vge-cc-guard
Przewodnik użytkownika

Lokalny sidecar bezpieczeństwa dla Claude Code. Pomaga zespołom używać Claude Code bez cichego przepuszczania wywołań narzędzi, pobranych treści, odczytów plików albo załączników promptów do kontekstu modelu, gdy są sprzeczne z lokalną polityką albo decyzją Vigil Guard Enterprise (VGE).

W SKRÓCIE

» Wymaga Node.js ≥ 20.10.0

» Interaktywny konfigurator TUI

» 7 obsługiwanych typów zdarzeń hook

» Bramka per narzędzie: allow / ask / block

» Ochrona ścieżek poświadczeń domyślnie włączona

» URL Access Baseline z 5 presetami

» Decyzje allow / block w zakresie sesji

» Dziennik audytu JSONL z filtrami

» Przeładowanie konfiguracji bez restartu CC

Node.js ≥20.10	7	10	5
wymagane środowisko	typy zdarzeń hook	polityki narzędzi	presety URL baseline

Zakres użytkownika | Zakres projektu | Interaktywny TUI | Polityki bramek per narzędzie | Dziennik audytu JSONL

DLA KOGO JEST TEN PRZEWODNIK

Użyj tego przewodnika, jeśli jesteś:

» Deweloperem używającym Claude Code z zainstalowanym `vge-cc-guard`.
» Liderem zespołu ustawiającym domyślną politykę dla projektu.
» Inżynierem bezpieczeństwa albo platformy pomagającym użytkownikom zrozumieć zablokowane działania.
» Operatorem wsparcia analizującym konfigurację użytkownika albo zdarzenia audytowe.

To przewodnik skierowany do użytkownika. Celowo skupia się na obserwowalnym zachowaniu, wyborach konfiguracyjnych i bezpiecznych praktykach operacyjnych. Nie opisuje wewnętrznych szczegółów implementacji poza tym, co użytkownicy muszą wiedzieć, aby obsługiwać narzędzie.

MODEL MENTALNY

Claude Code może poprosić o uruchomienie narzędzi takich jak `Read`, `Bash`, `WebFetch`, `Edit` albo `Write`. Niektóre z tych narzędzi mogą odczytywać wrażliwe pliki, pobierać niezaufane treści z internetu, uruchamiać komendy powłoki albo modyfikować repozytorium. vge-cc-guard działa obok Claude Code i odpowiada na trzy praktyczne pytania:

#	PYTANIE	JAK JEST ODPOWIEDZIANE
1	Czy to narzędzie w ogóle może zostać uruchomione?	Polityka bramki: allow, ask albo block przed wykonaniem.
2	Czy wynik narzędzia jest wystarczająco bezpieczny, aby przekazać go Claude?	Analiza wyniku VGE po wykonaniu: allow, frame, quarantine albo block.

#	PYTANIE	JAK JEST ODPOWIEDZIANE
3	Jeśli zablokowane, co powinno się stać?	Użytkownik wybiera: block, allow once albo allow for session.

Guard jest celowo konserwatywny. Domyślne podejście najpierw chroni bezpieczeństwo i audytowalność. Gdy wymagany jest wybór, użytkownik otrzymuje czytelny monit z ponumerowanymi opcjami.

CO CHRONI GUARD

POWIERZCHNIA	CO TO OZNACZA	DLACZEGO MA ZNACZENIE
Tekst promptu	Tekst wysyłany do Claude Code.	Tekst promptu może zawierać złośliwe instrukcje, skopiowaną treść z internetu albo dane, których polityka nie pozwala wprowadzić do kontekstu modelu.
Załączniki promptu	Pliki dołączane do promptu.	Załączniki mogą zawierać ukryte instrukcje albo wrażliwe dane.
Wykonanie narzędzia	Czy narzędzie może zostać uruchomione przed wykonaniem.	Niektóre narzędzia mutują pliki, pobierają URL-e albo uruchamiają komendy.
Wynik narzędzia	Treść generowana przez narzędzie po uruchomieniu.	Wynik narzędzia może zawierać prompt injection, sekrety albo niezaufane instrukcje.
URL-e	URL-e znalezione w wejściach narzędzi.	Niektóre cele URL nigdy nie powinny być pobierane, np. endpointy metadanych chmury.
Ścieżki poświadczeń	Znane lokalne pliki poświadczeń i wzorce.	Claude Code nie powinien ich przypadkowo odczytywać, edytować ani zapisywać.
Wynik subagenta	Wynik narzędzia generowany w imieniu subagenta.	Subagenci mogą zebrać ryzykowny wynik tak samo jak sesja nadrzędna.

CZEGO GUARD NIE ROBI

Guard nie zastępuje normalnej kontroli wersji, code review ani higieny sekretów. Nie gwarantuje, że każda zła komenda zostanie wykryta przed wykonaniem. Lokalne bramki narzędzi oraz ochrona URL-i/poświadczeń działają przed wykonaniem, ale klasyfikacja treści często odbywa się po wygenerowaniu wyniku przez narzędzie. Na przykład pobieranie strony może się wykonać, a potem pobrany wynik może zostać poddany kwarantannie, zanim Claude będzie mógł go użyć. Nie usuwa też potrzeby ludzkiego osądu. Gdy guard prosi Cię o decyzję, traktuj monit jak rzeczywistą decyzję bezpieczeństwa.

CODZIENNY PRZEPŁYW PRACY

Normalna chroniona sesja wygląda tak:

» Uruchamiasz Claude Code w projekcie.
» Claude Code wywołuje hooki zarejestrowane przez `vge-cc-guard`.
» Guard uruchamia lokalnego daemona albo kontaktuje się z nim.
» Wywołania narzędzi są dopuszczane, odrzucane, kierowane do natywnego monitu zatwierdzenia Claude Code albo sprawdzane po wykonaniu, zależnie od polityki.
» Decyzje VGE i lokalne decyzje guardrail są zapisywane w prywatnym lokalnym logu audytowym.
» Jeśli treść jest zablokowana, ale możliwa do odzyskania, wybierasz `block`, `allow once` albo `allow for session`.

W czystej sesji możesz nie zauważyć guarda poza okazjonalnymi pytaniami o zatwierdzenie. W ryzykownej sesji możesz zobaczyć komunikat blokady albo monit decyzyjny.

INSTALACJA

Zainstaluj pakiet globalnie (publikowany w scope npm `@vigil-guard`):

```
npm install -g @vigil-guard/vge-cc-guard

npm install -g @vigil-guard/vge-cc-guard@beta
```

Zainstaluj hooki dla swojego użytkownika:

```
vge-cc-guard install --apply --scope=user
```

Zainstaluj hooki tylko dla bieżącego projektu:

```
vge-cc-guard install --apply --scope=project
```

Podejrzyj zmiany bez zapisywania:

```
vge-cc-guard install --dry-run --scope=user

vge-cc-guard install --dry-run --scope=project
```

Wybór zakresu użytkownika albo projektu

ZAKRES	NAJLEPSZY DLA	PLIK USTAWIEŃ
User	Pojedynczy deweloper, który chce guarda we wszystkich projektach Claude Code.	~/.claude/settings.json
Project	Repozytorium, w którym zespół chce projektowe ustawienia hooków.	/.claude/settings.json

Zakres użytkownika jest wygodny. Zakres projektu jest łatwiejszy do uzasadnienia, gdy repozytorium ma własną politykę Claude Code. Po instalacji zrestartuj otwarte sesje Claude Code, aby pobrały nowe ustawienia hooków.

Co tworzy instalacja

Instalator rejestruje hooki Claude Code i tworzy lokalny stan w `~/.vge-cc-guard/`:

ŚCIEŻKA	CEL
~/.vge-cc-guard/config.json	Główna konfiguracja guarda.
~/.vge-cc-guard/audit.log	Lokalny log audytowy JSONL.

ŚCIEŻKA	CEL
~/.vge-cc-guard/debug.log	Lokalny log diagnostyczny.
~/.vge-cc-guard/cc-settings-backups/	Kopie zapasowe ustawień Claude Code.
~/.vge-cc-guard/install-records.json	Śledzi zainstalowane zakresy.

Klucze API są przechowywane w pliku konfiguracyjnym. Utrzymuj ten plik jako prywatny.

PIERWSZA KONFIGURACJA

Otwórz konfigurator:

```
vge-cc-guard config
```

Zacznij od API Keys & VGE Connection. Potrzebujesz:

» URL API VGE.
» Klucz API wejścia.
» Opcjonalny klucz API wyjścia.
» Opcjonalny identyfikator klienta.

Użyj Test Connection przed zapisaniem. Udany test aktualizuje `verified_at` razem z niewrażliwymi fingerprintami endpointu i kluczy. Surowe klucze API nigdy nie są zapisywane w metadanych weryfikacji.

KONFIGURATOR TUI

Konfigurator jest terminalowym UI. Typowe klawisze:

KLAWISZ	AKCJA
Up / Down	Przesuń wybór.
Enter	Otwórz, przełącz, potwierdź albo edytuj.
Tab / Shift-Tab	Przejdź między polami tam, gdzie jest to obsługiwane.
Space	Przełącz kompaktową opcję tam, gdzie jest to obsługiwane.
s	Zapisz na edytowalnych ekranach polityki.
r	Przywróć domyślne tam, gdzie jest to obsługiwane.
Esc	Wróć. Jeśli są niezapisane zmiany, naciśnij ponownie, aby je odrzucić.
Ctrl-C	Wyjdź natychmiast.

Główne ekrany:

EKRAN	CEL
API Keys & VGE Connection	Konfiguracja endpointu VGE i poświadczeń.
Tools Policy	Decyzja, które narzędzia mogą działać i które wyniki są analizowane.
Native CC Permissions	Edycja własnych reguł uprawnień allow/ask/deny Claude Code.
IDE Compatibility	Dostrojenie egzekwowania dla terminala w porównaniu z natywnymi panelami IDE.
Security Baseline	Konfiguracja ochrony ścieżek poświadczeń i bazowych reguł URL.
View Current Configuration	Przegląd i eksport zredagowanego podsumowania konfiguracji.
Live Events	Obserwacja zdarzeń hooków w czasie rzeczywistym.
Decision History	Przegląd ostatnich decyzji blokujących.
Audit Log	Przegląd lokalnych zdarzeń audytowych.
Stats	Podgląd liczników decyzji i stanu zdrowia.

API KEYS & VGE CONNECTION

Ten ekran odpowiada: "Gdzie jest VGE i którego klucza guard powinien używać?"

POLE	CO ROBI	CO WYBRAĆ
VGE API URL	Bazowy URL dla żądań VGE.	Użyj URL-a VGE swojej organizacji. HTTPS jest wymagany do normalnego użycia.
Client ID source	`Auto`, `Manual` albo `Disabled` — kontroluje, co, jeśli cokolwiek, jest wysyłane do VGE jako `metadata.clientId`.	Auto wysyła nazwę użytkownika systemu operacyjnego. Manual wysyła etykietę wpisaną przez Ciebie. Disabled całkowicie pomija pole.
Input API Key	Klucz dla tekstu promptu, załączników i sprawdzeń po stronie wejścia.	Wymagany. Użyj klucza ograniczonego do tej instalacji.
Output API Key	Opcjonalny klucz do skanów wyników narzędzi.	Zostaw puste, chyba że organizacja rozdziela klucze skanowania wejścia i wyjścia.

POLE	CO ROBI	CO WYBRAĆ
Test Connection	Sprawdza, czy URL i klucz wejściowy działają.	Uruchom przed zapisem, szczególnie przy pierwszej konfiguracji.

Jeśli `api_key_output` jest puste, guard ponownie używa `api_key_input` do skanów wyjściowych.

Jak działa Test Connection

Test Connection przechodzi przez lokalnego daemona, a nie bezpośrednio z procesu TUI. Ścieżka wygląda tak: TUI -> local daemon -> VGE. To ta sama ścieżka, którą przechodzi rzeczywisty ruch hooków, więc udane Test Connection dowodzi, że daemon może połączyć się z VGE przy użyciu wpisanych kluczy.

Tryby źródła Client ID

TRYB	CO JEST WYSYŁANE DO VGE	KIEDY UŻYWAĆ
`Auto` (domyślnie)	Nazwa użytkownika systemu operacyjnego, wykryta raz na proces daemona z `os.userInfo()`.	Chcesz przypisywać zdarzenia do dewelopera bez ręcznej konfiguracji.
`Manual`	Dosłowna wartość wpisana w polu Client ID.	Potrzebujesz stałej etykiety, takiej jak `ci-runner-7` albo `team-platform`.
`Disabled`	`metadata.clientId` jest całkowicie pomijane w payloadach VGE.	Jawnie nie chcesz ujawniać użytkownika systemu operacyjnego ani żadnego innego identyfikatora.

Uwaga prywatności. Tryb Auto wysyła nazwę użytkownika systemu operacyjnego (pośrednie PII) do VGE przy każdym wywołaniu analyze. Jeśli wdrożenie VGE przekracza granicę organizacyjną albo jurysdykcyjną, preferuj Manual z etykietą nieosobową albo Disabled.

TOOLS POLICY

Ten ekran odpowiada na dwa różne pytania dla każdego narzędzia:

» Czy narzędzie może działać, zanim istnieje jakikolwiek wynik?
» Jeśli działa, czy wynik powinien zostać przeanalizowany, zanim użyje go Claude?

POLE	WARTOŚCI	ZNACZENIE
gate	allow, ask, block	Kontroluje, czy narzędzie może zostać uruchomione przed wykonaniem.
analyze_output	on, off	Kontroluje, czy wynik tego narzędzia jest wysyłany do VGE i może zostać obramowany, poddany kwarantannie albo zablokowany.

Wartości bramki

BRAMKA	DOŚWIADCZENIE UŻYTKOWNIKA	UŻYJ, GDY
allow	Narzędzie może działać bez pytania przed wykonaniem.	Narzędzie jest powszechne, a Ty polegasz na skanowaniu wyniku albo innych guardrails.
ask	Claude Code pokazuje natywny monit zatwierdzenia przed uruchomieniem.	Chcesz ludzkiej kontroli przed wykonaniem, ale nie chcesz twardej odmowy.
block	Narzędzie zostaje odrzucone przed uruchomieniem.	Narzędzie jest zbyt ryzykowne dla tego profilu albo powinno być włączane tylko tymczasowo.

Widoczność kwarantanny subagentów

Tekst zwracany przez subagenta nie jest domyślnie skanowany przez politykę wyniku Task. Guard śledzi narzędzia uruchamiane przez subagenta. Gdy wynik narzędzia należący do subagenta trafia do kwarantanny, nadrzędny wynik Agent otrzymuje ustrukturyzowane powiadomienie:

```
[VGE_SUBAGENT_QUARANTINE_NOTICE]

notice_format_version: 1

session: <session_id>

agent_id: <agent_id>

agent_call: ac_<id>

total_quarantined: <count>

counts_by_status: pending=<n> resolved_block=<n>

entries:

- decision=dec_<id> tool=<tool> resource="..." reason=<code>

operational_note: <guidance>

[/VGE_SUBAGENT_QUARANTINE_NOTICE]
```

Powiadomienie jest informacyjne dla Claude. Mówi Claude, że subagent pracował na bezpiecznych placeholderach i że odpowiedź końcowa może być niekompletna. Nie uwalnia wyniku z kwarantanny. Nigdy nie zawiera surowego zablokowanego wyniku narzędzia ani wyników VGE.

DOMYŚLNA POLITYKA NARZĘDZI

NARZĘDZIE	DOMYŚLNA BRAMKA	ANALIZA WYNIKU	DLACZEGO
Bash	allow	on	Wynik powłoki ma dużą wartość i wysokie ryzyko. Wynik jest sprawdzany.
Read	allow	on	Odczyty plików są powszechne i mogą ujawniać wrażliwą treść.
Grep	allow	on	Wynik wyszukiwania może ujawniać ryzykowne fragmenty tekstu.
Glob	allow	off	Listy plików mają mniej treści i często są szumem.
WebSearch	allow	on	Wyniki z internetu są niezaufane i powinny zostać obramowane albo zablokowane.
WebFetch	allow	on	Pobrane strony mogą zawierać prompt injection albo niebezpieczne instrukcje.
Write	block	off	Zapisy mutują pliki, więc domyślne ustawienie jest konserwatywne.
Edit	block	off	Edycje mutują pliki, więc domyślne ustawienie jest konserwatywne.
Task	allow	off	Bazowy wynik należący do subagenta jest obsługiwany osobno.
*	ask	off	Nieznane narzędzia domyślnie wymagają natywnego zatwierdzenia.

NATIVE CC PERMISSIONS

Ten ekran edytuje własne tablice uprawnień Claude Code:

TABLICA	ZNACZENIE
permissions.allow	Claude Code może uruchamiać pasujące komendy bez monitu.
permissions.ask	Claude Code pyta przed uruchomieniem pasujących komend.
permissions.deny	Claude Code blokuje pasujące komendy przed uruchomieniem hooków.

To są uprawnienia Claude Code, a nie decyzje VGE. Są przydatne dla prostych wzorców komend, takich jak reguły odmowy dla powłoki.

IDE COMPATIBILITY

Terminalowy Claude Code daje najczytelniejsze doświadczenie dla decyzji blokujących. Natywne panele IDE nie zawsze pokazują konwersacyjne monity decyzyjne w ten sam sposób. Ustawienia kompatybilności pozwalają zmniejszyć ukryte tarcie bez wyłączania lokalnych zabezpieczeń.

USTAWIENIE	DOMYŚLNIE	CO KONTROLUJE
Prompt text	enforce	Czy niebezpieczny tekst promptu może blokować albo pytać.
Prompt file attachments	enforce	Czy niebezpieczne załączniki promptu mogą blokować albo pytać.
Subagent tool outputs	enforce	Czy wyniki narzędzi należące do subagentów uczestniczą w egzekwowaniu.
Prompt text VGE failures	fail_closed	Co się dzieje, jeśli skanowanie tekstu promptu nie może się zakończyć.
PostTool output failures	fail_closed	Co się dzieje, jeśli skanowanie wyniku narzędzia nie może się zakończyć.
PostTool overload	enabled	Krótkie cooldowny dla kwalifikujących się fail-open błędów web research.

enforce oznacza, że guard może zatrzymać sesję albo poprosić użytkownika o decyzję. off oznacza, że dany etap nie blokuje i nie pyta. fail_closed blokuje, gdy skan nie może się zakończyć. fail_open pozwala kontynuować pracę, gdy VGE jest niedostępne. Zmiana na fail_open wymaga drugiego potwierdzenia w konfiguratorze.

SECURITY BASELINE

Ten ekran obsługuje lokalne zabezpieczenia, które nie zależą od VGE.

Ochrona ścieżek poświadczeń

Domyślnie włączona. Blokuje Read, Edit i Write dla znanych ścieżek poświadczeń.

Przykłady chronionych ścieżek:

» `.env`, `*.env`
» `~/.ssh/*`
» `~/.aws/credentials`, `~/.aws/config`
» `~/.kube/config`
» `~/.config/gcloud/*`
» Klucze prywatne: `id_rsa*`, `id_ed25519*`
» Nazwy plików zawierające `credentials` albo `secrets`

URL Access Baseline

Blokuje ryzykowne cele URL przed uruchomieniem wywołań narzędzi podobnych do fetch.

PRESET	DOMYŚLNIE	CO BLOKUJE
cloud_metadata	on	Endpointy metadanych chmury i powiązane adresy.
unsafe_url_shapes	on	Formy URL, które są niebezpieczne albo niejednoznaczne.
oob_callback_collectors	off	Znane usługi callback używane we wzorcach eksfiltracji.
strict_internal_network	off	Cele sieci wewnętrznej/prywatnej.
public_paste_and_file_drops	off	Publiczne usługi paste i file-drop.

CLAUDE CODE CONTRACT HEALTH

Zastępowanie wyniku PostTool przez guarda zależy od tego, czy działający plik binarny Claude Code akceptuje określony kształt zastąpienia wyniku. Ta zależność jest śledzona jako live contract między guardem a zainstalowanym Claude Code. Zwykle nie musisz o tym myśleć. Konfigurator i `doctor` pokazują to, gdy coś się zmieni.

Kiedy kontrakt ulega degradacji

Typowe przyczyny:

» Claude Code został zaktualizowany.
» Plik binarny `claude` został przeniesiony pod nową ścieżkę.
» SHA-256 pliku binarnego się zmienił.
» Poprzedni wynik live probe został utracony albo nigdy się nie uruchomił.

Gdy kontrakt jest zdegradowany, zastępowanie wyniku L0 jest wyłączone. Blokowanie PostTool nadal działa przez bezpieczniejszą ścieżkę zdegradowaną: nadal otrzymujesz monity decyzyjne, a pierwotny zablokowany wynik nie jest uwalniany.

```
vge-cc-guard doctor --cc-contract

vge-cc-guard doctor --cc-contract --assume-live-pass
```

VIEW CURRENT CONFIGURATION

Użyj tego ekranu, gdy chcesz zrozumieć efektywną konfigurację bez edytowania.

Pokazuje: ścieżkę konfiguracji, URL VGE i zamaskowane klucze, identyfikator klienta, bramki narzędzi, ustawienia analizy wyników, egzekwowanie promptów i załączników, tryby awarii, status URL baseline.

> Akcja eksportu zapisuje zredagowane podsumowanie odpowiednie do zgłoszeń wsparcia. Klucze API są maskowane.

LIVE EVENTS, DECISION HISTORY, AUDIT I STATS

Live Events

Pokazuje aktywność hooków w czasie rzeczywistym. Użyj, gdy Claude Code wydaje się zatrzymany albo gdy chcesz zobaczyć, czy narzędzie jest dopuszczane, odrzucane czy sprawdzane.

Decision History

Pokazuje ostatnie decyzje blokujące. Szukaj decyzji pending, resolved_block, resolved_allow_once, resolved_allow_session.

Audit Log

Lokalny JSONL rejestrujący zdarzenia istotne dla bezpieczeństwa bez przechowywania surowego zablokowanego wyniku narzędzia. Zapisuje: co zostało zablokowane, które narzędzie wygenerowało zdarzenie, który decision ID był zaangażowany, jaki był wybór użytkownika.

Stats

Szybkie podsumowanie operacyjne: liczniki decyzji, liczniki sygnałów VGE, wskaźniki zdrowia i aktywne sesje.

KOMENDY

vge-cc-guard install

```
vge-cc-guard install --dry-run --scope=user

vge-cc-guard install --apply --scope=user

vge-cc-guard install --dry-run --scope=project

vge-cc-guard install --apply --scope=project
```

vge-cc-guard config / daemon / doctor

```
vge-cc-guard config

vge-cc-guard daemon

vge-cc-guard daemon status

vge-cc-guard daemon stop

vge-cc-guard doctor

vge-cc-guard doctor --no-vge

vge-cc-guard doctor --cc-contract
```

vge-cc-guard reset-session / uninstall

```
vge-cc-guard reset-session

vge-cc-guard uninstall --yes --scope=user

vge-cc-guard uninstall --yes --scope=project

vge-cc-guard uninstall --yes --scope=user --restore
```

FLAGA doctor	KIEDY UŻYWAĆ
(brak flagi)	Domyślnie. Raportuje stan lokalny i sprawdzenie łączności VGE.
--no-vge	Pomiń live round-trip VGE, gdy chcesz tylko szybkie sprawdzenie lokalne.
--cc-contract	Wypisz szczegółowy blok statusu kontraktu Claude Code.
--assume-live-pass	Ręczne nadpisanie zaufania. Używaj tylko przy osobnej weryfikacji.

DECYZJE BLOKUJĄCE

Najważniejszą interakcją użytkownika jest monit decyzji blokującej.

WYBÓR	ZNACZENIE	KIEDY UŻYWAĆ
1 / block	Utrzymaj element poza kontekstem modelu.	Użyj, gdy nie ufasz treści albo jej nie potrzebujesz.
2 / allow once	Dopuść ten dokładny element jeden raz.	Użyj, gdy sytuacja została sprawdzona i chcesz kontynuować jednorazowo.
3 / allow for session	Dopuść ten dokładny element do końca sesji.	Użyj, gdy ten sam dokładny element będzie potrzebny wielokrotnie w tej sesji.

Akceptowane odpowiedzi:

```
1

2

3

block

allow once

allow for session
```

Lub dokładne komendy:

```
vge block dec_<id>

vge allow dec_<id>

vge allow-session dec_<id>

vge allow dec_<id> continue with the task
```

Jeśli decyzja jest aktywna, a Twoja odpowiedź nie jest poprawną decyzją, guard ponownie pokaże monit. Ta odpowiedź nie zostanie wysłana do Claude jako treść zadania. To celowe. Zapobiega przypadkowej interpretacji tekstu zadania, gdy sesja jest wstrzymana na wybór bezpieczeństwa.

FORMAT KOMUNIKATÓW BLOKADY

Natychmiastowe blokady polityki

```
VGE Agent Guard: <Tool> is blocked by policy.

VGE Agent Guard: <path> is on the credential protection deny list.

VGE Agent Guard: URL denied by local URL access baseline (<reason>).
```

Format monitu decyzyjnego

```
VGE Agent Guard requires your decision.

Stage: <PreTool URL | PostTool output | Prompt input>

Decision ID: dec_<id>

Tool: <tool>

Resource: <sanitized resource label>

Reason: <block reason>

VGE: score=<score> | ruleAction=<action> | decision=<decision>

Reply with one of these choices:

1 = block

2 = allow once

3 = allow for session
```

Zredagowany placeholder wyniku narzędzia

```
[VGE SECURITY REDACTION]

Tool: <tool>

Resource: <resource id>

Router outcome: <outcome>

VGE decision: <decision>

Score: <score>

Original tool output was removed.

[/VGE SECURITY REDACTION]
```

Powiadomienie o kwarantannie subagenta

Gdy podrzędny subagent miał wyniki narzędzi w kwarantannie, nadrzędny wynik Agent może zawierać marker `[VGE_SUBAGENT_QUARANTINE_NOTICE]`. Marker oznacza, że odpowiedź subagenta może być niekompletna. Rozstrzygnij każdą oczekującą decyzję i ponów wywołanie Agent, jeśli odpowiedź potrzebuje materiału z kwarantanny.

STAN SESJI

TYP DECYZJI	CZAS ŻYCIA
allow once	Jedna dokładna ponowna próba albo wywołanie hooka.
allow for session	Do SessionEnd albo resetu sesji.
block	Blokada dokładnego zasobu na czas sesji.

```
vge-cc-guard reset-session
```

REFERENCJA KONFIGURACJI

Obsługiwana ścieżka edycji to `vge-cc-guard config`. Plik konfiguracyjny: `~/.vge-cc-guard/config.json`

POLE	ZNACZENIE
version	Wersja schematu konfiguracji. Bieżąca wartość to 1.0.0.
vge	Endpoint VGE, klucze API, etykieta klienta i budżety czasowe skanowania.
tools	Bramki wykonania per narzędzie i przełączniki analizy wyniku.
policy	Polityka bezpieczeństwa runtime i lokalne guardrails.

Pola vge

POLE	TYP	DOMYŚLNIE	DLACZEGO ISTNIEJE
api_url	URL	https://api.vigilguard	Mówi guardowi, gdzie wysyłać sprawdzenia VGE.
api_key_input	string	(puste)	Uwierzytelnia prompt, załącznik i sprawdzenia po stronie wejścia.
api_key_output	string	null	Opcjonalny osobny klucz dla sprawdzeń wyników narzędzi.
posttool_budget_ms	integer	120000	Maks. budżet czasu na skanowanie wyników narzędzi.
pretool_url_budget_ms	integer	120000	Przestarzałe pole kompatybilności.
pretool_url_total_deadline_ms	integer	120000	Przestarzałe pole kompatybilności.
userprompt_text_budget_ms	integer	120000	Maks. budżet czasu na skanowanie tekstu promptu.

Pola policy

POLE	DOMYŚLNIE	DLACZEGO ISTNIEJE
credential_protection	true	Blokuje znane ścieżki poświadczeń niezależnie od polityki narzędzi.
prompt_text_analysis	enforce	Kontroluje, czy tekst promptu może wyzwalać blokady albo decyzje.
prompt_attachment_analysis	enforce	Kontroluje, czy załączniki mogą wyzwalać blokady albo decyzje.
subagent_output_analysis	enforce	Kontroluje egzekwowanie dla wyników narzędzi subagentów.
posttool_research_output	quarantine_continue	Kontroluje jawne blokady VGE dla wyników WebFetch/WebSearch.
vge_failure_mode	fail_closed	Definiuje, co się dzieje, gdy skany VGE nie mogą się zakończyć.
session_idle_ttl_hours	24	Kontroluje, jak długo bezczynne pliki sesji pozostają na dysku.

ROZWIĄZYWANIE PROBLEMÓW

Claude Code nie wygląda na chroniony

```
vge-cc-guard doctor
```

Sprawdź: czy install --apply uruchomiono dla właściwego zakresu, czy Claude Code został zrestartowany po instalacji, czy settings.json zawiera wpisy vge-cc-guard hook.

Połączenie z VGE nie działa

Otwórz `vge-cc-guard config` i uruchom Test Connection. Sprawdź URL, klucz API, sieć/proxy.

Claude Code czeka na decyzję

Szukaj: `Decision ID: dec_`. Odpowiedz 1, 2, 3 albo komendą `vge ... dec_`. Jeśli utknięte:

```
vge-cc-guard reset-session
```

Zastąpienie PostTool jest odrzucone albo L0 wyłączone

```
vge-cc-guard doctor --cc-contract
```

Auto-probe samodzielnie odzyska kontrakt w większości przypadków. Blokowanie PostTool nadal działa przez ścieżkę zdegradowaną.

Web fetches są blokowane

Sprawdź Security Baseline: URL Access Baseline, presety, niestandardowe reguły deny. Jeśli URL jest lokalnie dozwolony, ale pobrana treść jest blokowana, to decyzja dotycząca treści, a nie celu URL.

Panele IDE ukrywają decyzje

Użyj terminalowego Claude Code dla pełnego doświadczenia. Jeśli zespół musi używać natywnego panelu IDE, przejrzyj IDE Compatibility i rozważ wyłączenie dotkniętego etapu egzekwowania.

BEZPIECZNE ZALECENIA OPERACYJNE

Zachowaj te ustawienia domyślne, chyba że masz konkretny powód:

» `credential_protection: true`
» analiza tekstu promptu: `enforce`
» analiza załączników promptu: `enforce`
» analiza wyników subagentów: `enforce`
» tryby awarii VGE: `fail_closed`
» URL Access Baseline: włączone
» analiza wyników `Read`, `Bash`, `WebSearch`, `WebFetch`: on

Rozluźniaj politykę stopniowo. Preferuj zmianę jednego ustawienia, przetestowanie workflow i przejrzenie zdarzeń audytowych przed kolejnymi zmianami.