GARLIC THREAT

STAY AHEAD OF THE THREAT.

what is Breach testing, actually

A Breach test is a 3-week, hands-on attack on your full app and full API, proven with working exploits. **Here is exactly what we test, and how.**

Breach, at a glance.

A 3-week, hands-on test of your full app and your full API, proven with working exploits.

THE NUMBERS

TESTING TIME

3

weeks of hands-on testing

START TO FINISH

~8

weeks, start to closed report

COVERAGE

All

your full app and full API

PRICE

$4,500

fixed, agreed before we start

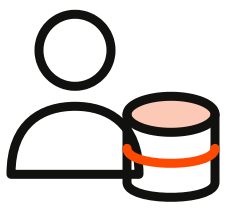
INCLUDED, EVERY TIME

- Audit-ready report
- Plain-English summary
- Attestation letter
- Walkthrough call
- 1 free retest

Who Breach is for.

A product with real users and real data, a funding round, or an enterprise deal that needs a serious report.

BEST FOR



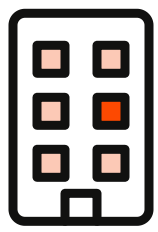
Real users, real data

Tested across your whole app and API.



A funding round

A report investors can trust.



An enterprise deal

A serious report for a serious buyer.

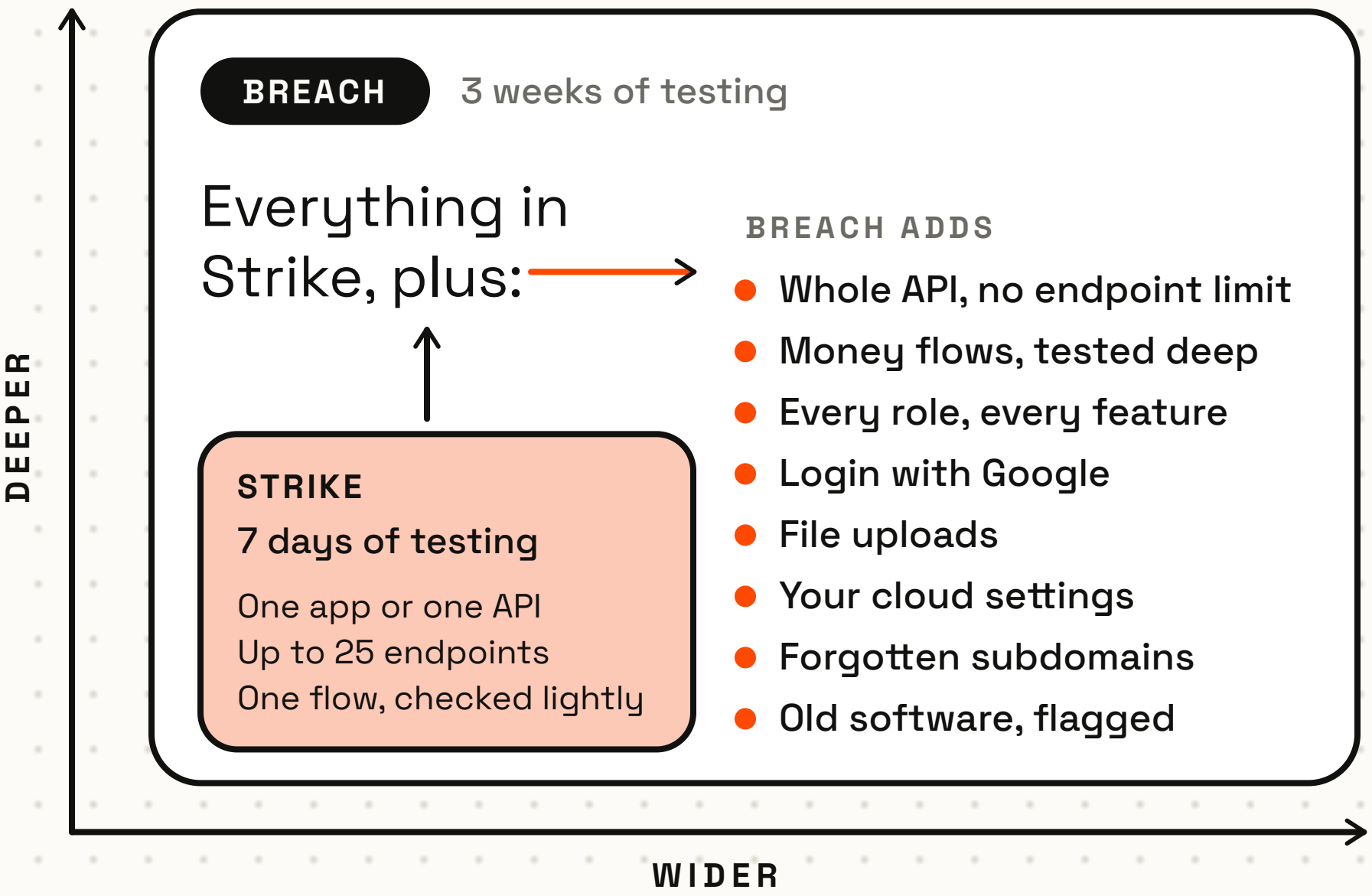
WHAT BREACH IS

Everything in Strike, *deeper and wider.*

Plain words first. If you are a developer, the technical term follows in brackets. IDOR

Deeper. *Wider.*

Breach includes everything in Strike, then goes further: your whole API, every role and your cloud settings.



Four checks. *Every time.*

They catch the flaw behind the big AI-app breaches: access rules left open, so a stranger can read or change your data. Row Level Security / RLS

01

We try to read your data **without logging in**, using your app's own public key.

logged-out query

02

We walk through your protected pages **with no login**, to see what opens anyway.

logged-out route walk

03

We log in as one user and try to **open another user's data.**

two-account ownership swap / IDOR

04

We search the code your app ships for **leaked keys and secrets.**

secret search

Eight more things we test.

On top of everything in Strike, Breach covers your whole app and your whole API.

01

Can someone change data they should not touch, by editing what the app sends?

mass assignment

02

Money flows, tested deep: can someone pay less than they should?

business logic

03

Can a normal user reach admin functions, even hidden ones?

roles and permissions

04

Can someone take over an account through Login with Google?

SSO / OAuth

05

Can someone upload a harmful file, or reach files they should not see?

file uploads

06

Can a stranger read or write your data through your cloud settings?

Supabase / Firebase

07

Are old test or admin addresses still live?

subdomains

08

Is old software with known public holes still running?

outdated libraries

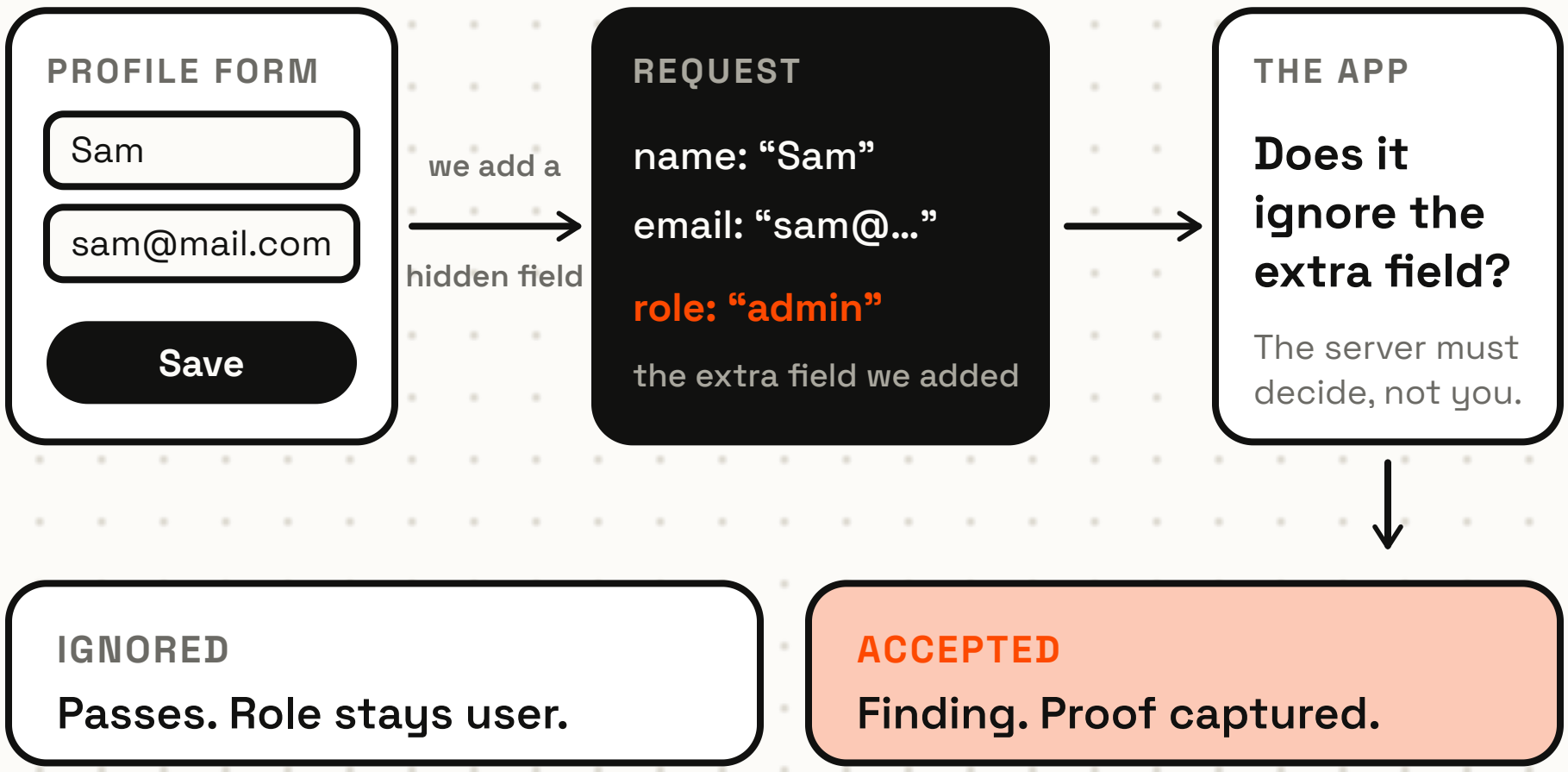
+

Everything in Strike, and your whole API with no endpoint limit.

Editing the request, *shown.*

Can someone change data they should not touch, just by editing what the app sends?

EDITED REQUEST MASS ASSIGNMENT

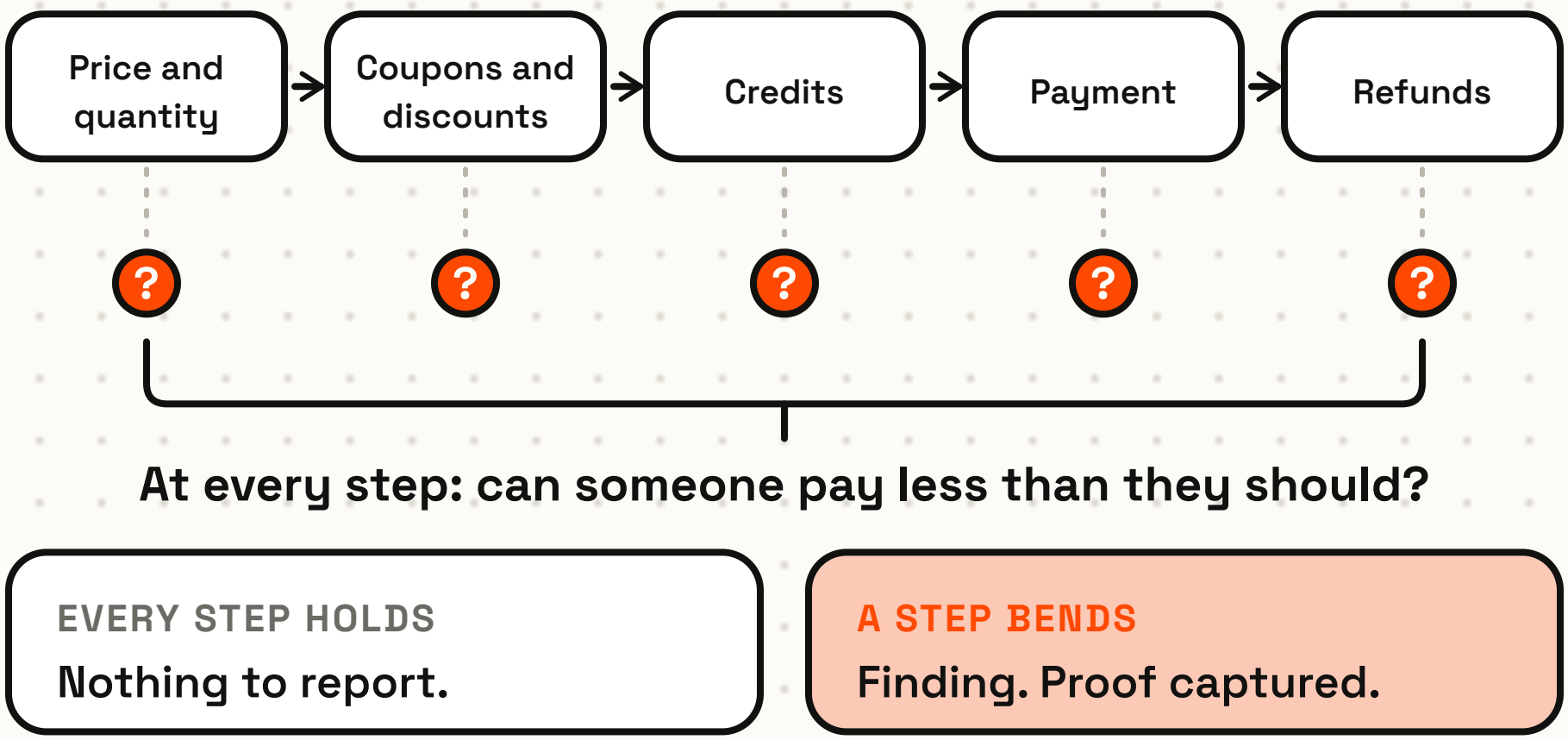


Illustrative values. We also try changing an ID to reach someone else’s record broken object authorisation

Money flows, *tested deep.*

Coupons, discounts, credits, refunds, changing prices or quantities, paying less than you should.

BUSINESS LOGIC



Illustrative flow. Strike checks one money flow lightly. Breach tests them deep.

Every role, *every feature.*

Can a normal user reach admin functions, including hidden ones?

ROLES AND PERMISSIONS

	Own account	Other people's data	Admin pages	Hidden admin functions
Visitor not logged in	✗	✗	✗	✗
Normal user logged in	✓	✗	✗	!
Admin admin login	✓	✓	✓	✓

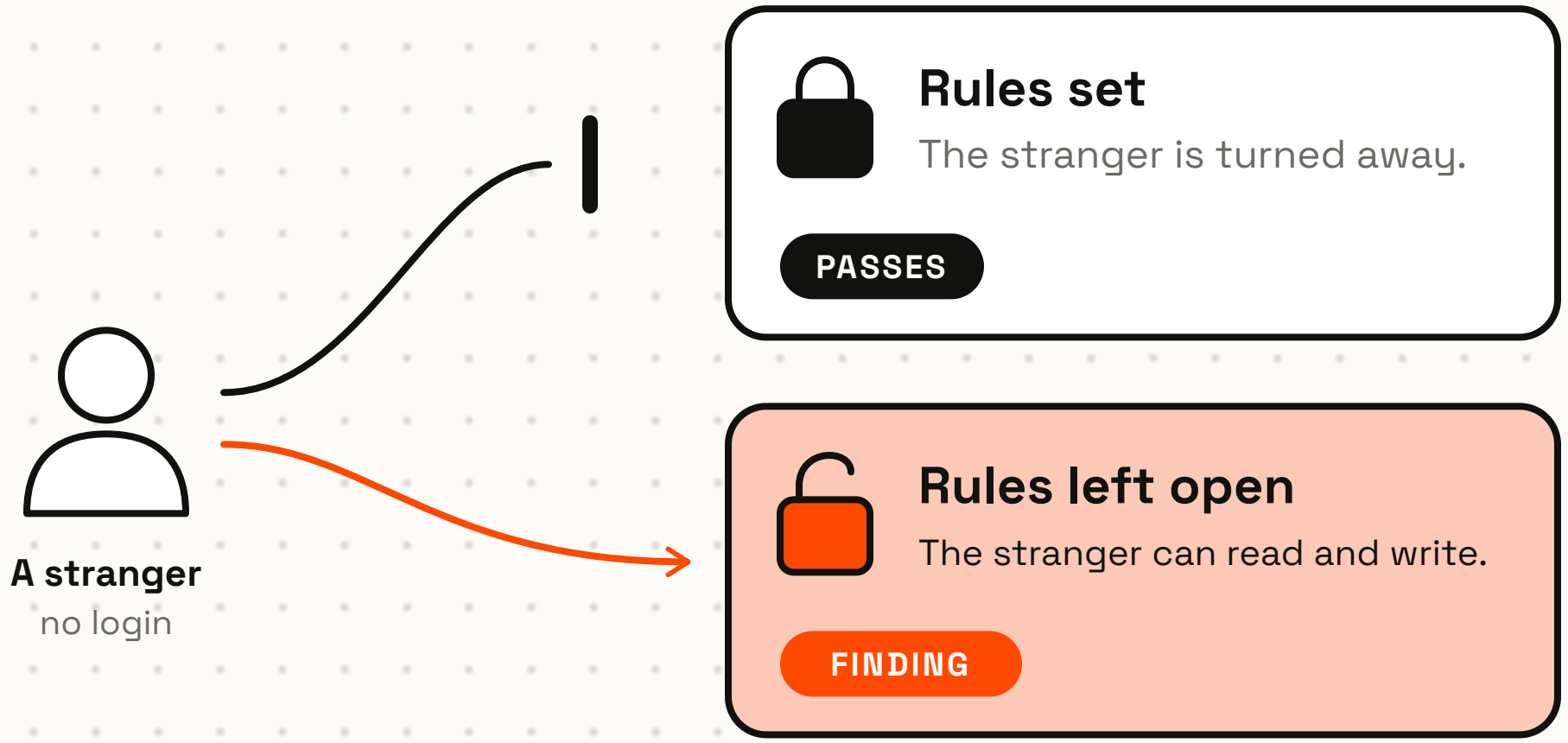
✓ Should be allowed ✗ Should be blocked ! Blocked, but we got in

Illustrative example. This is why we ask for a list of your roles, and what each one should be able to do.

Your cloud *settings.*

Can a stranger read or write your data? This matters most for apps built with AI tools.

RULES CHECK SUPABASE / FIREBASE



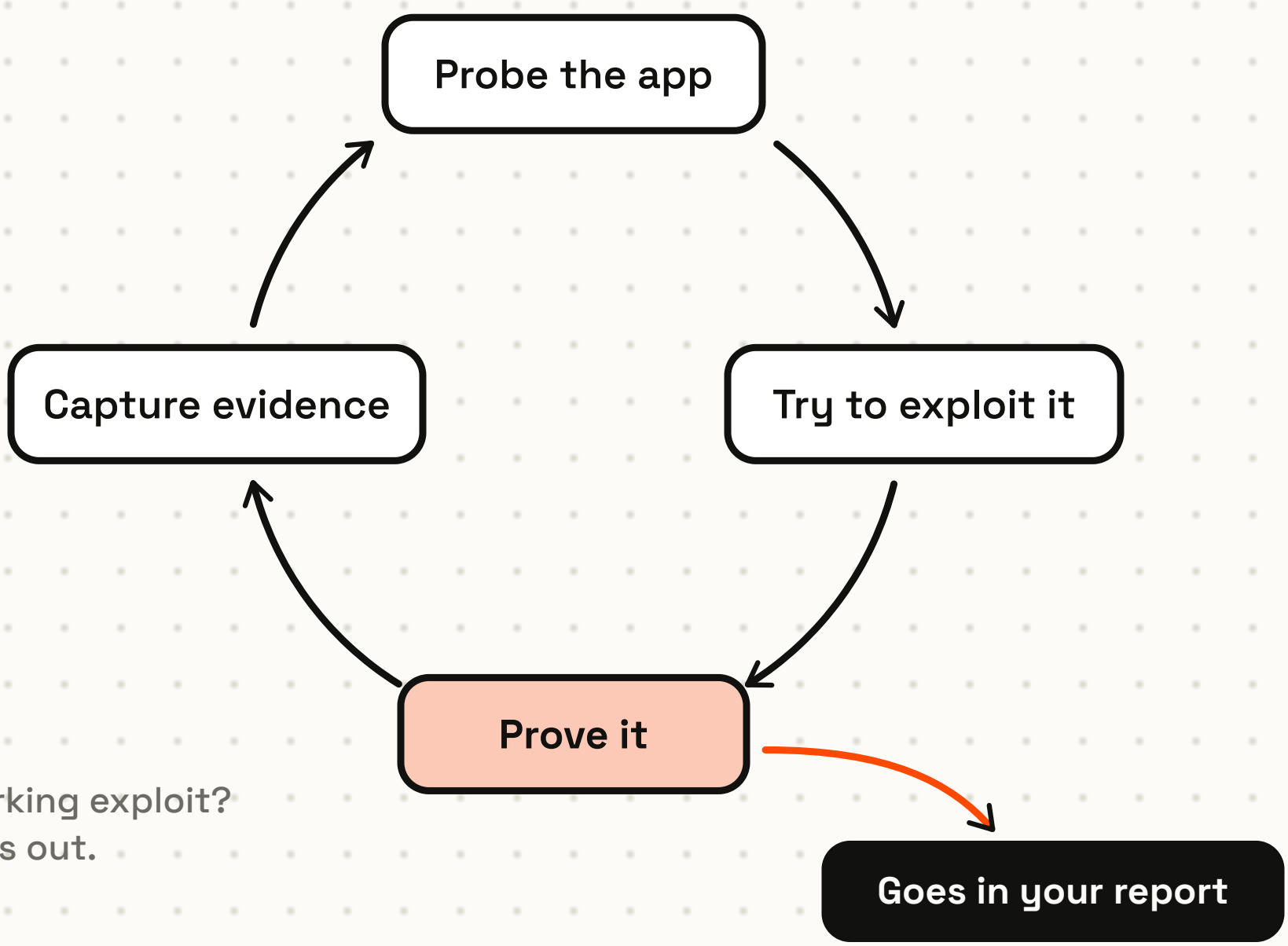
Database rules · Storage buckets · Settings only, never billing

Illustrative example. For this check we ask for read access to your cloud settings.

No proof, *no report.*

We attack your app the way a real attacker would, with your written permission, and prove every finding with a working exploit.

THE METHOD



Guided by OWASP, NIST and PTES. Every finding gets proof, a fix and a priority.

About 8 weeks, start to *closed.*

Three steps. You hear about critical problems as soon as we prove them, not at the end.

THE TIMELINE

1 We test · 3 weeks

Critical problems reach you the moment we prove them.

2 Report + letter · within 5 working days

3 You fix · we recheck once, free

30 days from the report

Start · Wk 1 · Wk 2 · Wk 3 · Wk 4 · Wk 5 · Wk 6 · Wk 7 · Wk 8

Anatomy of a *finding.*

Every finding carries proof, a fix and a priority. Plain English first, technical terms in brackets.

EXAMPLE FINDING · NOT CLIENT DATA

FIX NOW

Any stranger can open customers' files without logging in storage buckets

PROOF

> GET /storage/files/20417.pdf with no login

200 OK

{ "file": "[redacted]" }

THE FIX

Set the storage rules so that only the file's owner can read it.

THREE PRIORITIES

FIX NOW | FIX THIS WEEK | FIX SOON

What we need.

EVERY PLAN

The web address of your app

Two test logins: one user, one admin

One person we can ask questions

FOR BREACH, ALSO

Your API notes, if you have them.

A list of your user roles, and what each role should be able to do.

Read access to your cloud settings, if cloud checks are included. **Settings only, never billing.**

Our rules, *every test.*

01 **Written permission and an agreed scope** come before any testing starts. We test nothing outside it.

02 We never **delete data, crash your site,** email your staff, or test systems you do not own, unless agreed in writing under Custom.

03 Production gets **safe, gentle checks.** A staging copy is better when you have one.

04 Proof stays **private** and is shared only with you. We publish nothing without your written permission.

05 **Critical problems** are told to you as soon as we prove them. We do not hold them for the final report.

06 You can **stop the testing** at any time by telling your contact.

What Breach *leaves out.*

NOT IN BREACH

- ✕ Your internal network
- ✕ Emails to your staff
- ✕ Systems owned by other companies

THOSE BELONG IN

Custom quoted after a free scoping call

Several apps and APIs, your whole company with written permission, and evidence mapped to SOC 2 and ISO 27001 controls.

Not sure yet? Start with Strike. Go to Breach within 60 days and your Strike fee counts toward it.

Price and how you pay.

BREACH · FIXED PRICE

$4,500

Agreed before we start.

INCLUDED

3 weeks of testing, the audit-ready report, plain-English summary, attestation letter, walkthrough call and one free retest.

TWO PAYMENTS

50%
When you sign the scope

→

50%
When the watermarked draft report arrives

→

Final payment
clears, then the clean report and the letter are yours

You read the draft first, then pay the balance.

Our promise, in *one line.*

THE GUARANTEE

If your auditor, insurer or enterprise customer rejects this report as a valid penetration test, **we refund you in full.**

Send us the written rejection within 30 days of the report. We do not promise a clean result, and we do not promise to find problems. We promise a real test and a report that stands up.

START FREE

Try the free passive check. No card needed.

A look at what anyone on the internet can already see about your app. No attack, and no password needed.

BOOK A FREE SCOPING CALL

www.garlicthreat.tech

GARLIC THREAT

STAY AHEAD OF THE THREAT.