

# The academic frontier of IAM technology

Pieter Philippaerts

SUPPORTED BY



# An overview of IAM research

A short taxonomy of the topics researchers are exploring.



## Examples from research

Selected publications make each research direction concrete.

## Future discussions

A broad introduction today, with deeper discussion of individual subjects in future sessions.

# IAM protects the ability to do business

Every employee, application, and automated job needs access that matches its purpose.

## Security

Restrict sensitive actions to authorized people and workloads.

## Accountability

Connect an action to an identity, owner, and business purpose.

## Compliance

Retain evidence of access approvals, changes, and use.



# Three connected jobs in IAM

| IAM responsibility | Core question                                              | Typical mechanisms                                 |
|--------------------|------------------------------------------------------------|----------------------------------------------------|
| Manage identity    | Who exists, who owns the identity, and when should it end? | Join, move, leave; service creation and retirement |
| Authenticate       | How does this actor prove its identity?                    | Password, passkey, federation, workload credential |
| Authorize          | Which resource and action may this actor use?              | Roles, attributes, context, enforcement, and audit |

## Business Processes



Employee Onboarding



Customer Service



Sales & Order Management



Supply Chain & Procurement



Finance & Reporting



IT Operations & Support

## IT Processes



**User Lifecycle Management**  
(provisioning, changes, deprovisioning)



**Access Management**  
(roles, entitlements, approvals)



**Security & Compliance**  
(policy enforcement, auditing)



**Application Management**  
(deploy, configure, maintain access)



**Monitoring & Incident Management**  
(detect, respond, remediate)



**Identity Governance**  
(access reviews, certifications)

## Systems and Applications



**HR System**  
(Workday)



**CRM**  
(Salesforce)



**ERP**  
(SAP)



**Supply Chain**  
(Oracle)



**Finance**  
(Oracle, NetSuite)



**Collaboration**  
(Microsoft 365)



**Infrastructure**  
(Azure, AWS)



**Custom Applications**  
(internal)

## Identity and Access Management (IAM)



**Identities**  
(users, service accounts, devices)



**Authentication**  
(MFA, SSO)



**Authorization**  
(roles, policies, entitlements)



**Governance**  
(access reviews, certifications)



**Identity Services**  
(APIs, directories, federation)

# Seven questions at the academic frontier

The research moves from proving identity to controlling authority across its entire lifecycle.

- |           |                                   |                                               |
|-----------|-----------------------------------|-----------------------------------------------|
| <b>01</b> | <b>Strong authentication</b>      | How do we know who you are?                   |
| <b>02</b> | <b>Federation and OAuth</b>       | How does identity or authority cross systems? |
| <b>03</b> | <b>Decentralized identity</b>     | Can the holder carry a verifiable credential? |
| <b>04</b> | <b>Privacy</b>                    | What must a credential reveal?                |
| <b>05</b> | <b>Adaptive authorization</b>     | Should this request be allowed now?           |
| <b>06</b> | <b>Machine and agent identity</b> | Which software acts, and for whom?            |
| <b>07</b> | <b>Governance</b>                 | Can we explain and revoke effective access?   |

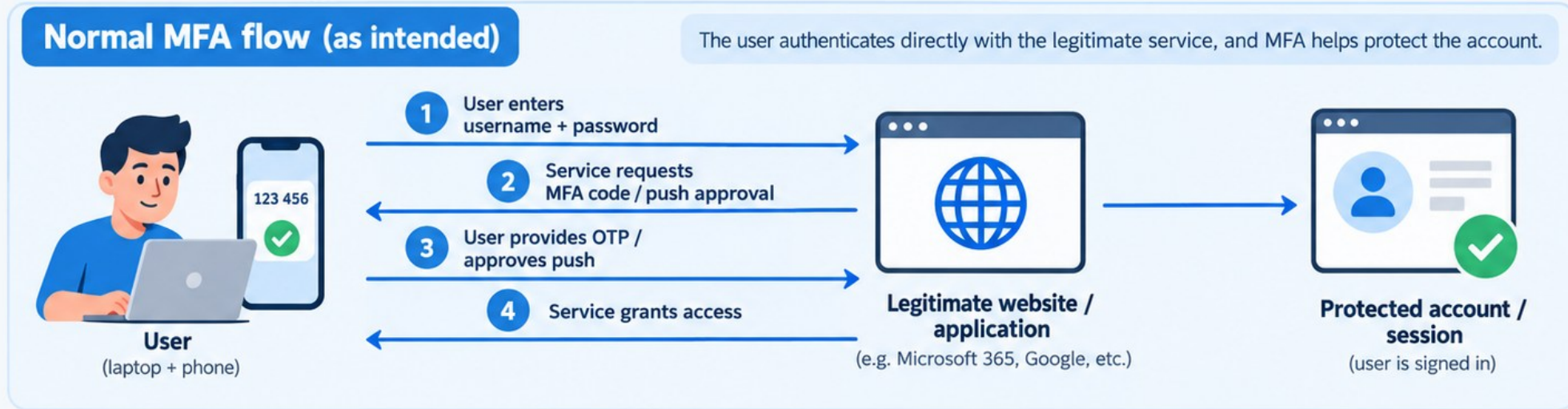
RESEARCH FRONTIER

# Strong authentication

How do we know who you are?



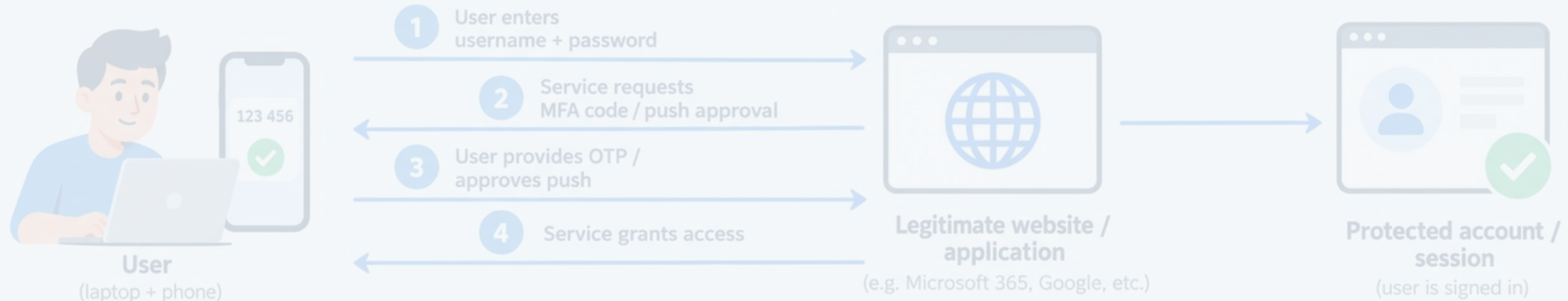
# Do we not have strong authentication already?



# Do we not have strong authentication already?

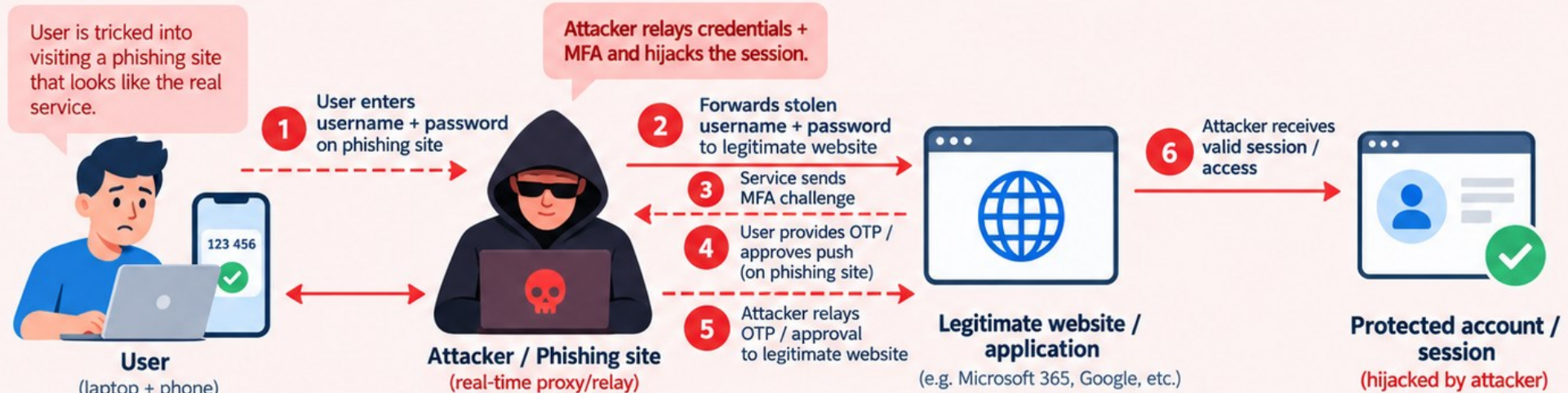
## Normal MFA flow (as intended)

The user authenticates directly with the legitimate service, and MFA helps protect the account.



## Phishing / real-time relay attack

The attacker sits in the middle, relaying the login and MFA in real time.



# Phishing resistance changes the authentication model

FIDO2 and WebAuthn bind a public-key credential to the intended service.

## Password plus a code

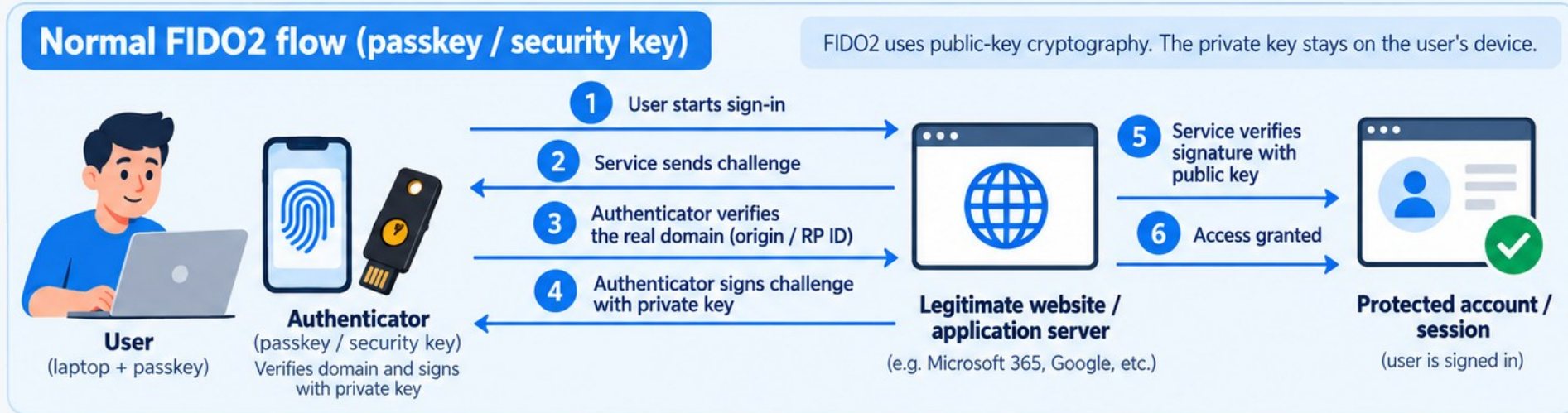
A fake site may capture the password and relay the one-time code.

## Passkey or security key

The authenticator signs a challenge for the relying party.  
A PIN or biometric can unlock the key locally.  
The credential is bound to the legitimate domain.



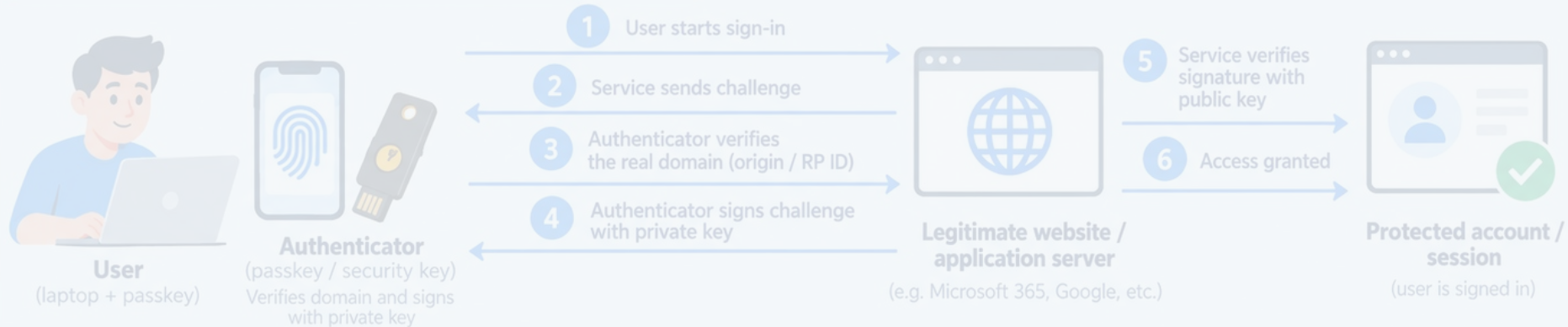
# Phishing resistance with FIDO2



# Phishing resistance with FIDO2

## Normal FIDO2 flow (passkey / security key)

FIDO2 uses public-key cryptography. The private key stays on the user's device.



## Phishing attempt is blocked

Attacker tries a real-time relay, but the authenticator will not sign for the phishing site.



# Security keys can remove friction from sign-in

A small-scale study: five-week field study with 34 employees at one IT company.

## Mean sign-in time (seconds)



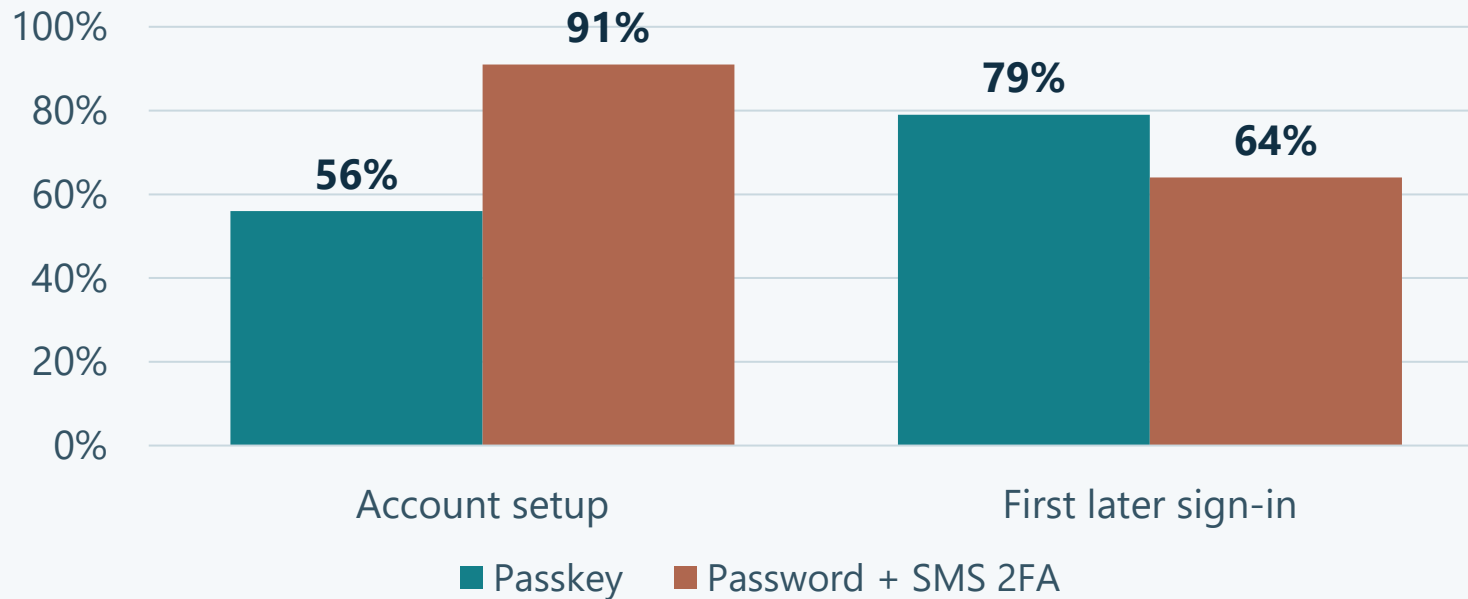
The overall time difference was not statistically significant. However, Employees reported fewer cognitive steps and higher satisfaction.

A rollout needs a coherent login journey as well as a stronger authenticator.

# Passkeys work better after the setup hurdle

A large-scale study: users of a healthcare service chose their sign-in method.

## Successful attempts at two distinct stages



### Setup is the weak point

Passkey registration succeeded for 1,664 of 2,950 users.

### Later use improves

The first subsequent attempt succeeded more often with a passkey.

Measure registration and ongoing use separately. Cross-platform use needs its own testing.

# The login screen is the visible tip of deployment

## Above the surface

A simpler passwordless sign-in.

## Deployment underneath

Recovery and break-glass access  
Legacy applications and SSO  
Device, browser & OS compatibility  
Enrollment & user onboarding  
Credential lifecycle management  
Shared accounts & shared devices  
Regulation & internal security policies

...



# The login screen is the visible tip of deployment

## Above the surface

A simpler passwordless sign-in

## Deployment underneath

Recovery and break-glass access

Legacy applications and SSO

Device, browser & OS compatibility

Enrollment & user onboarding

Credential lifecycle management

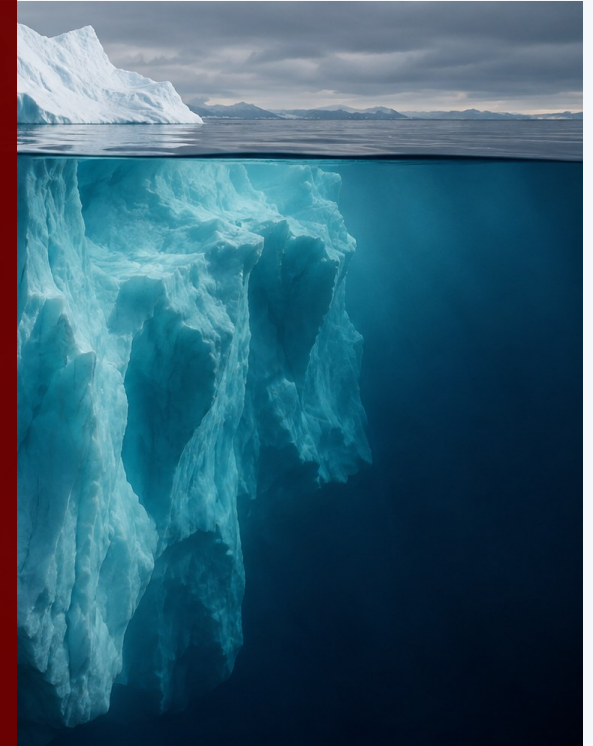
Shared accounts & shared devices

Regulation & internal security policies

...

## Top 5 Obstacles

1. No standardized fallback
2. Complexity and friction
3. Technical issues
4. Regulatory requirements
5. Security culture



# Account recovery must remove attacker persistence

31 participants attempted recovery across Google, PayPal, and LinkedIn.



## Starting point

An attacker already has a session and has added their own passkey.

## Why password reset falls short

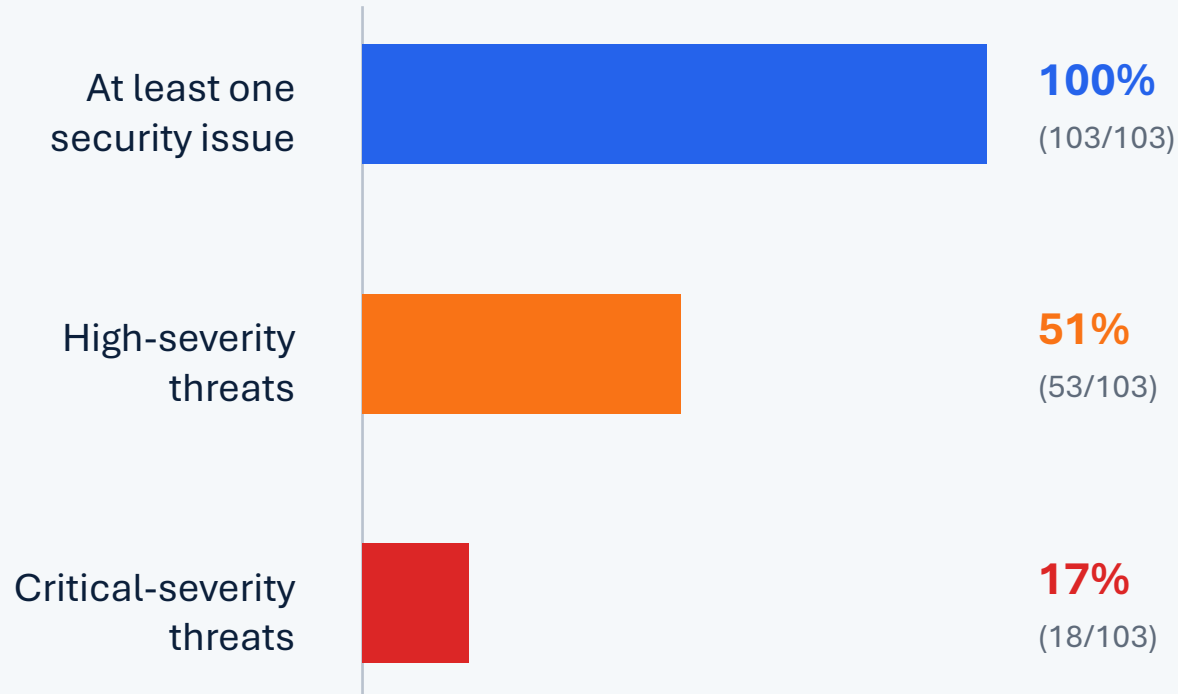
The attacker's credential or active session may survive.

## Recovery scope

Review passkeys, MFA and recovery factors, devices, sessions, connected apps, and grants.

# Passkey implementations are not automatically secure

## Security issues in passkey implementations



## Examples of vulnerabilities

- **No signature check (5%)**  
Website accepts a passkey assertion without verifying the cryptographic signature.
- **No origin validation (40%)**  
Passkeys could be accepted from unauthorized domains or subdomains.
- **Account enumeration (66%)**  
Attackers can infer whether an account exists or uses passkeys.
- **Weak challenge validation (21%)**  
Replay or session-manipulation protections may fail.

RESEARCH FRONTIER

# Federation, OAuth, and SSO

How does authority move between systems?



# Federation passes identity; OAuth delegates access

A business application may receive an identity assertion and separately obtain authority to call an API. Each handoff must preserve the user, client, audience, tenant, and session context.



**OpenID Connect identifies the user**

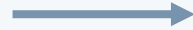
**OAuth limits permitted API operations**

# Valid credentials can still be bound to the wrong context

## Credential validity

The credential itself can be valid.

- ✓ Signature is valid
- ✓ Issued by a trusted identity provider
- ✓ Token is not expired
- ✓ Format and structure are correct



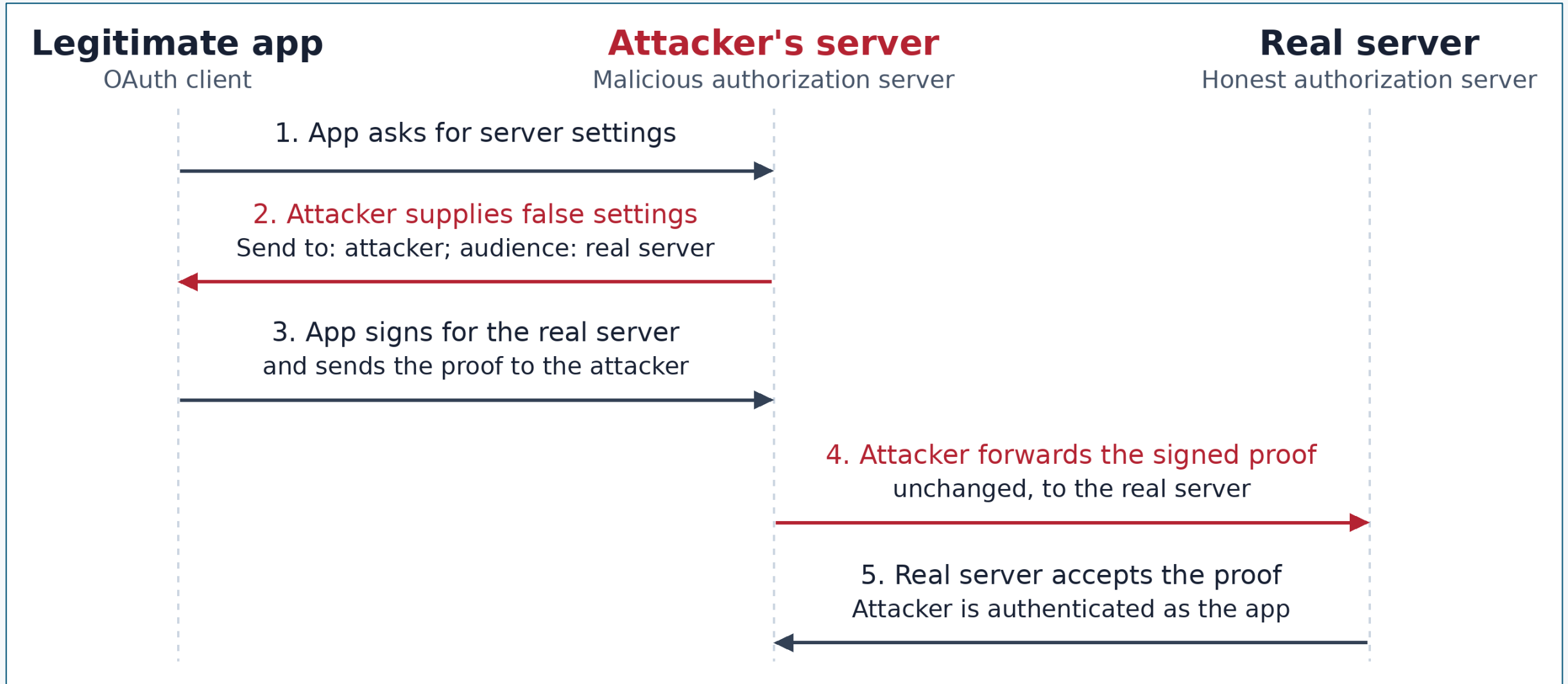
A valid credential is presented in a specific context

## Context binding

The credential must also be correct for the context in which it is used.

- 👤 Which user does it represent?
- 💻 Which client is allowed to use it?
- 🎯 Which audience or resource is it intended for?
- 🏢 Which tenant or environment does it belong to?
- 🔗 Which session is it part of?

# Audience injection redirects a valid client assertion



# Audience injection redirects a valid client assertion

## What is the attack?

1

Client assertions are signed JWTs used for client authentication.

2

A malicious AS can trick a client into signing an assertion whose audience points to an honest AS.

3

That assertion can be replayed at the honest AS to impersonate the client and obtain tokens or access.

## Why it matters



### High-security impact

Affects security-focused deployments, including PAR + private\_key\_jwt and FAPI-style ecosystems.



### Standards-level problem

The issue is in the standards design, not just a single implementation.



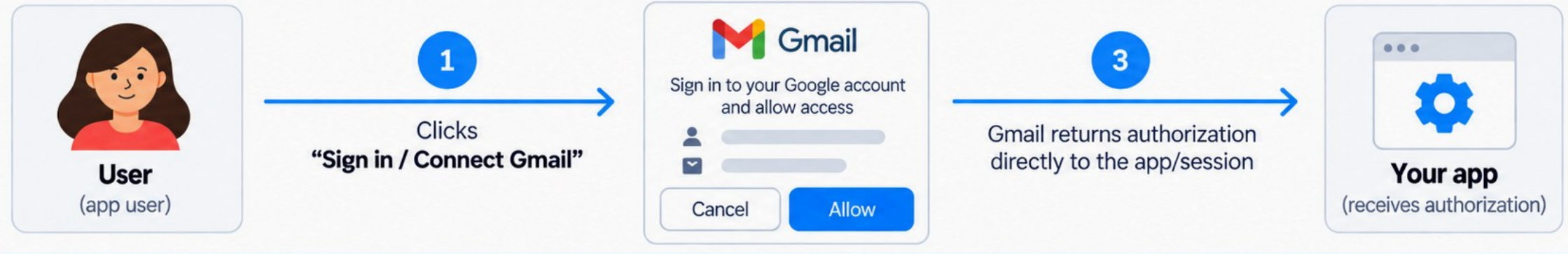
### Broad impact

Relevant to banks/open finance, healthcare, SaaS identity providers, and other SSO ecosystems.

# Brokered OAuth breaks the traditional security model

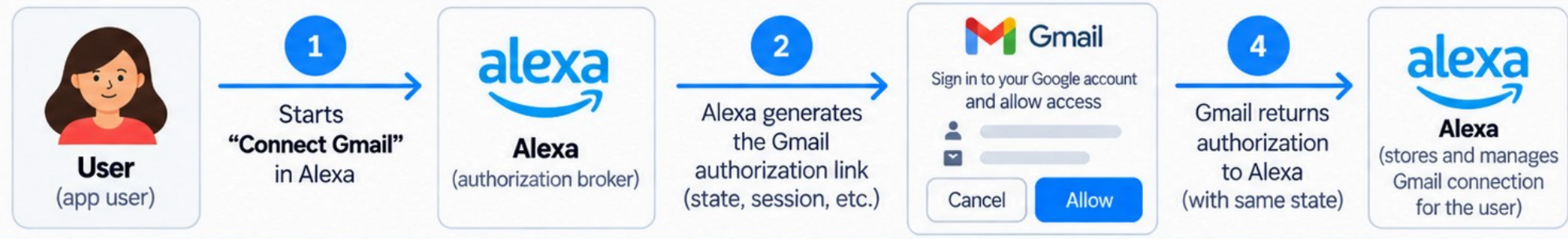
## Traditional OAuth (direct)

User interacts directly with Gmail for authorization

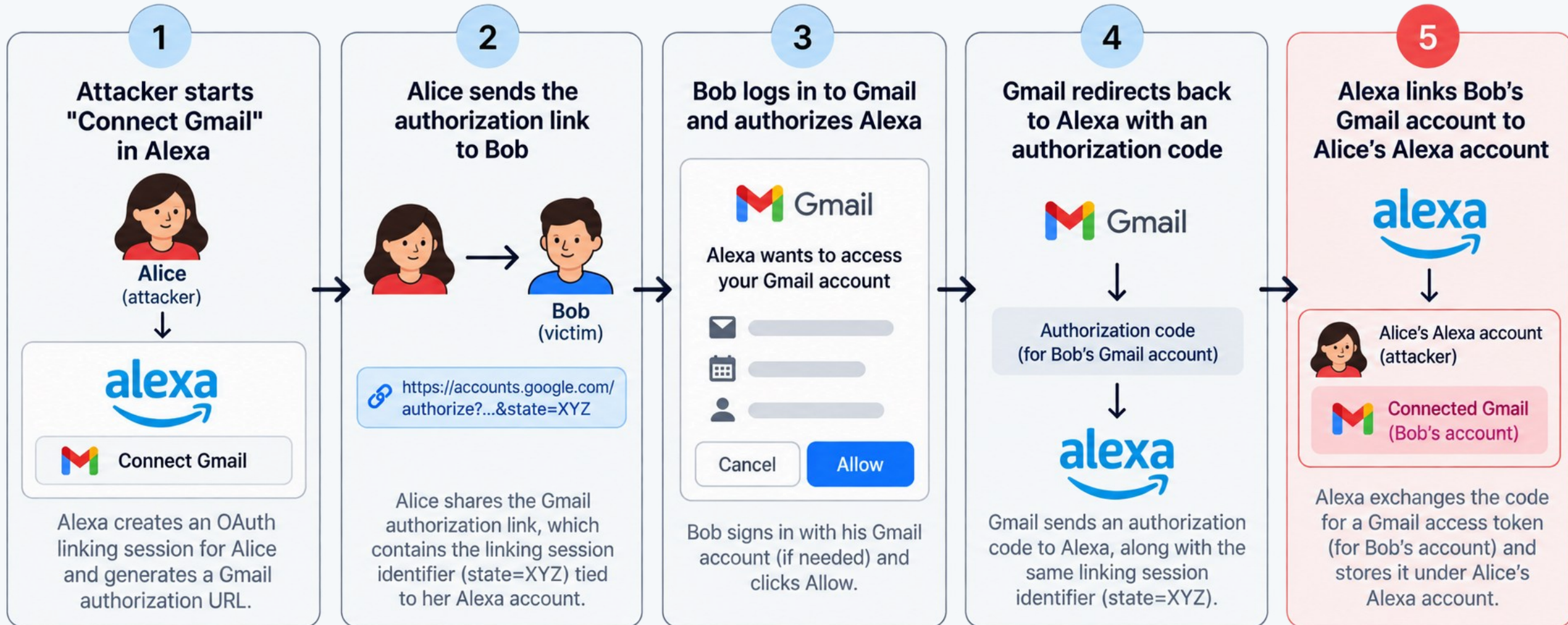


## Brokered OAuth (via Alexa)

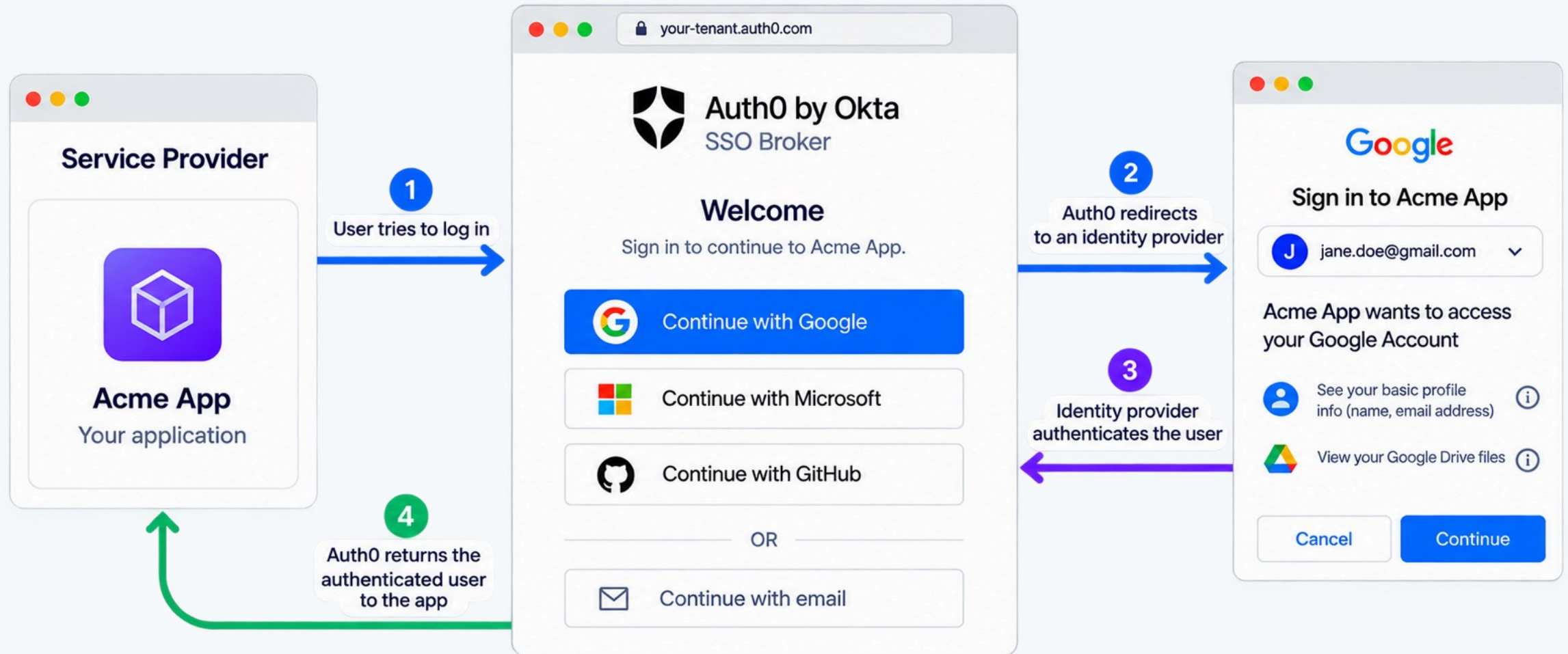
User authorizes Gmail through Alexa, which manages the connection



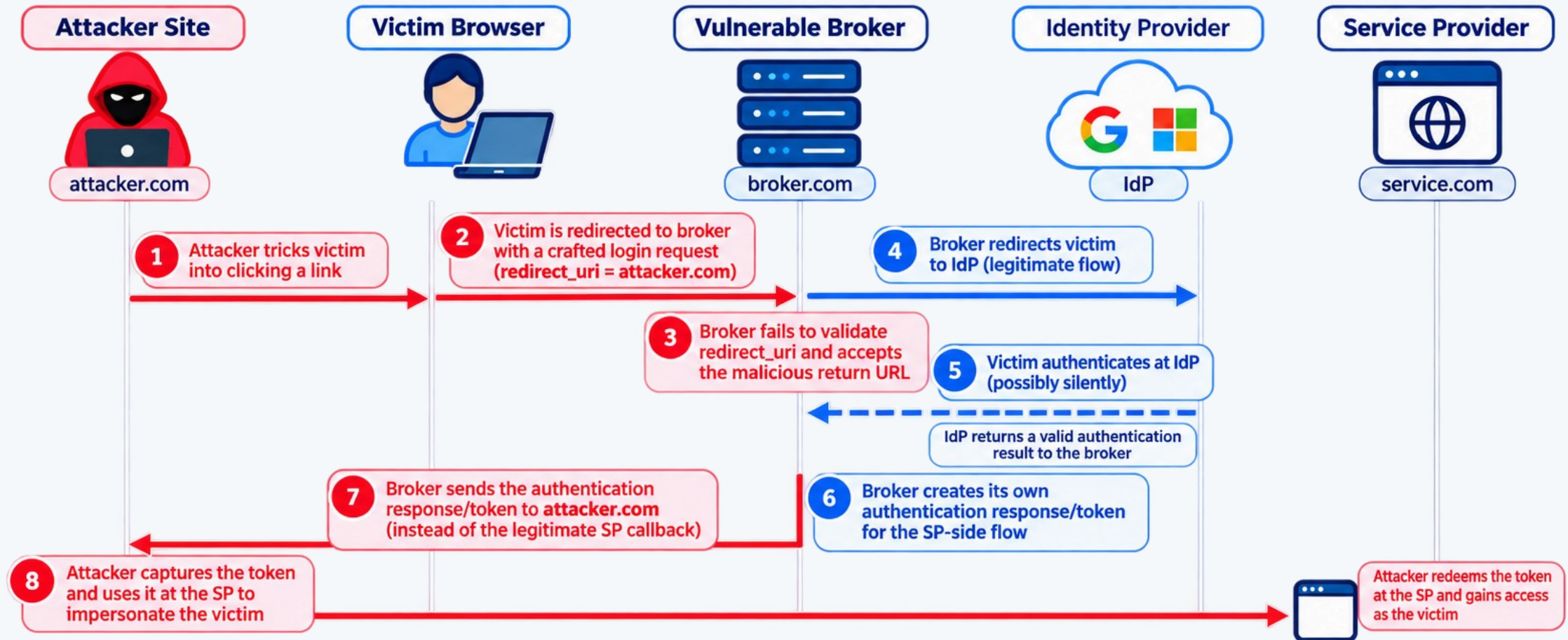
# Cross-user OAuth Session Fixation (COSF) attacks



# SSO brokers can weaken a protected authentication flow



# Open redirection attacks in brokered SSO



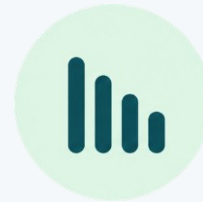
20% of brokers were vulnerable to this attack, leading to account takeovers on 1.950 service providers

# Security standards are catching up



## Use modern, secure patterns by default

Older approaches have been phased out in favor of safer standard practices.



## Reduce the value of stolen tokens

Limit access, scope, and lifetime so a leak has less impact.



## Protect the authorization journey end-to-end

Prevent redirection, impersonation, and interception during the process.



## Prefer secure defaults over custom choices

Build security into the platform so teams do not need to reinvent it.

RFC 9700 moves OAuth from “a flexible framework” toward a safer, opinionated security baseline. OAuth 2.1 will simplify the framework around the security practices we now consider standard.

RESEARCH FRONTIER

# Machine and AI-agent identity

What if the actor is software?



# Machine identities have a different operating rhythm

Machine Identity Management in Modern Enterprise Security reviews the expanding non-human identity estate.

## Service accounts and API clients

Often persistent, with broad dependencies

## Containers and CI/CD jobs

May exist for minutes and authenticate repeatedly

## SaaS integrations and devices

Cross administrative and vendor boundaries

## AI agents

Combine workload identity with delegated human goals



# A temporary workload can leave a permanent credential

The workload-identity literature highlights a mismatch between ephemeral jobs and reusable secrets.

## Illustrative CI/CD job

A runner exists for ten minutes but receives an API key valid for months.

## Residual exposure

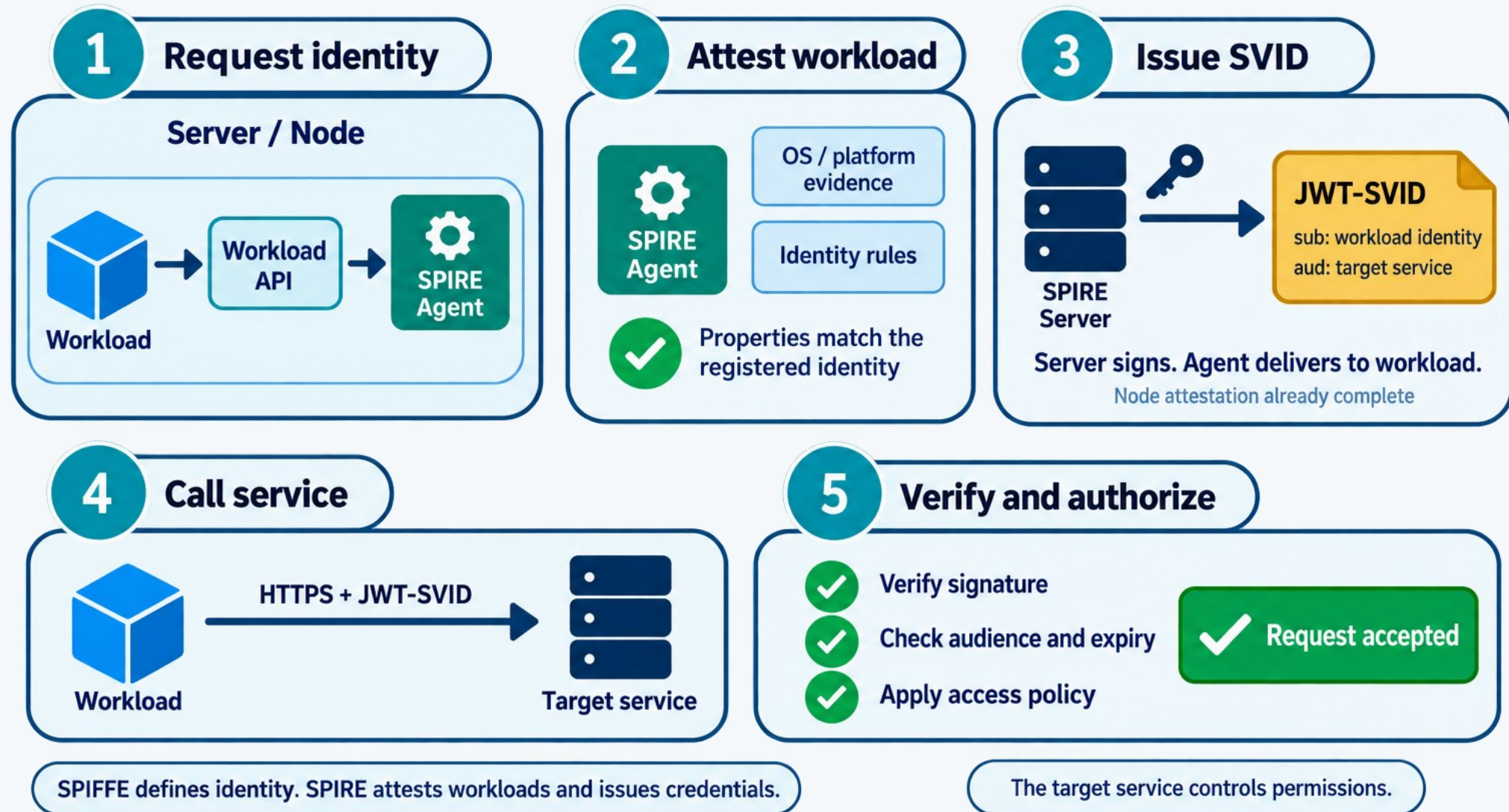
The key can remain in logs, caches, configuration, or compromised dependencies after the runner disappears.

## Lifecycle requirement

Issue, deliver, rotate, retire, and revoke credentials as part of one process.



# SPIFFE connects workload identity to the running process



# Credential lifetime and privilege are separate controls

Replacing a static administrator credential with a five-minute administrator credential still leaves broad authority.

|                        | Broad administrator rights        | Task-specific permissions               |
|------------------------|-----------------------------------|-----------------------------------------|
| Long-lived credential  | Broad authority and long exposure | Narrow authority, but persistent secret |
| Short-lived credential | Less time to misuse broad rights  | Narrow authority and shorter exposure   |

Changing the expiration time does not change the allowed actions.  
Use short lifetimes together with narrow permissions and resource-specific enforcement.

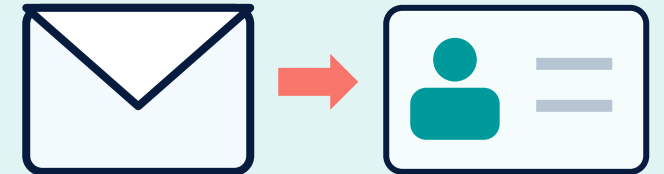
# An attacker can misuse an agent's legitimate access

- Malicious instructions arrive through emails, documents or web pages the agent reads.
- The agent may follow those instructions and use its tools to act on the attacker's behalf.

**Authenticated access does not mean an action reflects the user's intent!**

## Example

You ask an agent to summarize emails. An email tells it to forward private information elsewhere.

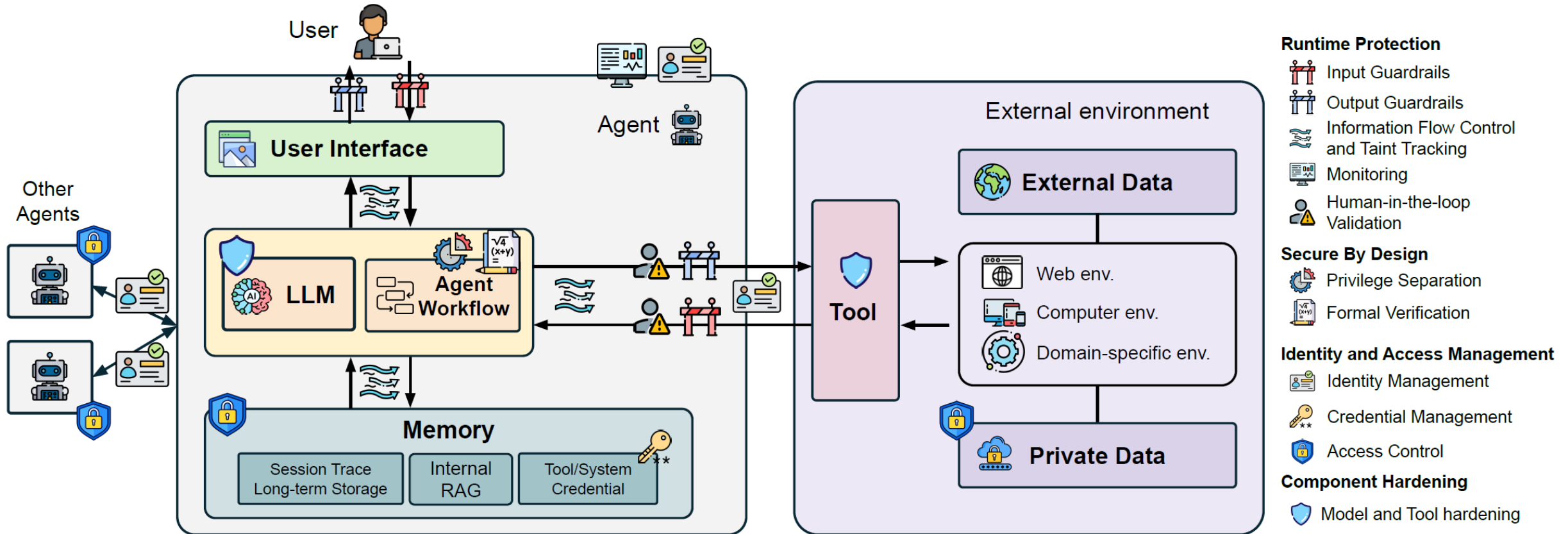


# Do not sell filters as authorization controls

## Defense performance is uneven

|                         |             |                                      |
|-------------------------|-------------|--------------------------------------|
| Input preprocessing     | 60–80%      | detection reported in survey         |
| Architectural defenses  | up to 95%   | protection against known patterns    |
| Known → novel variants  | >90% → <40% | detection can collapse               |
| Prompt hardening bypass | 40–60%      | success for context-ignoring attacks |
| Dual-LLM overhead       | 200–500 ms  | latency per query                    |

# Agent defenses span the entire execution path



RESEARCH FRONTIER

# Governance and evidence

How do we control all of this at scale?



# Appropriate access is the goal



## Excessive access

Unneeded permissions increase security risk.

22.8%\*

Mean estimate of excessive permissions



## Missing access

Insufficient permissions interrupt work.

\* = expert estimate from 10 IAM experts

Baumer et al. (2024). Digital Nudges for Access Reviews: Guiding Deciders to Revoke Excessive Authorizations.

# Data and policy quality affect security

## Stale identity attributes

Access decisions inherit outdated information.

## Unclear policies and owners

Reviewers struggle to judge who needs access.

## Redundant permissions

Removing one role can leave access through another.



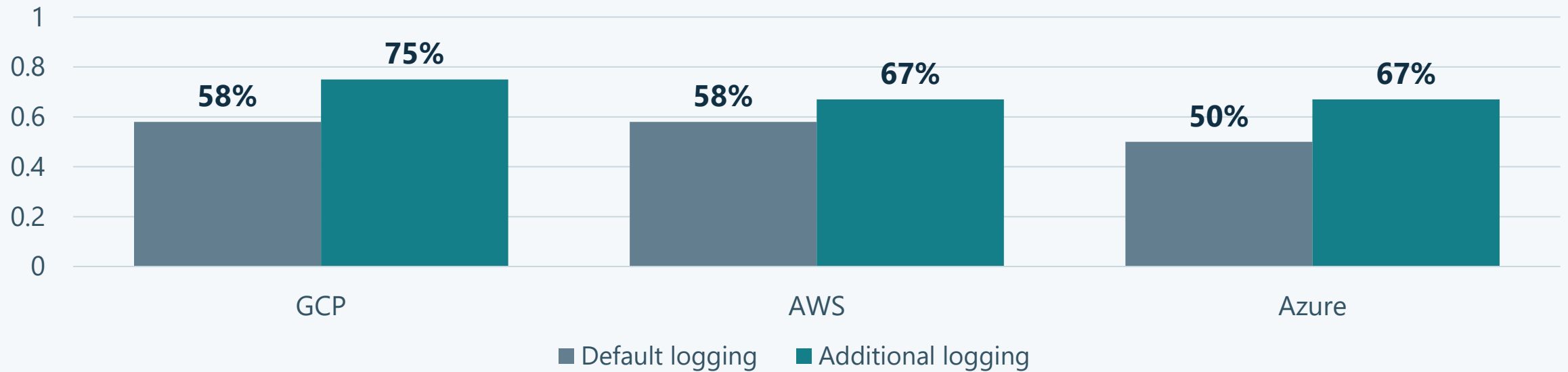
# 43 IAM metrics across 7 perspectives

| Perspective (metric count)    | Metrics                                                                                                                                                                                                                                                                     |
|-------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Identity lifecycle (8)        | Time until access is ready; IAM training; Accounts without an owner; Users with privileged access; Password strength; Biometric authentication strength; Authentication token strength; Multi-factor authentication strength                                                |
| Policy lifecycle (3)          | Bypassed policies or approvals; Time to implement policy changes; Segregation-of-duties violations                                                                                                                                                                          |
| Management and governance (5) | IAM control coverage; Open control findings; Access-decision traceability; Numbers of users, accounts, roles and applications; Inactive accounts and unused roles                                                                                                           |
| Identity quality (5)          | Incorrect identity attributes; Attribute-update delays; Missing or inaccessible identity data; Clarity of identity attributes; Identity-data privacy                                                                                                                        |
| IAM process quality (6)       | Request turnaround time and throughput; Actual versus planned time and cost; Failed process executions; Process recovery time; Process steps with appropriate authorization; Availability of required inputs and resources                                                  |
| IAM policy quality (8)        | Excessive or missing permissions; Access-decision evaluation time; Policy maintenance effort; Policy clarity; Role and rule complexity; Duplicate access paths; Automated policy updates; Policy descriptions and ownership                                                 |
| IAM software quality (8)      | Required feature coverage; Response time and throughput; Integration with other systems; User task success and satisfaction; Service uptime and recovery; Protection against unauthorized access or changes; Software maintenance effort; Installation and migration effort |

# Cloud logging coverage changes what an investigator can see

Researchers executed 12 IAM attack scenarios across Google Cloud Platform, Amazon Web Services, and Microsoft Azure.

## Attack scenarios with observable evidence in logs



Logging configuration is part of IAM security. But even with extended logging, some attacks leave no observable evidence.

# IAM governance as a recurring loop



# The next frontier combines intelligence, privacy, and control

## **IAM remains an evolving research field.**

New applications and forms of interaction continually introduce new questions about identity, trust, and access.

## **Security depends on the complete system.**

Protocols provide guarantees, but implementation choices and operational practices determine whether those guarantees hold.

## **Research challenges existing assumptions.**

It exposes weaknesses and informs improvements to standards, technologies, and deployment practices.

## **Which challenges should we explore together next?**