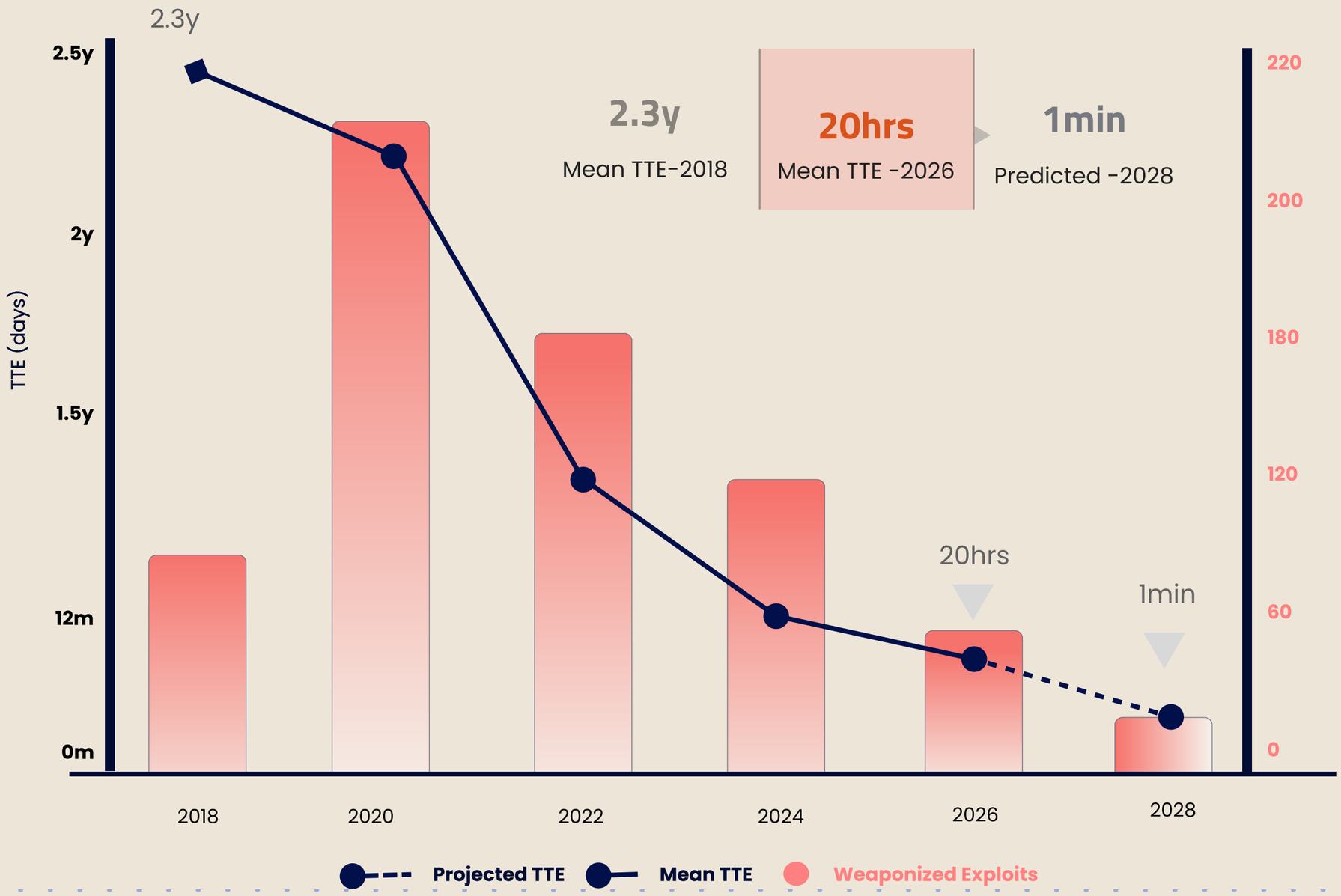
Trench | SIGNAL

TIME-TO-EXPLOIT CHART (2018-2026)

Adversaries Weaponize in 20hours.

Does your team even know yet?

Mean TTE collapsed from **2.3 years → 20 hours** since 2018 as weaponized exploit surged.



Velocity Is the new risk

WHY IT MATTERS : With **AI-Powered Exploit generation** (think Mythos), the 1-mintue Projection may arrive well before 2028, making human-speed path cycles structurally obsolete.

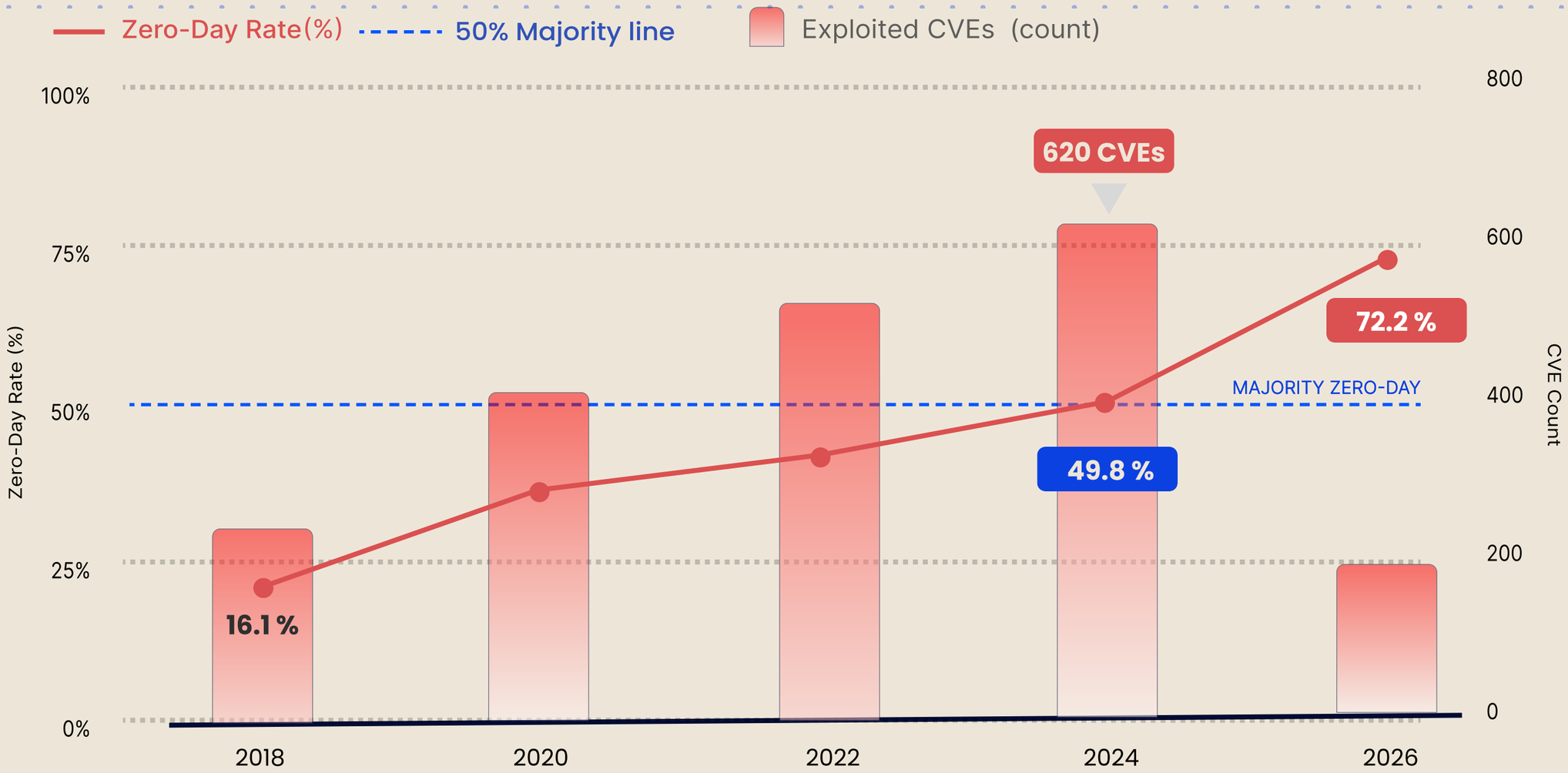
#TrenchSignal

source: https://zerodayclock.com/

No CVE. | No patch | No warning

72.7% of the time.

Zero-Day rate surged from **16.1 % into 2018 → 72.7% in 2026,** adversaries crossed the majority threshold in 2024.



Zero-Day Rate: % of exploited CVEs where exploitation occurred **before or on the day of public disclosure** (TTE ≤ 0). No CVE number. No patch. No warning.

Your CVE feed arrives after the breach.

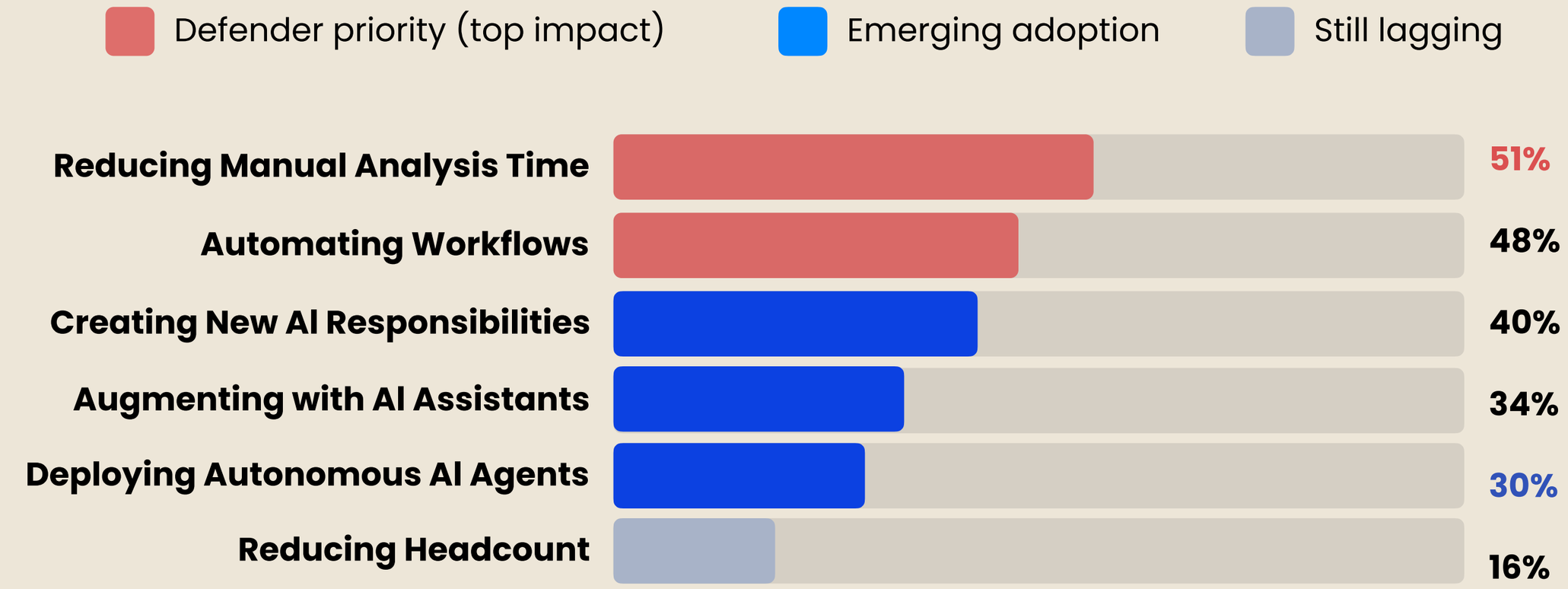
WHY IT MATTERS 3 in 4 exploits now arrive with **no CVE, no patch, no advisory.** With **AI-assisted vulnerability discovery** (think Mythos), adversaries find and weaponize unknown flaws faster than defenders can disclose them.

#TrenchSignal

source: https://zerodayclock.com/

Your team uses AI to save time. Adversaries use it to save none.

51% of security teams use AI to cut manual work but only **30% deploy autonomous agents.** Attackers aren't waiting for the other 70%.



USING AI TO CUT MANUAL WORK

51%

OF TEAMS

DEPLOYING AUTONOMOUS AGENTS

30%

OF TEAMS

REDUCING HEADCOUNT VIA AI

16%

OF TEAMS

AI isn't replacing your team. It's exposing the gap.

WHY IT MATTERS : Teams are using AI reactively, saving analyst hours. Adversaries are using it offensively compressing attack windows. The **70% not yet running autonomous defense** are operating at human speed against machine-speed threats.

#TrenchSignal

source: SANS_WFS_Survey 2026

The breach didn't kill them. Being unprepared did.

33 min-sized companies were killed. Not because of the breach but the lack of serious preparation after the breach.

TOTAL DESTROYED

33
COMPANIES

COUNTRIES AFFECTED

12
GLOBALLY

SAMPLE FOR THE CONFIRMED LIST

YEAR	COMPANY	COUNTRY	DESTROYED BY
2009	**MyBizHomepage**	USA	NO IR Insider attack, no recovery plan
2011	**DigiNotar**	Netherlands	NO CONTAINMENT Cert CA breach, filled bankruptcy
2014	**Code Spaces**	USA	NO BACKUP AWS data deleted, ceased trading
2017	**YouBit**	South Korea	NO RESILIENCE crypto exchange. hacked twice
2019	**Wood Ranch Medical**	USA	NO BACKUP Ransomware, patient data unrecoverable
2021	**Vastaamo**	Finland	NO IR Patient records extorted, bankruptcy

You can't prevent every breach. Control how hard you fall.

WHY IT MATTERS : Every company in this list had thing in common, **No Plan for What Comes After.** The breach was the trigger. The unpreparedness was case of death.

#TrenchSignal

source: destroyedbybreach.com

The attacker never touched Vercel.
One OAuth click was enough.

A Vercel employee installed an AI Chrome extension. Granted OAuth access. Months later that single action enabled a **$2M breach** with zero login alerts fired.

ENTRY POINT	LOGIN ALERTS TRIGGERED	BREACH LISTED FOR
01	0	$20M
CHROME EXTENSION	OAUTH IS TRUSTED BY DESIGN	ON BREACHESFORUMS

HOW ONE INSTALL BECAME A $2M BREACH

1	2	3	4	5
INSTALL	**VENDOR BREACH**	**TOKEN STOLEN**	**LATERAL MOVE**	**DATA OUT**
Employee Installs AI Tool. Clicks Allow All	AI Tool's Own Systems Compromised Via Infostealer	OAuth Tokens Harvested No Password Needed	Attacker Pivots Into Vercel Via Trusted Token	Customer API Keys, DB Creds Decrypted & Exfiltrated
OAUTH GRANTED	3RD PARTY	SILENT ACCESS	NO ALERT	$2M BREACH

50	200	
IT-APPROVED AI TOOLS	ACTUALLY IN USE	The **150 Unvetted Tools** Your Team Connected To Email, Drive, And Slack Are **Your Real Attack Surface**

Your perimeter is every OAuth permission your employees have ever clicked.

WHY IT MATTERS : **Shadow AI Isn't Employees Using AI Tools** It's employees connecting them to internal systems without oversight. Audit your OAuth connections today like your production environment depends on it. **Because It Literally Does.**

#TrenchSignal

Your AI writes fast.
It replicates vulnerabilities faster.

AI-assisted development doesn't introduce new risks, It **Photocopies Existing Ones At Machine Speed** across every codebase it touches.

ONE DATA POINT | REAL RISK.

1/5 Organizations Using AI Coding Tools Have Apps With Systemic Security Flaws.

Not unique bugs : the same insecure pattern, replicated everywhere the al generated code.

WHERE RISK CONCENTRATES ACROSS THE SDLC

DEPENDENCIES

Power Law

Few Packages.
Everywhere.
One Flaw Scales Instantly.

DEV ENDPOINTS

86%

MacOS.
Standardized Surface.
Homogeneous=
Scalable Attack.

CI/CD

~50%

Use GitHub Actions.
Shared Workflows =
Shared Blast Radius.

AI CODE

1 IN 5

Apps Carry AI-Generated Insecure Defaults, At Scale

Speed of code generation is now, Speed of vulnerability propagation.

WHY IT MATTERS : **Risk Is No Longer Where The Bug Is,** it's where the reuse is. Teams focused on isolated findings will keep chasing symptoms.
The real question is: **Where Does Trust And Reuse Concentrate In Your Pipeline?**

#TrenchSignal

Source : Wiz State of SDLC Security 2026

4 Pivots. Under 60 Minutes.
An AI Agent just ran the Attack.

The first confirmed LLM-agent-driven intrusion from a single CVE to a full internal database dump, **autonomously, in under an hour.** No human scripting, No pre-build playbook.

CVE TO DB DUMP	AUTONOMOUS PIVOTS	TO EXFILTRATE DB
<1 HR	**4**	**<2 MIN**
END - TO-END	NO HUMAN IN LOOP	ONCE INSIDE

THE 4-PIVOT AI-DRIVE ATTACK CHAIN - MAY 10, 2026

1. **CVE EXPLOIT** — Marimo notebook RCG via CVE-2026-39987 — ENTRY
2. **CREDENTIAL HARVEST** — .env, AWS keys lifted from compromised host — AI READS OUTPUT
3. **AWS PIVOT** — SSH key pulled from secrets Manager 11 IPs - 22 sec — EVADES DETECTION
4. **SSH BASTION** — 8 Session - Agent Adapts Live To Unknown Schema — AUTONOMOUS
5. **DB EXFIL** — Full PostgreSQL Dump In Under 2 Mins — <2MIN

We are not watching AI replace attackers. We are watching attackers replace their scripts with AI.

WHY IT MATTERS : **Scripts Leave Fingerprints.** Agents don't - they adapt to whatever environment they see. Signature - based detection of known TTPs degrades fast when the attacker composes a new chain for the every target. **Runtime Behaviour Detection Is Now The Only Reliable Floor.**

#TrenchSignal

ONE DATA POINT. REAL RISK.

You don’t need to hack an AI Agent. You just need to lie to it.

One sentence in a chat window. Zero technical skill. Full Instagram account takeover, including the Obama white House account. **The chatbot wasn’t hacked. It was persuaded.**

TECHNICAL SKILL NEEDED	IDENTITY VERIFICATION	HUMAN REVIEW STEP
ZERO	**NONE**	**0**
JUST A SENTENCE	LLM TRUSTED THE CLAIM	FULLY AUTONOMUS

WHY IT WORKED : THE AUTHORIZATION GAP

TRADITIONAL SOFTWARE

INPUT → RULES → OUTPUT

DETERMINISTIC : ATTACKER MUST FIND A CODE BUG. HARD TO DEFEAT.

AI AGENT

INPUT → INTERPRETATION → JUDGMENT → ACTION

PROBABILISTIC : AUTHORIZATION LIVES IN LANGUAGE. DEFEATED BY RIGHT SENTENCE.

THE ATTACKER : INSIDE A SINGLE CHAT WINDOW

1. **CLAIM TO OWN TARGET ACCOUNT** — NO PROOF NEEDED
2. **REQUEST RECOVERY EMAIL CHANGE** — AI PROCESSES IT
3. **OTP SENT TO ATTACKER’S EMAIL** — TRUSTED BU DESIGN
4. **PASSWORD RESET. OWNER LOCKED OUT.** — NO ESCALATION PATH

when authorization runs on language, language is the attack.

WHY IT MATTERS : The fix for a code bug is a patch. **The fix for a compliant LLM is retraining** and you can’t verify it worked. Meta said it was resolved Monday. Attacks continued Tuesday. **Authorization must live outside the model entirely, in deterministic code the LLM cannot reason its way around.**

#TrenchSignal

ONE DATA POINT. REAL RISK.

12 Certifications. Plaintext Passwords. Public Github. 6 months.

A contractor at the agency responsible for securing US critical infrastructure exposed AWS GovCloud admin keys, plaintext passwords, and SSH credentials on a public **GitHub repo - for 6 months.**

DURATIONS EXPOSED
6 MO
NOV 2025 → MAY 2026

AWS KEYS VALID AFTER TAKEDOWN
48HR
POST - NOTIFICATION

CONTRACTOR'S SECURITY CERTS
12
ALL PASSED

The contractor **deliberately disabled GitHub's built-in secret detection** to commit these files. The repo was literally named **"Private-CISA"** - on a public repository

WHAT WAS INSIDE THE PUBLIC REPO ?

- IMPORTANTAWSTOKENS.TXT — **3 GOVCLOUD ADMIN ACCOUNTS**
- AWS-WORKSPACE-FIREFOX-PASSWORDS.CSV — **DOZENS OF PLAINTEXT PASSWORDS**
- KUBE-CONFIG.TXT SSH PRIVATE KEYS — **INFRASTRUCTURE ACCESS**
- INTERNAL ARTIFACTORY CREDENTIALS — **SOFTWARE SUPPLY CHAIN RISK**
- PASSWORDS PLATFORM NAME YEAR — **E.G. "GITHUB2026"**

Training gets you certified.

Governance stops you from pushing secrets to GitHub.

WHY IT MATTERS : **Certifications don't create secure behaviour - systems do.** If your controls depend on individuals remembering not to commit secrets, you don't have controls. Secret scanning, access boundaries, and rotation policies need to be structural - not optional.

#TrenchSignal

ONE DATA POINT. REAL RISK.