CHECKLISTE VOR DEM PRODUKTIVGANG

Prompt-Injection-Checkliste

Zwanzig prüfbare Punkte aus den OWASP-Empfehlungen zu LLM01 und der BSI-Handreichung zu Evasion Attacks — jeder mit Prüfweg und mit dem Kriterium, an dem er durchfällt.

Die Reihenfolge folgt dem Weg der Daten: erst Entwurf und Rechte, dann Eingang, dann Ausgang, zuletzt Betrieb.

Stand: 30. August 2026 · **Gökhan Köse**

promptinjections.de/schutz/checkliste

20 Punkte · 4 Prüfblöcke

Quellen: OWASP · BSI

WARUM DIESE LISTE

Eine einzige E-Mail genügte — ohne Klick, ohne Bestätigung

VORFALL · JUNI 2025

EchoLeak machte vor, wie wenig nötig ist: Eine präparierte E-Mail brachte Microsoft 365 Copilot zum Abfluss interner Daten — ohne dass das Opfer klickte, öffnete oder bestätigte. Die Lücke war kein Modellfehler, sondern eine Architektur, in der Fremdinhalt, private Daten und ein Weg nach außen zusammentrafen.

Diese Checkliste prüft genau das. Wer wenig Zeit hat, beginnt bei Punkt 1 — die tödliche Trias entscheidet, wie schwer ein Treffer wiegt.

SO IST DIE LISTE SORTIERT

Der Weg der Daten: Entwurf und Rechte → Eingang und Kontext → Ausgang und Wirkung → Betrieb und Nachweis.

SO WIRD GEPRÜFT

Jeder Punkt nennt einen **Prüfweg** und ein **Durchfall-Kriterium**. Ein Punkt gilt erst als erfüllt, wenn das Kriterium nachweislich nicht zutrifft.

WER PRÜFT WAS

Wer alles allein abhakt, hakt zu schnell ab

Die Liste ist nicht für eine Person gedacht. Vier Rollen teilen sie sich auf.

PUNKTE 1-5, 9, 13

Architektur und Entwicklung

Entscheiden über Rechte, Werkzeugzuschnitt und Ausführungsumgebung, bevor Code entsteht.

PUNKTE 4, 15, 16, 17, 20

Betrieb und Plattform

Verantworten Token-Laufzeiten, ausgehende Ziele, Protokolle und den Ernstfall.

PUNKTE 8, 10, 18, 19

Sicherheit und Red Team

Greifen an, statt zu bestätigen: versteckte Zeichen, auslesbarer System-Prompt, Testlauf, Speicherinhalt.

PUNKTE 5, 11, 14

Fachbereich und Freigabe

Legen fest, welche Aktion einen Menschen braucht und welches Ausgabeformat nachgelagerte Systeme akzeptieren.

I

PUNKTE 1 BIS 5

Entwurf und Rechte

Was hier falsch entschieden wird, lässt sich später nicht filtern. Rechte, Werkzeuge und Freigabepunkte werden festgelegt, bevor Code entsteht.

PUNKT 1 VON 20 · KERNPRÜFUNG

Tödliche Trias geprüft

PRÜFEN

Hat das System gleichzeitig Zugriff auf private Daten, verarbeitet es Fremdinhalte und kann es nach außen kommunizieren?

Treffen alle drei zu, ist Datenabfluss nur eine Frage der Gelegenheit — eine der drei Eigenschaften muss weg.

Siehe: promptinjections.de/schutz/toedliche-trias

DURCHGEFALLEN, WENN

niemand benennen kann, welche der drei Eigenschaften das System **nicht** hat.

ZUSTÄNDIG

Architektur und Entwicklung · Wer wenig Zeit hat, beginnt hier: Dieser Punkt entscheidet, wie schwer jeder Treffer wiegt.

PUNKTE 2 UND 3

02 Rechte aus der Aufgabe abgeleitet

ARCHITEKTUR

PRÜFEN

Zu jedem Recht den konkreten Arbeitsschritt benennen, der es braucht. Rechte, die aus der Rolle des aufrufenden Nutzers kopiert wurden, sind fast immer zu weit.

DURCHGEFALLEN, WENN

der Agent mehr darf, nur weil der Nutzer mehr darf.

Siehe: promptinjections.de/schutz/least-privilege

03 Werkzeugsatz zugeschnitten

ARCHITEKTUR

PRÜFEN

Die pro Anfrage verfügbare Werkzeugliste ausgeben lassen und gegen die Aufgabe halten. Ein globaler Katalog macht jede Injektion mächtiger, als sie sein müsste.

DURCHGEFALLEN, WENN

das Modell Werkzeuge sieht, die es für diese Aufgabe nie braucht.

PUNKTE 4 UND 5

04 Zugangsdaten kurzlebig

BETRIEB

PRÜFEN

Laufzeit und Geltungsbereich jedes Tokens nachsehen, das in den Kontext gelangt. Dauerhafte Anmeldedaten überleben die Sitzung und wandern mit jeder Ausgabe mit.

DURCHGEFALLEN, WENN

ein Token im Kontext steht, das morgen noch gilt.

05 Menschliche Freigabe definiert

FACHBEREICH

PRÜFEN

Für jede Aktion mit Geld-, Daten- oder Außenwirkung schriftlich festhalten, wer freigibt. Ohne benannten Freigebenden ist die Freigabe ein Vorsatz, kein Kontrollpunkt.

DURCHGEFALLEN, WENN

eine folgenreiche Aktion allein vom Modell ausgelöst werden kann.

Einordnung: OWASP LLM06 — Übermäßige Handlungsvollmacht

II

PUNKTE 6 BIS 10

Eingang und Kontext

Jeder Weg, auf dem fremder Text in den Kontext gelangt, ist ein Eingang. Was nicht auf der Liste steht, wird auch nicht geprüft.

PUNKTE 6 UND 7

06 Datenquellen inventarisiert

ARCHITEKTUR

PRÜFEN

Alle Wege auflisten, auf denen fremder Text in den Kontext gelangt — Dateien, Web, E-Mail, Werkzeugergebnisse, Gedächtnis. Was nicht auf der Liste steht, wird auch nicht geprüft.

DURCHGEFALLEN, WENN

die Liste beim Nachfragen noch wächst.

Siehe: promptinjections.de/angriffsvektoren

07 Fremddaten markiert

ARCHITEKTUR

PRÜFEN

Im tatsächlich gesendeten Prompt nachsehen, ob abgerufener Text erkennbar von Betreiber-Anweisungen getrennt ist. Markierung senkt die Trefferquote, ersetzt aber keine Rechtegrenze.

DURCHGEFALLEN, WENN

Fremdtext und Anweisung im selben Block stehen.

Siehe: promptinjections.de/schutz/spotlighting

PUNKTE 8 UND 9

08 Unsichtbare Zeichen entfernt

RED TEAM

PRÜFEN

Mit einem Testdokument arbeiten, das Unicode-Tag-Zeichen, Zero-Width-Zeichen oder Homoglyphen enthält. Was der Mensch nicht sieht, liest das Modell trotzdem.

DURCHGEFALLEN, WENN

versteckter Text die Antwort verändert.

Siehe: promptinjections.de/schutz/eingabepruefung

09 Anweisungen bleiben in der Systemrolle

ARCHITEKTUR

PRÜFEN

Suchen, ob Fremddaten jemals in die Systemrolle geraten — etwa über Zusammenfassungen oder Vorlagen mit Platzhaltern.

DURCHGEFALLEN, WENN

abgerufener Text in der Rolle landet, der das Modell die höchste Priorität zuweist.

PUNKT 10 VON 20

System-Prompt ohne Geheimnisse

PRÜFEN

Davon ausgehen, dass der System-Prompt öffentlich ist, und ihn mit diesem Blick lesen.

Schlüssel, interne Regeln und Preisgrenzen gehören nicht hinein.

Siehe: promptinjections.de/techniken/prompt-leaking

DURCHGEFALLEN, WENN

sein Auslesen einen Schaden bedeuten würde.

WAS NICHT IN DEN SYSTEM-PROMPT GEHÖRT

API-Schlüssel · Zugangsdaten · interne Preis- und Rabattgrenzen · Namen interner Systeme · Umgehungsregeln

III

PUNKTE 11 BIS 15

Ausgang und Wirkung

Freitext, der als Steuerbefehl weiterverwendet wird, ist die Grundlage der meisten Folgeschäden. Hier entsteht der Unterschied zwischen Störung und Schaden.

PUNKTE 11 UND 12

11 Ausgabeformat wird geprüft

FACHBEREICH

PRÜFEN

Ein Schema festlegen und jede Antwort dagegen validieren, bevor sie weiterläuft. Freitext, der als Steuerbefehl weiterverwendet wird, ist die Grundlage der meisten Folgeschäden.

DURCHGEFALLEN, WENN

eine unerwartete Ausgabe nachgelagerte Systeme erreicht.

Siehe: promptinjections.de/schutz/ausgabepruefung

12 Links und Ressourcen bereinigt

ARCHITEKTUR

PRÜFEN

Markdown-Bilder und Links in der Ausgabe testen — ein Bild mit Daten in der URL genügt für den Abfluss.

DURCHGEFALLEN, WENN

die Ausgabe eine Ressource von einem beliebigen fremden Host laden darf.

Siehe: Datenabfluss über Markdown-Bilder

PUNKTE 13 UND 14

13 Code nur in der Sandbox

ARCHITEKTUR

PRÜFEN

Netzzugang und hinterlegte Geheimnisse der Ausführungsumgebung nachsehen. Modellgenerierter Code ist nicht vertrauenswürdiger Code.

DURCHGEFALLEN, WENN

generierter Code ins Netz kommt oder dauerhafte Zugangsdaten sieht.

14 Ausgaben gelten als Fremdeingabe

FACHBEREICH

PRÜFEN

Jede Stelle suchen, an der eine Modellausgabe in ein anderes System fließt, und dort dieselbe Validierung verlangen wie bei Nutzereingaben.

DURCHGEFALLEN, WENN

ein nachgelagertes System der Ausgabe vertraut, nur weil sie vom eigenen Modell stammt.

Siehe: OWASP LLM05 — Improper Output Handling

PUNKT 15 VON 20

Ausgehende Ziele eingeschränkt

PRÜFEN

Die Liste erlaubter Ziele für Netzaufrufe ansehen — sie sollte kurz sein und jeder Eintrag begründet.

Ohne Begrenzung ist jeder Kanal nach außen ein möglicher Abflussweg — auch ein reiner Bildaufruf.

DURCHGEFALLEN, WENN

das System eine beliebige Adresse erreichen kann.

ZUSTÄNDIG

Betrieb und Plattform · Zusammen mit Punkt 1 der zweite Hebel gegen Datenabfluss: der Weg nach außen.

IV

PUNKTE 16 BIS 20

Betrieb und Nachweis

Ein bestandener Lauf beweist keine Sicherheit, ein durchgefallener beweist eine Lücke. Was im Nachhinein nicht rekonstruierbar ist, ist nicht geprüft.

PUNKTE 16 UND 17

16 Werkzeugaufrufe protokolliert

BETRIEB

PRÜFEN

Das Protokoll eines echten Laufs lesen — es muss zeigen, welches Werkzeug mit welchen Argumenten lief. Dazu Ratenlimits und Budgets.

DURCHGEFALLEN, WENN

sich im Nachhinein nicht rekonstruieren lässt, was der Agent getan hat.

17 Verhalten überwacht, nicht nur Text

BETRIEB

PRÜFEN

Auf ungewöhnliche Handlungsketten messen — viele Aufrufe hintereinander, neue Ziele, plötzliche Rechteanfragen.

DURCHGEFALLEN, WENN

die Überwachung ausschließlich Eingabetext filtert.

Siehe: promptinjections.de/schutz/erkennung

PUNKTE 18 UND 19

18 Testlauf ist Teil der Freigabe

RED TEAM

PRÜFEN

Vor dem Produktivgang einen dokumentierten Angriffslauf fahren, tokenauffällige und flüssig formulierte Eingaben getrennt. Ein bestandener Lauf beweist keine Sicherheit, ein durchgefallener beweist eine Lücke.

DURCHGEFALLEN, WENN

nie jemand versucht hat, das System zu brechen.

Siehe: promptinjections.de/schutz/penetrationstest

19 Speicher prüfbar und rückrollbar

RED TEAM

PRÜFEN

Gedächtnis und Index einsehen, durchsuchen und einzelne Einträge entfernen können. Eine einmal abgelegte Injektion wirkt bei jedem späteren Abruf erneut.

DURCHGEFALLEN, WENN

niemand sagen kann, was aktuell im Speicher steht.

Siehe: gespeicherte Prompt Injection

PUNKT 20 VON 20

Verfahren für den Ernstfall

PRÜFEN

Benennen, wer welchen Zugang abschaltet, wie der Umfang eingegrenzt und wer informiert wird.

Der Ernstfall ist der falsche Zeitpunkt, das zu klären.

DURCHGEFALLEN, WENN

auf die Frage, wer abschaltet, niemand antwortet.

IM VERFAHREN NAMENTLICH FESTGEHALTEN

Abschaltung Zugänge · Eingrenzung Umfang · Information Betroffene · Information Aufsicht · Wiederanlauf

KURZFASSUNG ZUM ABHAKEN

Die zwanzig Punkte ohne Erläuterung

Zum Kopieren in ein Ticket, ein Freigabeprotokoll oder ein Review-Dokument.

ENTWURF UND RECHTE

- [ ] 01 Tödliche Trias geprüft
- [ ] 02 Rechte aus der Aufgabe abgeleitet
- [ ] 03 Werkzeugsatz zugeschnitten
- [ ] 04 Zugangsdaten kurzlebig
- [ ] 05 Menschliche Freigabe definiert

EINGANG UND KONTEXT

- [ ] 06 Datenquellen inventarisiert
- [ ] 07 Fremddaten markiert
- [ ] 08 Unsichtbare Zeichen entfernt
- [ ] 09 Anweisungen bleiben in der Systemrolle
- [ ] 10 System-Prompt ohne Geheimnisse

AUSGANG UND WIRKUNG

- [ ] 11 Ausgabeformat wird geprüft
- [ ] 12 Links und Ressourcen bereinigt
- [ ] 13 Code nur in der Sandbox
- [ ] 14 Ausgaben gelten als Fremdeingabe
- [ ] 15 Ausgehende Ziele eingeschränkt

BETRIEB UND NACHWEIS

- [ ] 16 Werkzeugaufrufe protokolliert
- [ ] 17 Verhalten überwacht, nicht nur Text
- [ ] 18 Testlauf ist Teil der Freigabe
- [ ] 19 Speicher prüfbar und rückrollbar
- [ ] 20 Verfahren für den Ernstfall