EACG
Enterprise Architecture Consulting Group

SECURITY DIALOGUES
No. 14 · Part 2 of 2 · 6 November 2026
Marcus Hale (PSIRT) · Tess Calder (red team) · Jessy Lang

Unsigned Instructions

Seven threats and seven rules for AI inside a product

Everything an AI system reads might be an instruction, and nothing proves who wrote it. The model cannot reliably tell friend from foe, so assume it can be persuaded and design so that persuasion is not enough. The episode uses boostME, an AI coach that reads heart-rate data and adapts training plans, as its example.

UNSIGNED INSTRUCTIONS
Any text an AI reads may act as an order.

1. What can it read?

2. What can it do?

3. Where does the output go?

THREATS AND WHAT TO BUILD IN AGAINST THEM

1	**Unsafe plans** A wrong or overconfident answer becomes physical harm.	The model proposes, deterministic rules written by people who know the domain decide: limits on load and intensity, mandatory rest, a fixed response for danger signs. Decide your intended purpose deliberately, write it down and make every claim match. Fall back to a conservative static default.
2	**Garbage in, injury out** Wrong or fake sensor readings drive decisions.	Plausibility checks on every input, sensor quality flags, patterns instead of single readings, authenticated device data where possible. Ask what happens at the extremes: zero, three hundred, a flat line.
3	**Unsigned instructions** Text the product reads acts as an order.	Limit what a successful trick can reach: no rights beyond the functions the dialogue needs, no internet, scoped keys. Add an injection guard as the first line, not the last. Validate data on every other path. Review the guard's scope whenever a new text source appears. Test with hostile content.
4	**Whose heartbeat is this?** Sensitive data also lives around the vault.	Extend the key, the logged access and the grant to everything the model derives. Keep grants narrow by purpose and kind of coach, out of reach of any text. Watch the access log yourself. Treat inference servers and their logs like the vault, and test the separation between users.
5	**The coach that can call home** Tools act on the model's word; usage runs away.	The backend authorizes by the logged-in user's identity, never by what the model asks for. Authorization lives in the backend, not in the prompt. Rate and length limits, a budget alarm that someone receives, no secrets in prompts.
6	**Supply chain under the coach** Model, inference stack, libraries and prompts all change.	A bill of materials that includes the AI parts. Patching that someone owns, also for local models. Verified sources. Reference scenarios you rerun whenever anything changes.
7	**The voice that sounds sure** Invented advice delivered with confidence.	Ground answers in a library vetted by qualified people. Allow the answer "I don't know, ask a person". Keep a human in reach and a way to flag bad advice, with a team that reads the flags.

THREE PRACTICES ON TOP

Threat-model before release
Draw the system, ask the three questions at every arrow, write down the controls.

Red-team it first
Give someone permission to be hostile: through the chat, the data path, fake sensor data and every import.

Keep a way out
Be able to switch the AI off within minutes and fall back to a conservative static mode. A product that cannot survive without its AI has a single point of failure.

Listen to the episode
Unsigned Instructions and the rest of the series.
eacg.de
/en/podcast/s14-unsigned-instructions

Digital Product Design
From the first idea to a viable first release.
eacg.de
/en/consulting/digital-products

PSIRT Services
Intake, assess and disclose vulnerabilities, built or run as a service.
eacg.de
/en/consulting/psirt

© 2026 EACG GmbH