다.

다섯째, **공격자 역이용 가능성**입니다. 만약 시스템이 어떤 행동을 기반으로 인증하는지 공격자가 알아내면, **흉내내기 공격**(Impersonation Attack)이 가능해집니다. 예를 들어 "타이핑 속도와 오류율" 정도만 반영하는 시스템이라면 공격자가 느리게 오타내며 타이핑해 속도를 맞춰버릴 수 있습니다[82]. 어느정도 흉내가 가능한 특징들은 그만큼 취약해집니다. 이를 막으려면 시스템이 어떤 특징들을 쓰는지 **비밀로 유지**하거나, 복잡한 다변량 패턴을 사용해야 합니다. 여섯째, **분석 회피 공격**도 생각할 수 있습니다. 예컨대 공격자가 PC에 RAT(Remote Access Trojan)을 심어 원격으로 제어하면, 원격 제어는 종종 로컬 사용자 패턴과 차이가 납니다. 하지만 공격자는 RAT 도구를 "로컬 입력처럼" 작동하도록 교묘히 설정할 수 있습니다. 혹은 아예 사용자의 세션 쿠키 탈취 후 **API 호출**만으로 악용하면(사용자 UI를 거치지 않고), 행동 인증은 감지를 못할 수 있습니다. 이러한 경우, 연속 인증만 의존해선 안 되고 다른 세션 무결성 보호 (예: 네트워크 레벨 모니터링)와 결합해야 합니다.

4) 활용 사례 (산업별 적용 현황)

연속 인증은 현재까지 **주로 특수 고보안 환경**이나 **실험적 솔루션**에 적용되어 왔지만, 최근 상용 서비스에도 일부 도입되는 추세입니다. **군사/국방 분야**에서는 몇 년 전부터 연속 인증 개념을 테스트하고 있습니다. 미 국방부 DARPA는

Active Authentication 프로그램으로 키스트로크 기반 군인 PC 연속 인증을 연구했고, 미 공군은 2019년부터 일부 내망에 **Continuous Multi-factor Authentication** 솔루션을 시범 적용하여, 로그인 후 15분마다 사용자 확인(지문 재확인 등)하는 방안을 운영했습니다. 또한 미 에너지부 산하 연구소 등에서도 연구원 시스템에 행동 모니터링을 도입해 내부자 위협을 줄이는 시도를 하고 있습니다[91].

금융권에서도 일부 활용이 보입니다. 예컨대 러시아 Sberbank는 2018년 **Continuous Behavioural Biometric Authentication** 기술을 모바일 앱에 넣어, 사용자의 스마트폰 잡는 자세, 터치 습관 등으로 계정 소유자 여부를 지속 판단하는 기술을 발표했습니다. 한국의 일부 증권사 HTS에서도 연속 인증 개념을 도입해, 사용자가 자리 비울 때 자동 로그아웃, 매매 패턴이 평소와 다르면 위험 경고 하는 기능을 넣은 사례가 있습니다. **스마트폰 잠금 해제**에도 연속 인증 개념이 접목된 사례가 있습니다. 안드로이드의 Smart Lock 기능 중 **On-Body Detection**은 기기가 움직이고 있는 동안은 사용자 손에 있다고 가정해 잠금이 유지되도록 하고, 내려놓으면 잠그는 간단한 연속 검증입니다. 또한 **웨어러블 연동**(시계와 폰이 일정 거리 이내면 잠금 유지, 멀어지면 잠금)도 연속적인 존재 확인 개념입니다.

기업 보안 측면에서, 일부 기업들은 직원 VPN 연결 시 정기 재인증을 요구하거나, 클라우드 대시보드 등에서 일정 시간마다 세션 연장을 위해 OTP 재입력을 시키는 식으로 연속 인증의 "명시적" 버전을 실행 중입니다. 보다 투명한 방식으로는, 한국의 한 대형 은행은 내부 망 PC에 BioCatch의 키스트로크 분석 소프트웨어를 설치해, 직원 로그인 세션 중 계속 키 입력을 분석하고 자기 사람이 아니면 경고하는 시스템을 시범 가동한 바 있습니다.

시험 감독/온라인 교육 분야에서도 연속 인증 활용이 늘고 있습니다. 코로나 시기 이후 원격 시험이 많아지면서, 응시자가 본인인지 계속 확인하기 위해 시험 소프트웨어가 **카메라로 얼굴 주시** 및 **타이핑 모니터링**을 합니다. 예컨대 Duolingo English Test 같은 온라인 시험은 수험자 얼굴을 AI로 실시간 추적해 다른 사람이 대리 응시하지 못하게 하고, 시선추적 등으로 부정행위를 감시

하는데, 이 역시 일종의 연속 인증/검증이라 볼 수 있습니다.

모바일 결제에서도, Apple은 Apple Pay 트랜잭션 시마다 FaceID/TouchID로 본인 확인을 요구하여 일종의 "연속 인증"(매 결제 시 인증)을 수행합니다. Google은 지문 센서 탑재 안드로이드폰에 **"Trust Mode"**라는 개념을 넣어, 기기를 내려놓았다 들면 패턴/PIN을 다시 묻도록 한 적도 있습니다.

전반적으로, 연속 인증은 아직 대중화 단계는 아니나, **고강도 보안 요구 환경**에서 서서히 실용화되고 있습니다. 또 많은 보안 연구와 특허가 축적되어 있어, 향후 기술 성숙도가 올라가면 **일반 기업용 IAM** 솔루션에도 포함될 가능성이 큽니다. 실제 Microsoft Entra(옛 Azure AD) 로드맵에도 Continuous Access Evaluation이 추가되고 있고, Okta도 2023년 새로운 risk engine에 **session monitoring** 기능을 언급했습니다. **제로트러스트 도입 기업**들은 "한번 로그인했다고 끝 아니다"에 동의하기 때문에, 연속 인증 아이디어를 MFA와 조합해 내부 규정에 반영하는 추세입니다.

5) 기술적/운영상 장단점

연속 인증의 **장점**은 앞서 논의한 대로 **세션 보안성의 비약적 향상**입니다. 인증 행위를 한 번에 그치지 않고 **지속 확인**하므로, **세션 하이재킹, 사용권 위임 등의 위험**을 실시간 차단할 수 있습니다. 결과적으로 **인증 신뢰도**가 세션 전체에 걸쳐 유지되며, 공격자가 뚫어야 할 관문이 상시 존재하게 됩니다. 또 다른 장점은 **투명한 보안**을 구현 가능하다는 점입니다. 사용자가 느끼지 못하는 방식(예: 키 입력)으로 인증하는 경우, 사용자는 불편 없이 보호를 받습니다. 이는 사용경험(UX)을 크게 해치지 않는다는 이상을 실현할 수 있습니다. 특히 행동 기반 연속 인증은 **"사용자가 곧 토큰"**이라는 개념이라, 비밀번호나 OTP처럼 기억하거나 소지할 것 없이 그냥 일상 행동으로 인증되니 **편의성**이 좋습니다 [88].

세 번째 장점은 **보안 이벤트 대응 속도**입니다. 연속 인증이 없다면, 공격이 진행되어도 관리자가 로그 분석으로 나중에 알아챌 것입니다. 하지만 연속 인증

은 자동화된 실시간 탐지라서, 이상 발생 수 초 내로 세션을 끊어버릴 수 있습니다. 이는 피해 확산을 최소화합니다. 네 번째, **사후 인증 감사 근거**를 남긴다는 이점도 있습니다. 모든 세션에 대해 행동 로그와 인증 점수가 쌓이므로, 나중에 "이 세션 중간에 사용자가 바뀌었다"는 forensic 분석 근거가 됩니다. 다섯째, **다단계 인증 대체 가능성**입니다. 만약 연속 인증 기술이 매우 정확해진다면, 처음부터 MFA를 줄이고 행동 인증만으로 진행하게 할 수도 있습니다. 즉 **사용자 부담을 줄이면서** 보안을 유지하는 형태입니다. 실제로 TypingDNA 같은 회사는 "비밀번호 입력하는 동안의 타이핑으로 2FA를 대체"하는 제품을 제안합니다.

연속 인증의 **단점**과 한계도 분명합니다. 첫째, **기술 미성숙과 구현 복잡성**입니다. 행동 생체 기술은 전통 생체만큼 상용화가 안 되어 있고, 환경 변화에 민감합니다. 이를 바로 서비스에 적용하면 오작동 우려가 있어, 상당한 튜닝과 사용자 교육이 필요합니다. 이와 관련한 둘째 단점은 **False Reject 문제**입니다. 본인이 정상적으로 쓰고 있는데도 시스템이 오인해 차단하면, 사용자에게 극심한 불편을 줍니다. 예컨대 중요한 업무 중인데 갑자기 시스템이 꺼지거나, 은행 이체 중인데 "당신 실제 주인 맞음?" 하고 틀렸다고 거래를 취소하면 신뢰를 잃게 됩니다. 잦은 오탐은 연속 인증을 폐기하도록 압력을 줍니다.

셋째, **도입 비용과 성능 부담**입니다. 연속 인증은 전문 솔루션 도입 또는 개발이 필요하고, 그것을 운영하려면 추가 인프라(예: 행동 데이터 서버)와 연산 능력이 필요합니다. 소규모 서비스에는 과할 수 있습니다. PC 한 대 한 대에 에이전트를 까는 것도 관리포인트 증가입니다. 네번째, **사용자 사생활 침해 우려**입니다. 연속 인증은 일거수일투족을 감시한다고 느껴질 수 있습니다. 특히 직원들은 "화장실 간 사이까지 감시당한다"는 불만을 가질 수 있어, 노동조합 이슈나 인권 문제로 비화될 가능성도 있습니다. 이러한 우려를 불식시키지 못하면 조직에 도입이 어렵습니다.

다섯째, **범용 적용의 어려움**입니다. 행동 패턴 인증은 사람마다 다르기에 **개인별 맞춤 모델**이 필요하고, 사용 환경도 제각각입니다. 어떤 사람은 마우스만 쓰고 어떤 사람은 키보드 단축키를 즐겨 쓰는 등, 전부 모델링 하기가 쉽지 않

습니다. 그러다 보니 특정 업무나 특정 애플리케이션 범위 내에서는 통계가 가능해도, 범용 OS 수준에서 적용하긴 난제가 많습니다. 여섯째, **법적/윤리적 책임 문제**입니다. 만약 연속 인증 시스템이 누구를 잘못 내쫓는 바람에 큰 피해가 났다면(예: 의료 수술 로봇 시스템에서 연속 인증이 오작동하면?), 그 책임 소재가 모호합니다. 아직은 첨단기술 영역이라 표준이나 법적 고려가 부족합니다.

마지막으로, **공격 루프홀** 역시 고려해야 합니다. 일부 공격은 연속 인증을 겉보기엔 통과하면서 악용할 수 있습니다. 예컨대 RDP로 접속해 원격으로 피해자가 현재 쓰는 PC를 미러링해 조작하면, 로컬 입력 패턴과 거의 동일하게 보일 수 있습니다(실제로 APT 공격에서 이런 수법도 관찰됨). 이런 경우엔 연속 인증만으로 막긴 힘듭니다. 또 얼굴인식 continuously 하는 시스템은 **딥페이크 영상**으로 카메라를 속일 위험이 있습니다.

운영상의 문제로는, 연속 인증 시스템이 도입되면 **사용자 지원 업무**가 늘 수 있습니다. "왜 내 세션이 끊겼나" 문의가 들어오면 해명해줘야 하고, 언제나 소수이지만 genuine user가 차단될 수밖에 없기 때문에 이를 처리할 헬프데스크 절차를 마련해야 합니다.

6) 글로벌 주요 표준 및 정책 흐름

연속 인증은 아직 명문화된 글로벌 표준이 없지만, 개념 자체는 여러 가이드라인에 반영되어 있습니다. **제로트러스트** 관련 정책들은 "**연속적 신원 검증**"을 기본 원칙으로 강조합니다[72]. 미 연방의 2021년 사이버행정명령과 Zero Trust 전략 문서들은 연속 인증(Continuous Validation)을 목표로, 연방 기관들에 **간헐적 재인증** 기술 채택을 요구하고 있습니다. 미 상무부 NIST도 연속 인증 기술 연구에 관심을 보여, 2019년 "Zero Trust Architecture" 초안에서 **Continuous Authentication** 구현 방안 사례를 언급했습니다.

표준화 단체 측면에서, **FIDO 얼라이언스**는 아직 연속 인증 프로토콜을 표준으로 만들진 않았지만, 2022년 White Paper에서 FIDO 인증 수단과 연속 리스

크 평가의 결합을 제시했습니다. 또한 FIDO 3.0 비전 토론에서 **"컴퓨터 사용 중 틈틈이 지문 인증 센서를 터치하게 하여 세션 유지"** 같은 아이디어도 논의되었습니다. **ISO/IEC JTC1 SC37** (바이오메트릭스) 내에서는, 행태 생체(behavioral biometrics)에 관한 기술보고서들이 발간되었고, SC27 (보안기술)에서도 "User Continuous Authentication"이라는 연구 항목이 논의된 적 있습니다. 그러나 아직 구체적 표준 문서로 이어지진 못했습니다. **ETSI**나 **OASIS** 같은 곳에서도 formal effort는 아직 관찰되지 않습니다.

다만 **IEC TC9**(전자 철도 표준)에서는 2019년 기차 운전사 연속 인증 관련 제안이 있었고, **ICAO**(국제민간항공기구)에서도 공항 출입 통제에 한 번 얼굴인증 후 게이트간 이동 시 연속 검증을 고려하는 논의가 있었습니다. 이런 특정 vertical 분야에서 필요성을 인지하고 가이드라인 만들 가능성이 있습니다.

정부 정책 측면에서는, 아직 연속 인증 도입을 법으로 요구하는 곳은 없으나, 인도 같은 나라의 일부 기밀 인프라 가이드에 행동 생체 인증 권고가 포함된 사례가 있는 것으로 전해집니다. 우리나라에서도 2021년 발간된 금융보안원 보고서 등에서 연속 인증 기술을 미래 인증 수단으로 소개하고 금융권 활용을 검토할 것을 제안하기도 했습니다.

윤리/규제 측면에서 걱정되는 **프라이버시** 문제는 개인정보보호법들이 연속 인증 데이터도 보호대상으로 간주하고 엄격 관리토록 하는 정도입니다. EU GDPR 아래, 행동 생체 정보는 **생체 데이터**로 분류될 소지가 있어 (식별 목적으로 사용되므로) 명시적 동의와 보호 장치가 요구됩니다. 연속 인증 도입 시 이와 관련된 법률 준수를 검토해야 하며, 유럽 어떤 기업은 직원 연속 인증 도입이 GDPR 위반인지 법적 자문을 구하기도 했습니다.

향후 표준화 가능성으로, **CISA**(미국 사이버안보국)나 NIST가 연속 인증 best practice를 발행하거나, **공공조달 기준**에 "연속 사용자 검증 기능 포함"을 요구할 수 있습니다. 이는 산업계 표준을 사실상 만드는 효과가 있습니다. 또한 기술적으론, **W3C**가 웹 환경에서 연속 인증 API를 논의할 가능성도 있습니다. 예를 들어 WebAuthn 향후 버전에 사용자presence를 주기적으로 확인하는

extension을 넣는 식입니다.

요약하면, 연속 인증은 아직 "모범사례" 단계이지만, **제로트러스트 보안전략**에 필수 요소로 점점 자리잡아가는 중이며, 표준은 행동 생체 분야를 통해 서서히 기반이 마련되고 있는 상황입니다. **정책 방향**은 궁극적으로 "1회성 인증에서 상시 인증으로" 나아갈 것이라는 점은 분명해 보입니다.

7) 향후 발전 전망 및 과제

연속 인증은 미래 보안에서 중요한 역할을 할 것으로 예상되며, 기술적 발전과 함께 실용화 과제가 추진될 것입니다. **발전 전망** 몇 가지를 살펴보면:

1. **AI 기반 다중모달 인증**: 현재는 키스트로크, 마우스 등 개별 신호 위주지만, 앞으로는 **AI가 다양한 센서 데이터를 종합 분석**하여 훨씬 정교하게 사용자 식별을 할 것입니다[93][94]. 예컨대 PC에서는 키보드+마우스+얼굴표정, 모바일에서는 터치+가속도센서+환경소음 등을 동시에 고려하는 **멀티모달 딥러닝 모델**이 개발될 것입니다. 이는 정확도를 크게 높여 false reject를 줄여줄 것으로 기대됩니다.

2. **인공지능 공격 대응**: 공격자도 AI를 이용해 사용자 행동을 흉내낼 수 있게 될 것입니다. 이에 대비해 **대항 AI 모델**, 즉 공격자의 모방행위를 탐지하는 모델이 필요합니다. 예를 들어 연속 인증 시스템은 사람이 입력하는 것과 봇이 입력하는 것을 미세한 차이(타이밍 정규성 등)로 구분하는 식의 **대적학습(Adversarial Training)**이 도입될 수 있습니다.

3. **사용자 맞춤형 인증 전략**: 모든 사용자가 연속 인증에서 동등하게 취급될 필요는 없습니다. 향후 시스템은 사용자의 과거 신뢰도나 선호도에 따라 **개인화된 연속 인증** 전략을 적용할 수 있습니다. 예컨대 보안 민감도가 높은 사용자는 더 엄격하게, 아닌 사용자는 약하게 적용하거나, 혹은 사용자가 "나는 조금 귀찮아도 강한 보안 원해"라고 설정하면 더 자주 검증하는 옵션도 가능할 것입니다.

4. **프라이버시 보호 기술 접목**: 연속 인증의 데이터를 프라이버시 침해 없

이 활용하기 위해, **연합 학습**이나 **로컬 인퍼런스** 같은 기술이 발전할 것입니다[81]. 예컨대 여러 기관이 행동 데이터를 모아 공동 학습하되, 원데이터는 공유 안 하고 모델만 교환하여 성능을 높이는 방식입니다. 또는 사용자의 raw 행동 데이터는 로컬에서만 처리되고 서버에는 결과 점수만 보내서 사생활 노출을 줄이는 아키텍처가 표준화될 수 있습니다.

5. **하드웨어 지원**: 현재는 소프트웨어로 구현하지만, 미래엔 CPU나 주변 장치 자체에 연속 인증 기능이 내장될 가능성도 있습니다. 예를 들어 키보드에 타이핑 특징을 분석하는 마이크로컨트롤러 탑재, 마우스에 사용자의 그립/압력 센서 내장 등이 현실화되면, 더 신뢰성 있는 데이터 수집과 낮은 오버헤드로 연속 인증이 가능해질 것입니다.

6. **일상 생활로의 확장**: 연속 인증 개념은 컴퓨터 뿐 아니라 출입통제, 자동차 운전 인증 등 **피지컬 월드**로도 확장될 전망입니다. 예컨대 차량의 운전자 부주의 방지 시스템이 운전자 얼굴/행동을 모니터링하는 것은 연속 인증의 한 사례입니다. 또 스마트홈에서 집 안 사람의 동작/음성으로 주인 여부를 실시간 판단해 도난을 막는 기술도 생각할 수 있습니다.

과제로는, 첫째 **정확도 극대화**와 **오류 최소화**가 핵심입니다. 이를 위해 대규모 실제 사용자 데이터셋 구축과 알고리즘 개선이 필요합니다. EER을 1% 미만으로 줄이는 것이 목표가 되어야, 실무 적용에서 신뢰를 얻을 수 있습니다. 둘째 **사용자 경험 개선**입니다. 비정상 시나리오 외에는 사용자가 연속 인증의 존재조차 의식 못하는 방향이어야 합니다. 당장은 어려워도, 장기적으로는 연속 인증이 사용자의 편의까지 돕는, 예컨대 "사용자 특성을 인지해 UI 최적화" 같은 플러스 요인이 되도록 연구될 수도 있습니다. 셋째 **표준과 상호운용** 문제 해결입니다. 현재 여러 기관에서 제각각 개발하는 상태라, 호환성 없는 솔루션 파편화가 우려됩니다. 국제 표준 수립과 테스트베드 운영으로, 어느정도 통일된 프레임워크 (예: 점수 범위 정의, 공통 벤치마크 시나리오 등)를 만들어야 할 것입니다. 넷째 **정책 및 법규 수용**입니다. 연속 인증을 도입하려면 개인정보 규정과 노동법 등을 준수해야 합니다. 이에 대한 명확한 가이드라인을 마련하고, 이해관계자의 우려를 불식시키는 노력이 중요합니다. 다섯째 **보안 대책 이중화**입니다. 연속 인증이 도입되더라도 기존 초기 인증과 MFA가 완전히 대체

되는 것은 아니므로, 이들 간 역할을 어떻게 분담하고 충돌 없이 운영할지 가이드가 필요합니다.

마지막으로, **사용자 수용성**을 높이는 것이 관건입니다. 기술이 좋아도 사용자가 거부하면 소용없습니다. 따라서 연속 인증 시스템은 사용자에게 **가치**를 전달해야 합니다. 이를테면 "편하게 계속 일하세요, 위험하면 우리가 지켜줄게요"라는 메시지가 전달되어 안심을 준다면, 사용자가 긍정적으로 생각할 것입니다. 반대로 "계속 지켜보고 있어요" 느낌이면 반발이 생길 것입니다. 이 **심리적 측면**까지 고려하는 디자인과 운영이 연구되어야 연속 인증이 광범위하게 자리잡을 것입니다.

요약하면, 연속 인증은 **언제 어디서나 지속되는 보안 확인**이라는 개념으로, **AI 시대 인증 패러다임**의 중요한 축이 될 전망입니다. 다만 기술적 완성, 프라이버시 대책, 사용자 인식 개선이라는 3박자가 맞아야 실현될 것이며, 이것이 풀린다면 미래에는 별도 로그인 과정 없이도 기계가 사람을 계속 인식하여 "항상 로그온" 상태를 안전하게 유지하는, **매끄러운 인증 경험**이 가능해질 것입니다.

사. 프라이버시 보호 인증 (Privacy-Preserving Authentication)

1) 기술 개요 및 작동 원리

프라이버시 보호 인증은 사용자 신원을 확인하는 과정에서 **개인정보 노출을 최소화**하고, 인증 행위 자체로 인한 트래킹이나 민감정보 유출을 막도록 설계된 인증 기술들을 말합니다[95]. 전통적인 인증(비밀번호, 바이오메트릭 등)은 사용자가 누구인지 식별하거나 비밀을 교환하는 과정에서 개인 식별정보(이메일, 지문 등)가 드러나는 경우가 많았습니다. 프라이버시 보호 인증 기술은 이러한 식별정보를 직접 주고받지 않고도 **필요한 증명만 수행**하거나, 주고받더라도 **암호화 및 불가추적성**을 보장함으로써 **익명성**이나 **비연결성(unlinkability)**을 달성하는 것이 목표입니다[96].

가장 핵심적인 원리 중 하나는 **영지식 증명(Zero-Knowledge Proof, ZKP)**입니다. 영지식 증명은 **어떤 비밀이나 속성에 대해 참임을 증명하되, 그 내용은 전혀 공개하지 않고도 검증을 통과**시키는 기술입니다[44]. 예를 들어 사용자가 18세 이상인지 확인할 때, 보통 신분증 전체를 보여줘 생년월일 등 불필요한 정보까지 노출됩니다. 하지만 ZKP를 쓰면, 사용자는 자신의 생일 정보를 토대로 "내 나이가 18 이상임"을 암호학적으로 증명하고 검증자에게 그 증명값만 제시하여, 실제 생일은 알려주지 않고도 성인 여부를 확인시킬 수 있습니다[44]. 이러한 ZKP 원리는 다양한 프라이버시 보호 인증 프로토콜의 근간을 이룹니다.

또 다른 핵심 기술은 **익명 인증서/자격증명(Anonymous Credentials)**입니다[97]. 이는 사용자가 어떤 권한이나 자격을 인증받을 때, **실제 아이덴티티를 드러내지 않고** 인증기관이 발행한 **익명 토큰/증명서**로 대신 증명하는 방법입니다. 대표적인 사례로 **Group Signature(그룹 서명)**가 있습니다. 그룹 서명에서는 특정 그룹(예: 직원 전체)의 구성원들에게 하나의 그룹 공개키에 대응하는 각기 다른 개인키를 주고, 구성원은 그룹 서명을 생성하면 **누가 서명했는지는 드러나지 않지만 그룹 멤버 중 한 명인 것은 검증**되게 합니다[97]. 이를 이용하면 "이 사용자는 우리 회사 직원임"을 증명하되 구체적으로 어떤 직원인지는 숨길 수 있습니다. IBM의 **Idemix**, Microsoft의 **U-Prove** 등은 이러한 개념의 구현으로, 익명화된 인증 티켓을 활용합니다.

동형 암호(Homomorphic Encryption)도 프라이버시 보호 인증에 활용됩니다[63]. 동형 암호는 암호화된 상태로 연산이 가능한 기술로, 이를 통해 서버가 사용자 비밀(예: 생체 특징)의 암호문을 받아 **복호화 없이 매칭 연산**을 수행하고, 결과만 확인하는 식입니다[63]. 즉 서버는 사용자의 실제 지문이나 얼굴 데이터를 전혀 볼 수 없고, 암호문 상태로 비교해 인증 여부만 알게 되는 것입니다. 이 역시 인증 과정에서 개인정보 노출을 제거하는 강력한 원리입니다.

페더레이션/토큰 기반 인증 구조에서도 프라이버시를 강화하는 흐름이 있습니다. OpenID Connect 같은 프로토콜에 **"가명 식별자(pseudonymous**

identifier)" 개념이 있어, IdP가 각 RP(서비스)마다 다른 난수 ID를 발급해 동일 사용자가 여러 RP에서 추적당하지 않게 합니다. 예를 들어 OIDC pairwise 설정을 쓰면 RP1에서는 사용자가 alice123로 식별되고 RP2에서는 bob987로 식별되어 둘이 동일인인지 RP들이 알 수 없게 됩니다. FIDO2(WebAuthn)도 사이트마다 고유한 계정 ID(공개키 식별자)를 생성하여, 사이트 간 사용자를 매칭하기 어렵게 해놨습니다. 이러한 **비연결성(Unlinkability)** 원칙이 프라이버시 보호에 중요합니다[22].

종합적으로, 프라이버시 보호 인증에서는 **"필요한 최소한의 정보만 증명"**한다는 모토가 적용됩니다[44]. 이때 사용자에게 **데이터 사용 통제권**을 주는 것도 중요합니다. 예를 들어 어떤 서비스에 로그인 시 나의 신원 전체가 아니라 **특정 속성**(연령, 소속 등)만 선택적으로 제출하거나, 혹은 **익명 세션**으로만 접근하게 하는 옵션을 줍니다[22]. 이를 지원하는 백엔드 기술이 앞서 말한 ZKP, 익명 증명서 등입니다.

구체적인 작동 흐름은 예시로 다음과 같습니다: 사용자는 **신뢰기관(IdP)**으로부터 **속성 증명서(Attribute Credential)**를 하나 발급받습니다 (여기에는 "나이는 20대, 시민권 있음" 등 포함). 이 증명서는 **사용자에게만 풀 수 있는 암호**로 보호되거나, 클라이언트 서명으로만 제시 가능합니다[44]. 사용자가 서비스(RP)에 접속해 인증할 때, 평소처럼 자기 신원 공개하지 않고, **증명서와 영지식 증명**으로 필요한 속성만 입증합니다. 예컨대 쇼핑몰에서 주류 구매시 "나 19세 이상임"을 IdP로부터 받은 증명서와 ZKP로 증명하고, 쇼핑몰은 그 증명을 검증해 OK 처리하지만 고객의 이름, 생일, 주소 등은 전혀 알 수 없는 것입니다. 이때 IdP는 각 RP에 다른 토큰을 줘 **동일 사용자인지 모르게** 하므로, 쇼핑몰A와 쇼핑몰B가 고객 정보를 대조해 "같은 사람"임을 알 길이 없습니다[22]. 또한 증명서에는 **만료 및 철회(revocation)** 메커니즘이 있어, 사용자가 자격을 잃으면 IdP가 철회리스트를 업데이트하여 더 이상 증명이 안 통하게 합니다[97].

이상의 기술로 달성되는 속성들은 **익명성(Anonymity)** - 누군지 모른다, **비연결성** - 여러 세션을 이어볼 수 없다, **불가능증명성(Denial)** - 사용자가 추후 자신

이 했다는 증명을 부인할 수 있는지 여부 (어떤 상황에선 원하면 부인 가능하게 해주기도 함), **최소 공개** - 인증에 불필요한 데이터는 안 주는 것 등입니다 [44][22].

2) 주요 구현 방식 및 프로토콜

프라이버시 보호 인증을 위한 구현으로 몇 가지 유명한 프로토콜과 라이브러리가 있습니다. **Idemix (Identity Mixer)**는 IBM Research에서 개발한 익명 자격증명 시스템으로, **CL 서명** (Camenisch-Lysyanskaya 서명)이라는 특수한 서명 방식을 사용해 **선택적 속성 공개**와 영지식 증명을 지원합니다. Idemix에서는 사용자에게 발급된 증명서에서 원하는 항목만 **Zero-Knowledge로 증명**하거나, 항목들 간의 논리식(예: 나이>18)을 증명하는 것이 가능합니다. **U-Prove**는 Microsoft가 개발한 또 다른 익명 크레덴셜 기술로, 사용자에게 **블라인드 서명된 토큰**을 주고 사용자 스스로 필요한 속성을 넣어 발행할 수 있게 합니다. 이 또한 특정 속성만 제시하는데 유용하며, 구조가 단순해 빠른 장점이 있습니다.

페더레이션 프로토콜 확장도 있습니다. **OpenID Connect**의 **"Selective Disclosure JWT (SD-JWT)"** 표준화가 진행 중인데, 이는 OIDC ID 토큰 내에 암호화된 속성들을 넣어 RP에 따라 일부만 공개될 수 있도록 하는 것입니다. 예컨대 SD-JWT를 쓰면 IdP가 사용자 이름, 생일, 이메일을 ID 토큰에 주는데 이것들이 각각 해시+증명 형태로 들어가 RP가 요구하는 것만 풀어볼 수 있게 합니다. 이 표준이 완성되면 기존 OIDC 로그인을 통해서도 최소정보 인증이 가능해집니다.

Zero-Knowledge Proof 분야에서, 라이브러리들(ZK-SNARKs, ZK-STARKs 등)이 발전해 웹에서도 ZKP가 사용 가능해졌습니다. ZKP를 직접 프로토콜에 넣은 예로, **TLS 1.3**의 시도 중 하나로 **Zero-Knowledge Password Proof (ZRTP)**라는 제안이 있었는데, 이는 비밀번호를 서버에 보내지 않고 영지식하게 검증하는 PAKE(Password Authenticated Key Exchange)의 한 형태입니다. PAKE (예: SRP, SPAKE2) 자체도 프라이버시 보호 인증의 구현이라 볼

수 있습니다 - 사용자는 비밀번호를 평문으로 안 보내고도 서버랑 공유 비밀을 성립시키죠.

또 **생체인증 템플릿 보호** 기술들도 프라이버시 보호 맥락입니다. ISO/IEC 24745는 **Biometric Template Protection**으로, 생체 특징을 원상복구 불가능하게 암호 변환하거나, **객체 암호화 상태로 비교**(Homomorphic encryption matching)하도록 권고합니다[15]. 이를 구현한 FIDO 등의 표준에서는 지문 데이터를 AES로 암호화한 뒤 매칭하거나, 생체정보에서 유래한 해시만 사용하고 원본은 폐기하는 방식이 있습니다.

Privacy Pass 프로토콜(IETF RFC 8941)도 흥미로운 구현입니다. Cloudflare 등이 쓸 목적으로 만든 것으로, 사용자가 CAPTCHA를 풀면 **익명 토큰**을 발급받아 다른 사이트에 증명으로 제출해도 사용자를 추적할 수 없게 했습니다. 이는 **Blind Signature(맹목 서명)** 기술로 구현되어, 서버가 유저를 식별 못하는 토큰을 서명해줍니다. Apple의 **Private Access Token**도 이 프로토콜을 기반으로 하여, iOS 기기에서 **사용자 IP나 계정 정보 노출 없이** 클라이언트가 신뢰할 만한 기기임을 웹사이트들에 증명하도록 합니다.

Blockchain & SSI (Self-Sovereign Identity) 분야 역시 프라이버시 보호 인증과 맥을 같이 합니다 (블록체인 기반 인증은 앞서 별도 장에서 다룸). W3C **Verifiable Credentials**과 **Decentralized Identifiers** 표준은 기본적으로 사용자가 **필요한 속성만 담긴 VC**를 제출하도록 하고, DID 자체도 사이트마다 다른 것으로 발급 가능해 **상호연결이 어렵게** 한 것이 설계 목표입니다[55][98]. 또한 **BBS+ 서명**이라는 ZKP 친화적 서명 방식으로 VC 내 특정 claim만 공개 증명하는 기능도 추가되고 있습니다.

사설 영역에서도, 프라이버시 보호 인증의 구현은 자주 등장합니다. 예를 들어 Google은 2022년 **"Private Federated Login"**이라는 연구를 공개했는데, IdP와 RP가 연합 로그인 시 사용자 식별자를 암호화 교환하여 **양쪽도 특정 사용자가 어떤 RP들에 로그인하는지 알 수 없게** 하는 방식을 제안했습니다. 이는 PAIR (Private Authentication of Identity References)라는 암호기술을 활용

했다고 합니다.

익명화 네트워크(Tor 등)와 인증을 결합하려는 시도도 있습니다. Tor는 익명성 보장하지만 기존 로그인하면 의미가 없어집니다. 그래서 **Anonymous Credential + Tor**로 로그인을 해도 서비스가 사용자 진짜 ID는 모르고 권한만 확인하게 하는 프로젝트들도 연구되었습니다.

이렇듯 구현 방식은 다양하나, 공통으로 **암호기술(대칭, 비대칭, 영지식 등)을 활용해 필요 정보만 주고받는** 패턴입니다. 대부분은 **클라이언트 측에 추가 SW** (예: 지갑앱, 브라우저 extension) 필요하고, **서버 측에 검증 라이브러리**가 필요한 등 일반 인증보다 복잡합니다.

3) 보안성 분석

프라이버시 보호 인증의 보안성은 양면이 있습니다. **개인정보 보호 측면**에서는 사용자의 민감정보가 줄어들고, 추적이 어려워지므로 **프라이버시 위협에 대한 보안**이 향상됩니다[44]. 그러나 **인증 자체의 보안 강도** 측면에선, 기술 구현을 잘못하면 취약점이 생길 수도 있습니다. 몇 가지 분석해보겠습니다:

장점/강점: 우선, 사용자의 비밀이 노출되지 않으므로, **"인증 과정 유출 -> 이후 공격"** 시나리오를 차단합니다. 예를 들어 비밀번호 기반에서는 서버 DB 해킹으로 해시가 유출되고 크래킹되면 끝인데, 영지식 패스워드 인증(PAKE)을 쓰면 서버에 해시도 없고 네트워크에도 힌트가 안 남으니 이러한 공격면이 없습니다[44]. 또 **피싱 공격**에도 강합니다. 피싱 사이트가 영지식 인증을 흉내내기는 어렵고, 설령 유저 속여도 얻어낼 수 있는 정보가 없기 때문입니다. 예컨대 사용자에게 "당신 18세 이상임 증명해주세요" 했을 때, 영지식 값 하나 받아봐야 다른 사이트에서 쓸 수 없습니다. **재전송 공격**에도 안전한 경우가 많습니다. 증명 값은 1회용 nonce를 포함해 만들어지므로, 도용해봤자 쓸모없습니다.

또한 **대규모 프라이버시 침해 공격**이 무력화됩니다. 흔히 공격자가 여러 서비스의 로그를 모아 사용자 행태를 분석하거나, 광고주들이 SSO 로그를 공유해

사용자 프로파일링하는 일이 있는데, 프라이버시 보호 인증은 **데이터 연계성**을 끊어버리므로 이런 **빅브라더형 공격**에 대한 방어입니다[22].

사용자 신분 숨김은 보안성과 때로 상충하지만, 일부 시나리오에서는 오히려 보안 이점이 될 수 있습니다. 예를 들어 정부 민원 서비스를 익명 인증으로 접근 가능하게 하면, 공격자가 특정인을 노려 그 사람 민원 내용을 염탐하는 게 어려워집니다.

한계/단점: 첫째로, 체계가 복잡해지므로 **오류 가능성**이 높습니다. 영지식 증명, 익명 credentials 등은 구현이 까다롭고, 설계 실수나 코너케이스에서 **논리적 취약점**이 생길 수 있습니다. 실제로 초기 U-Prove에서는 프라이버시엔 좋지만 이중발행 방지를 위해 추가 메타데이터가 필요했고, Idemix도 철회 구현이 복잡해서 공격 표면이 있었습니다. 즉 **새로운 공격면**이 생깁니다. 예컨대 영지식 증명에서는 **동기화 공격**: 증명값을 여러 서비스에서 동시에 써서 식별자를 추출하는 등 (Idemix는 이 방지 위해 증명마다 퍼머넌트 판별자 제한, 하지만 그것도 프라이버시 vs 보안 트레이드오프).

둘째, **인증 강도** 자체는 비슷하나, 추후 **책임 추적성** 이슈가 있습니다. 프라이버시 보호 특성상, 나중에 **누가 로그인했는지 알기 어렵게 설계**되어, 만약 부정사용이 발생해도 **식별/추적이 어려워집니다]**[99]**. 이는 내부 보안 관점에선 단점일 수 있습니다. 그래서 어떤 시스템은** 조건부 익명성**을 넣어, 법적 명령에 의해만 익명 해제가 가능토록 (그룹 서명에 마스터키로 서명자 밝혀내기 등) 설계합니다. 그러나 이것 역시 백도어 논란이 있어 balancing이 어려움입니다.

셋째, **성능 오버헤드**입니다. ZKP, 동형암호 등은 계산이 무겁고, 통신도 추가 왕복이 생길 수 있습니다[63]. 이로 인해 시간 지연이 증가하면, 공격자가 **타이밍 분석**으로 뭔가 유추 가능해질 위험도 있습니다.

넷째, **상호운용 어려움**에 따른 보안 약화 가능성입니다. 프라이버시 보호 인증 체계를 지원하지 않는 서비스가 많으면, 사용자는 그들과 호환 위해 전통 인증

도 써야 합니다. 그럼 **보안 취약한 예전 방식**과 병용해야 하므로 전체 보안 수준은 낮은 쪽에 끌려갑니다.

다섯째, 관리자가 사용자 활동을 익명화하면, **어뷰저 차단**이 까다롭습니다. 예를 들어 온라인 게시판을 완전 익명 인증으로 운영하면, 악성 유저를 추방하거나 다중 계정 악용을 막기 어려워질 수 있습니다. 이를 위해 **익명 속 평판 시스템** 같은 추가 보완이 필요합니다.

여섯째, **암호 primitive 안전성** 의존도가 높습니다. ZKP나 블라인드 시그 등의 안전은 수학적 가정에 의존하는데, 만약 미래에 그 알고리즘 깨지면, 한꺼번에 privacy 인증 구조가 무너질 수 있습니다. (물론 전통 RSA/ECC도 마찬가지지만, 더 복잡한 스킴일수록 검증이 어려워 취약점 잠재 가능성도 높습니다).

사례: FIDO2/WebAuthn 자체는 개별 서비스간 식별자 공유 차단으로 privacy-friendly라 했지만, 한 연구에 따르면 WebAuthn API 타이밍 등으로 같은 기기인지 fingerprinting 가능하단 지적도 있었습니다 (브라우저에서 몇 ms 차로 FIDO device model 추정 등). 이런 side-channel도 고려해야 합니다.

4) 활용 사례 (산업별 적용 현황)

프라이버시 보호 인증은 여러 영역에서 점진적으로 도입되고 있습니다. **정부 전자ID** 분야에서, 에스토니아 e-ID의 차세대 모델이 **자격증명 최소 공개**를 지향하고 있습니다. EU도 eIDAS 2.0 프레임워크에 **"사용자 컨트롤, 최소 데이터 공유"** 원칙을 넣고 **영지식 기반 연령/자격 증명** 시범을 하고 있습니다. 예컨대 독일은 전자신분증 NFC에 **Idemix 기반 익명인증** 기능을 시험 적용하여, 사용자가 이름/주소 숨기고 나이나 자격만 증명하는 PoC를 한 바 있습니다.

의료 분야에서는 환자 프라이버시 중요해, **영지식 기반 자격 증명** 사용이 논의됩니다. 의사가 자기 면허증명 VC만 제출하고 이름은 숨긴 채 처방 시스템에 로그인하거나, 환자가 민감 질병 기록 접근시 자신이 특정 권한 있음을 증명하되 신원은 노출 안 하는 식입니다. 일부 EU 연구 프로젝트 (e.g. KRAKEN)에

서 의료 SSI 활용을 다루고 있습니다.

사이버보안 제품 측면에서, Cloudflare는 **Turnstile**이라는 프라이버시 친화적 CAPTCHA 서비스를 내놓았는데, 여기서는 **Privacy Pass**를 활용해, 이용자에 대한 식별정보 없이도 인간 확인을 하고 익명 토큰 발급해 웹사이트들이 그걸로 검증하게 합니다. 애플 iOS16도 Private Access Token을 구현하여, 아이폰 사용자들이 CAPTCHA 거의 없이 PAT로 인증 통과하면서도 개인정보는 안 주게 했습니다.

메시징/커뮤니티 서비스에서도, 익명 프로토콜이 쓰입니다. Telegram 등은 **익명 로그인** 기능을 추가했는데, 이는 전화번호 없이 blockchain 기반 익명번호로 계정 생성하는 (Fragment ID) 방식을 도입했습니다. 또한 여러 SNS들이 **OAuth를 통한 "로그인 없이 인증"** 기능 (예: 임시 손님계정) 제공 시 Privacy Pass류 토큰을 고려하기도 합니다.

VPN/VPS 서비스 중에는 **익명 결제 + 익명 인증**을 접목해, 사용자 이메일 등 없이도 계정 운영하는 곳도 있습니다 (종이 신원 전혀 안 남김).

암호화폐/블록체인 dApp에서는 익명 로그인 기본이고, 다만 그 주소 자체는 공개되므로 프라이버시 문제 있습니다. 이를 개선하려 **ZK login** 개념이 나옵니다: 사용자가 Metamask 주소 영지식증명으로 dApp 로그인, dApp은 주소 안 받고 로그인만 허용. EY, Consensys 같은 블록체인 기업들이 **ZK Attestation** 같은 표준 제안했습니다.

학술/NGO에서도 개인정보 최소화 인증 적용 사례가 있습니다. UNESCO는 인터넷 사용자에게 **익명 전자ID** 부여 논의 (디지털 권리 관점)한 적 있고, 토르 브라우저는 Privacy Pass로 클라우드서비스 접근 시 CAPTCHA 줄이는 기능을 도입했습니다.

인터넷 표준화에서 가장 눈에 띄는 변화는, 과거에는 SAML/OAuth 등에서 사용자 식별자 통일이 편리성 논점이었지만 이제는 **"각 RP에 pairwise 난수**

ID"가 권고되고, OIDC에서도 그것 기본값으로 바뀐 점입니다[100]. 이를 채택하는 서비스들이 늘면서, SSO하더라도 개별 서비스끼리 ID 매칭 어려워지고 있습니다.

국내도 2021년 과기정통부 지원으로 **자기주권 신원증명** 시범사업에 Idemix, ZKP 등이 활용되어, 대학 졸업증명서 익명 제출 서비스 등을 구현했습니다. 또한 2020년 경찰과 협력해 **영지식 나이인증 앱**(신분증 없이 얼굴인증만으로 성인 확인, 미성년자는 얼굴인증 불가하게) 을 PoC한 바 있습니다.

실버타운/나이트클럽 등 프라이버시 민감 현장에서도, 영지식 출입인증 (예: 얼굴인증하되 기록 남지 않게) 채택 논의가 있습니다.

다만 현실적으로는 아직 상용화 초기이고, **모바일 운전면허증** 같은 경우도 결국 경찰이 스캔하면 다 보이는 형식이어서, 프라이버시 보호형으론 갈 길이 남았습니다.

5) 기술적/운영상 장단점

프라이버시 보호 인증의 **장점**은 이용자 개인정보와 프라이버시를 지키면서도 필요한 인증 기능을 수행한다는 점입니다[95]. 이는 **규제 준수**와 **사용자 신뢰** 모두에 이득입니다. GDPR, 개인정보보호법 등 규제에서는 **최소한 정보수집, 목적외 사용금지**를 요구하는데, 프라이버시 보호 인증은 이를 기술적으로 보장하므로 법적 위험을 줄입니다. 또한 데이터 유출시 피해 최소화(애초 수집 안 했으므로 누출될 게 없음) 효과도 있습니다. 사용자 입장에서도, 자신의 신원이 노출 안 되고 추적 안 당한다고 인식하면 **서비스 이용 안심감**과 **만족도**가 증가할 수 있습니다[95]. 예를 들어 어떤 온라인상 민감한 활동(의료 상담, 정치 토론 등)도 익명성을 유지하면서 인증된 사람끼리 할 수 있다면 참여가 늘어날 수 있습니다.

보안적으로도, 앞서 언급했듯 **피싱, 크리덴셜 유출 위험** 감소, **데이터 브로커 추적 차단** 등의 부수 이익이 있습니다. 중앙 ID 저장소가 필요 없으므로 **대형 ID 서버 공격면**도 줄어듭니다. IdP가 있어도 사용자 mapping 정보 안 가지니

해커도 못 훔칩니다[101].

또 하나 장점은 **상호운용성 증대**입니다. 아이러니하게 들리지만, 프라이버시 보호 인증 표준 (VC/DID, OIDC SD-JWT 등)이 나오면 전 세계 서비스가 통일된 방식으로 최소정보 교환하므로 **운영 효율**이 좋아집니다. 사용자는 **원-클릭 익명 로그인**이 가능해 편리합니다.

단점은, 기술적 복잡성이 높아 **구현오류 가능성**이 있다는 것, 그리고 **추적 불가능이 범죄악용**될 우려가 있다는 점입니다[102]. 예를 들어 완전 익명 게시판이면 악플러, 범죄모의가 있어도 잡기 어렵습니다. 이런 문제로 현실에선 **조건부 프라이버시** (익명이지만 사법기관 요청 시 해제 가능 등)를 넣기도 하는데, 이는 오남용 위험도 있습니다. 즉 **익명성 vs 추적성**의 균형이 과제입니다.

또한 **추가 리소스 요구**(클라이언트에 지갑앱 설치 등)로 인한 **사용자 진입장벽**이 단점입니다. 일반인이 이해하기 어려워 채택 꺼릴 수 있습니다. 기업 쪽도 **기존 시스템 개조**가 필요해 투자해야 하니 주저할 수 있습니다. 이 때문에 프라이버시 인증이 널리 퍼지지 못하고 부분 도입만 되는 경우가 많습니다.

성능도 이슈입니다. 많은 암호연산이 필요해, 한꺼번에 수만명 인증 같은 상황에서 Latency나 CPU 부담이 커질 수 있습니다. Cloudflare 등은 Privacy Pass를 초당 수천건 처리 테스트 했으나, 그건 간단 블라인드 서명 수준이라 그나마, ZKP 쪽은 아직 속도 개선 여지 많습니다.

호환성 문제: 프라이버시 보호 기능이 모든 시나리오에 적용 못되는 경우도 있습니다. 예를 들어 금융 거래는 감사 위해 신원기록이 필요하므로 완전 익명인증 채택 어려울 수 있고, 이럴 땐 굳이 복잡한 인증을 도입 안하려 할 수 있습니다.

사용자 측 불편: 어느 정도 자기 정보 관리 책임이 사용자에게 옵니다. 익명 증명서를 분실하면 다시 발급받아야 하고, 백업 안 하면 복구 못하는 등. 기존엔 중앙 서버가 다 알아서 해줬지만, 자기주권형은 편의성 측면 희생이 조금

있습니다.

정책 측면: 법/규제 기관이 익명 인증을 싫어할 수도 있습니다 (범죄 악용 우려). 그래서 규제 장벽 만날 수 있습니다. 이를 설득하려고, "영지식이나 DID도 법원영장 시 특정조건으로 익명 해제 가능"같은 메커니즘을 가질 때 수용될지 논의 중이나, 그럼 기술이 다시 복잡/약화됩니다.

운영: 관리자가 익명 인증 사용자를 차단(예: 사이트에서 영구탈퇴) 시 식별 어려워, **운영정책 수립**에 어려움이 있습니다. 토론 사이트에서 한 익명 사용자를 밴하면, 그가 다른 익명 증명서로 다시 오는 것을 막기 어렵습니다. 이런 운영 과제를 어떻게 해결(예: 평판 토큰, 사용약관 위반시 증명서 철회 등)할지도 고민거리입니다.

아. 결론

현대 인증 기술은 **보안성과 편의성, 그리고 프라이버시**를 모두 향상시키기 위해 다각적인 혁신을 이루고 있습니다. **생체인증**과 **패스워드리스 인증**은 사용자 경험을 개선하면서도 피싱 저항성 등 보안을 강화하였고, **블록체인 기반 분산 인증**과 **프라이버시 보호 인증**은 신원 확인 과정에서 중앙 통제와 개인정보 수집을 최소화하여 **사용자 주권과 프라이버시**를 높이는 방향으로 나아가고 있습니다[39][95]. 한편 **위험 기반 인증**과 **연속 인증**은 정적인 1회성 인증을 뛰어넘어, **맥락 인텔리전스와 지속 모니터링**으로 동적이고 적응적인 보안을 제공하고 있습니다[67][72]. 이러한 기술들은 상호 보완적으로 활용되어, 궁극적으로 **"투명하면서도 강건한"** 인증 체계를 지향합니다.

글로벌 표준과 정책도 이러한 흐름을 적극 수용하고 있습니다. FIDO 얼라이언스의 패스키를 비롯해, W3C의 DID/VC 표준, IETF의 Privacy Pass, NIST의 Zero Trust 지침 등에 이르기까지, **국제 표준들은 보안성과 상호운용성을 높이는 동시에 개인정보 최소화 원칙을 강조**하고 있습니다[44][72]. EU의 eIDAS 2.0 규정이나 미국 연방의 사이버 전략은 기술 표준과 연계되어 업계에 **인증 혁신의 방향을 제시**하고 있습니다. 결과적으로 향후 수년 내에 지금까지 병행

되어 온 전통 비밀번호 및 중앙집중 인증 방식에서 벗어나, **패스워드 없는 공개키 기반 인증, 사용자 개인 디바이스 기반 MFA**, 그리고 **분산 신원증명 기반 속성 인증**이 주류로 부상할 전망입니다[27][50]. 또한 양자 컴퓨팅 시대를 대비하여 **양자 안전 인증** 알고리즘들도 표준화되어 (예: NIST PQC 표준) 기존 인증 프로토콜에 통합될 것이 요구되고 있습니다[103].

물론 이러한 전환에는 여러 도전과제가 있습니다. **사용자 인식 전환과 구식 시스템 호환** 문제가 있고, **성능 및 실용성** 개선이 더 필요하며, **프라이버시와 수사 필요성 간 조화** 등 정책적 합의도 필요합니다. 하지만 보안 공격의 고도화와 사용자 요구 수준의 상승에 힘입어, 인증 기술의 진화는 피할 수 없는 방향입니다. **"신뢰할 수 있는 인증 없이는 디지털 사회의 안전도 없다"**는 인식 아래, 산업계와 표준기구, 정부가 협력하여 보다 강인하고 사용자 친화적인 인증 체계를 구축하는 노력이 지속될 것입니다. 궁극적으로 미래 인증은 **다중요소(Multi-factor)**, **맥락기반(Contextual)**, **지속검증(Continuous)**, 그리고 **프라이버시보호(Privacy-preserving)**의 키워드로 요약될 수 있으며, 오늘 살펴본 각 기술들은 그러한 미래상의 일부분을 구현하고 있습니다. 이들 기술을 균형 있게 적용함으로써, 우리는 **사용자는 불편 없이 안전하고, 공격자는 숨어들 틈 없는** 인증 환경을 만들어나갈 수 있을 것입니다[104].

참고문헌

- FIDO Alliance (2022), *"The 2023 Online Authentication Barometer"*, FIDO Alliance 보고서[31][27]
- NIST SP 800-63B (2020), *"Digital Identity Guidelines: Authentication and Lifecycle Management"*, NIST (미국 표준기술연구소)[105][9]
- Gilad David Maayan (2024), *"10 Authentication Trends in 2024 and Beyond"*, Tripwire 블로그[64][96]
- Cloudflare (2024), *"NIST's first post-quantum standards"*, Cloudflare Blog (Luke Valenta et al.)[103][106]
- Okta (2024), *"Risk-Based Authentication: What You Need to Consider"*, Okta Identity 101 가이드[67][65]
- Camenisch et al. (2010), *"Specification of the Identity Mixer Cryptographic Library"*, IBM Research Report (Idemix)[44][97]

- Microsoft (2013), *"U-Prove Technology Overview"*, MSDN Magazine [44]
- Zhu et al. (2023), *"Continuous Authentication in the Digital Age: An Analysis of Reinforcement Learning-based Continuous Authentication"*, MDPI Sensors Journal[91][82]
- FFIEC (2021), *"Authentication and Access to Financial Institution Services and Systems"*, FFIEC Guidance[35]
- W3C (2021), *"Verifiable Credentials Data Model v1.1"*, W3C Recommendation (제시된 VC/DID 표준)[62][60]

[1] [23] [27] [30] [33] [37] Why Aren't We Using Passkeys? Obstacles Companies Face Deploying FIDO2 Passwordless Authentication

https://www.usenix.org/system/files/usenixsecurity24-lassak.pdf

[2] [4] [5] [7] [8] [9] [10] [11] [12] [13] [14] [15] [16] [19] [20] [21] [34] [79] [105] NIST Special Publication 800-63B

https://pages.nist.gov/800-63-3/sp800-63b.html

[3] What is Biometric Authentication? | Definition from TechTarget

https://www.techtarget.com/searchsecurity/definition/biometric-authentication

[6] [24] [25] [26] [28] [29] What is the Difference Between FIDO2 and Passkeys?

https://www.corbado.com/faq/what-is-difference-fido2-passkeys

[17] [18] What is Biometric Authentication? Use Cases, Pros & Cons | OneSpan

https://www.onespan.com/topics/biometric-authentication

[22] [44] [50] [51] [52] [57] [63] [64] [69] [70] [71] [72] [74] [75] [76] [85] [89] [95] [96] [97] [99] [101] [102] [104] 10 Authentication Trends in 2024 and Beyond | Tripwire

https://www.tripwire.com/state-of-security/authentication-trends

[31] 2023 FIDO Report Findings: People Prefer Passwordless - Descope

https://www.descope.com/blog/post/2023-fido-report-findings

[32] The 2023 Workforce Authentication Report - FIDO Alliance

https://fidoalliance.org/the-2023-workforce-authentication-report-embracing-the-passwordless-future/

[35] [36] [78] [80] The FFIEC Recommends Passwordless MFA | HYPR

https://blog.hypr.com/ffiec-endorses-passwordless-mfa

[38] [40] [43] [45] [46] [47] [48] [49] [54] [55] [59] [60] [61] [62] [98] [100] Decentralized Identity: The Ultimate Guide 2025

https://www.dock.io/post/decentralized-identity

[39] [41] [42] [56] A Taxonomic Approach to Understanding Emerging Blockchain Identity Management Systems

https://nvlpubs.nist.gov/nistpubs/CSWP/NIST.CSWP.01142020.pdf

[53] [58] New NIST report highlights limits of 'blockchain' for identity, access management | InsideCyberSecurity.com

https://insidecybersecurity.com/daily-briefs/new-nist-report-highlights-limits-%E2%80%98blockchain%E2%80%99-identity-access-management

[65] [66] [67] [68] Risk-Based Authentication: What You Need to Consider | Okta

https://www.okta.com/identity-101/risk-based-authentication/

[73] What Is Adaptive Authentication & When to Use It - Descope

https://www.descope.com/learn/post/adaptive-authentication

[77] Risk-Based Authorization: A Comprehensive Guide for IAM ...

https://mojoauth.com/ciam-101/risk-based-authorization-iam-passwordless-threat-breach

[81] Privacy-preserving continuous authentication using behavioral ...

https://link.springer.com/article/10.1007/s10207-023-00721-y

[82] [92] Evaluation of Deep Learning Models for Continuous Authentication ...

https://www.sciencedirect.com/science/article/pii/S1877050923012735

[83] [84] [87] [88] [93] [94] What Is Behavioral Biometrics: How Does It Work Against Fraud

https://www.feedzai.com/blog/behavioral-biometrics-next-generation-fraud-prevention/

[86] [PDF] Behavioral Biometrics for Continuous Authentication

https://onlinescientificresearch.com/articles/behavioral-biometrics-for-continuous-authentication.pdf

[90] Behavioral Biometrics for Continuous Authentication - ResearchGate

https://www.researchgate.net/publication/382855823_Behavioral_Biometrics_for_Continuous_Authentication

[91] Continuous Authentication in the Digital Age: An Analysis of ... - MDPI

https://www.mdpi.com/2073-431X/13/4/103

[103] [106] NIST's first post-quantum standards

https://blog.cloudflare.com/nists-first-post-quantum-standards/