無償のツールと AI で、脆弱性管理をここまで回す（判定の台帳と、組む手順）

当社は5つの工程に分けて、工程ごとに機械が担う所と人が担う所を決めました。 この1枚は、機械に任せた所を組む手順と、人が決める所で何を残すかの台帳です。

拠り所は、11月5日の国の注意喚起、NIST の手引、米国 CISA の判定の考え方、国内の評価制度の期限の4つです。**どれも公開されている物差しで、当社が作った基準ではありません。**

機械は見つけるところまでです。どうするかは人が決めます。この台帳は、その決めたことを置く場所です。**判定に使える時間を増やすものではありません。**

使う順（最初に開くシートから）

11シートありますが、下の順に4つを開けば進みます。 残りの7シートは、必要になったときに引きます。

貼る先は、コード支援の AI（Claude Code など）です。 手元のコードを読み書きできる所で使ってください。

#	いつ	開くシート	そこですること
1	はじめに	**組む手順**	7つの手順を読んで、全体の形を掴む。この時点では作りません
2	最初の作業	**AI に組ませる（A）**	一覧の見本を写して、自分の構成で「見るべき物」を人が書く。ここだけは AI に任せません
3	手順2から6	**AI に組ませる（B）**	「AI に頼む一文」をそのまま貼る。出来たら「人が確かめる一文」で、わざと赤を出して確かめる
4	1件目が出たら	**AI に渡す指示** → **判定の台帳**	指示の骨組みに、手順2〜5で出た検査の出力と依存の一覧を添えて渡し、1〜9列の下書きを作る。台帳に写し、10〜12列（対応の種類・決めた人・決めた日）は人が書く

迷ったときに引くシート＝優先度と判定の要素（何を見て決めるか）、期限の物差し（いつまでにするか）、記入の例（書き方）。

読む物＝検査の作り方（なぜその形か）、頻度と30日90日（どのくらいで回すか）、AI の任せ方（どこまで任せるか）。

対象は、当社の構成に限りません

一般的なクラウド基盤（例＝AWS・GCP・Azure）の上で動くウェブアプリ・API・データベース・ライブラリでも、同じ5つの工程と同じ台帳で回せます。

置き換えるのは2か所だけです。 手順2の監査機能（使っている言語の物に）と、手順6で叩く先（自分の公開している URL に）。**それ以外は、そのまま使えます。**

台帳の列

#	列	書くこと	誰が埋めるか
1	識別子	CVE 番号など。無ければ自社の連番	機械
2	見つけた日	検知の日付	機械
3	どこで見つけたか	依存関係の検査／静的な検査／外からの受け身の検査／脅威情報の取り込み／外部からの連絡	機械
4	外から届くか	インターネットから直接届く／社内からだけ／届かない	人（機械の下書きを確かめる）
5	悪用されているか	公的な一覧に載っている／報告がある／無い	機械
6	攻撃を自動化できるか	できる／できない／分からない	人（機械の下書きを確かめる）
7	影響の範囲	全部取られる／一部だけ／限定的	人
8	優先度	下の「優先度の4つ」から1つ	人
9	期限	日付。数え始めの日も書く	人（機械が候補を出す）
10	対応の種類	直す／つなぐ／受け入れる	人

#	列	書くこと	誰が埋めるか
11	決めた人	役割名。個人名でなくてよい	人
12	決めた日	日付	人
13	閉じた日・見直す日	直したら閉じた日。受け入れたなら次に見直す日と、その理由	人

列は13です。4から7の4つが、優先度を決める材料です。10から12の3つが、この台帳の芯です。

11と12を空欄のまま運用すると、**半年後に「なぜこれを受け入れたのか」が誰にも分からなくなります。** 直した件は記録が無くても困りませんが、**受け入れた件は記録が無いと困ります。**

優先度の4つ

優先度	意味	次にすること
すぐ動く	外から届き、悪用されている	3営業日以内に緩和（経路を閉じる・止める・分ける）。14日以内に修正
順番に入れる	外から届くが、悪用の報告が無い。または悪用されているが外から届かない	期限を決めて、通常の作業の列に入れる
記録して見直す	いまは当たらないが、構成が変われば当たる	台帳に残し、構成を変える前にもう一度見る
記録だけ	当たらない。使っていない機能	台帳に残す。作業は起こさない

この4つは、米国の CISA が公開している SSVC（2022年11月10日）の結論の4つと同じ並びにしています。訳と言い換えは当社のものです。

日数の正本は、下の「自社の期限を、この表から決める（3つの区分）」です。 優先度と区分は、同じ条件で同じ日数になるように揃えてあります。

判定の4つの要素

要素	問い	なぜこれを見るか	どこから取るか（無償）	手順の何番目で埋まるか
外から届くか	インターネットから直接たどり着けるか	届かない物は、同じ深刻度でも急がない	自社の一覧。本番に出る依存か、公開している面か	手順6（公開している面を叩く）と、手順2の本番だけの再実行
悪用されているか	実際に使われた報告があるか	深刻度の数字より、使われている事実が先に来る	悪用が確認された脆弱性の一覧（米国 CISA の KEV・公開 JSON）	手順5
攻撃を自動化できるか	手作業が要るか、まとめて撃てるか	自動化できる物は、広く・速く来る	悪用される確率の推定（EPSS）を目安に。**当社はまだ取り込んでいません。いまは人が勧告の本文を読んで判断**	自動では埋まりません（人）
影響の範囲	全部取られるか、一部で止まるか	一部で止まる物は、つなぎの策が効く	勧告の CVSS（OSV の公開 API）と、自社の資産の重要度	手順4（CVSS を足す）＋人が重要度を当てる

深刻度の数字（CVSS の基本値）だけで順番を決めません。基本値は、自社の事情を含んでいません。CVSS を作っている FIRST 自身が、基本値を脅威と環境の値で補うべき（should enrich）と書いています（CVSS v4.0 仕様書・2023年11月1日）。

米国 CISA の BOD 26-04（2026年6月10日）も、この4つの要素で区分を決めています。**ただしこの指令は米連邦の文民行政機関だけを拘束するもので、日本の企業の義務ではありません。**

期限の物差しの比較表

物差しを6つ並べます。どれも、その対象のために作られた物差しです。日本の企業一般の義務ではありません。自社の期限は、この表を見ながら自社で決めてください。

物差し	対象	期限	延長・例外	出典の日付
CISA BOD 26-04	米連邦の文民行政機関（契約で定めれば委託先も）	4つの要素で区分を決める。最短3日（修正か緩和に加えて、侵害の有無の調査も）。最も低い区分は「次の大きな更新のとき」	CISA は免除や例外を出さないと明記。外から届かない状態にすれば区分が下がる	2026-06-10（手引 2026-08-25）
CISA BOD 22-01	同上	公的な一覧に載った物は2週間（古い物は6か月）	廃止済み。BOD 26-04 に置き換わった	2021-11-03
SCS 評価基準 4-4-4-2（★3・★4）	任意の評価制度を申請する事業者	**アップデートプログラムがリリースされてから14日以内**。対象は、ベンダーが「重大」または「高リスク」と説明する脆弱性を直す物、深刻度の基本値が7.0以上の物、レベルの詳細がベンダーから提供されていない物	期限内に当てられない場合は、機能の無効化・ベンダー推奨の回避策・ネットワークからの分離・不正な通信の遮断のいずれかで リスクを下げる。基本値7.0以上でも、情報公開の時点で攻撃が観測されていない（蓋然性が低い）物は、対応計画と代替策を前提に期限の延長を認める。**公的機関等の一覧に載っていて攻撃が観測されている物は、引き続き14日**	IPA・要求事項・評価基準 Ver.1.1（2026-10-28）
重要インフラ統一基準 3.3.6③	重要インフラ事業者が安全基準に盛り込む事項	日数は決めない。修正を当てる作業方針を確立すること	リスクに応じた措置	2026-07-31 決定・2026-10-01 施行
NIST SP 800-40 Rev.4	一般向けの手引	日数を決めない。資産のまとまりごとに自分で期限を決める	例外は追跡し、定期的に見直す	2022-04
国の注意喚起	重要インフラ事業者等	「速やかに」（日数は無い）	事業継続を含む総合的な判断。手順を事前に作ること	2026-05-18

共通しているのは、**期限が1つの数字ではない**ことです。どれも「外にどれだけ出ているか」と「実際に悪用されているか」で変え、状況が変わったら見直す形になっています。

数え始めの日も決めておいてください。BOD 26-04 は、公的な一覧に追加された日か、自分で検知して管理の仕組みに載せた日の、**早い方**から数えると書いています。

自社の期限を、この表から決める（3つの区分）

表を見ても、そのまま使える数字は多くありません。 6つのうち日数が入っているのは3つで、残りは「速やかに」「自分で期限を決める」「作業方針を確立する」です。

だから、借りる数字を1つに絞らず、3つの区分に組みます。

区分	当てはまる条件	期限
1	**悪用が確認されている × 外から届く**	**3営業日で緩和、14日で修正**
2	**修正版がある × 外から届く**	**修正版が出てから14日。** 無理なら、代替策と対応計画を作って延ばす
3	**外から届かない、または修正版が無い**	**次の四半期に見直す**

14日は、国内の任意の評価制度（SCS 評価基準 4-4-4-2）から借りています。 成長企業がいちばん当たりやすい制度だからです。 **区分1の「3営業日で緩和」は、米国の連邦機関向けの指令の考え方**を当てたものです。**その指令は米連邦の文民行政機関だけを拘束します。日本の企業の義務ではありません。 区分2の「無理なら代替策と対応計画」も、同じ評価制度が認めている延長の形**です。

これは当社が勝手に作った基準ではありません。 公開されている物差しの条件を、3つの区分に並べ直しただけです。**自社の事情に合わせて日数を変えて構いません。**

深刻度の数字だけでは、いま危ないかが分からない

実測で分かったこと（2026年11月・当社の依存で）。

取れた深刻度の値は、ほとんどが「基本値」だけでした。 3件のうち、**実際に攻撃されているかを示す指標が付いていたのは1件だけ**です。

基本値は、最初の一度だけ付いて、その後ほとんど動きません。 一方で「いま攻撃されているか」は毎日変わります。**だから基本値だけ見ていると、今日のリスクが分かりません。**

手当て＝悪用されている物の一覧を、別に取りに行きます。 無償で公開されています。

深刻度が高くても、外から届かず、攻撃も観測されていなければ、順番は下げられます。 深刻度が中くらいでも、すでに悪用されていれば、いちばん先に直します。

この2行が、深刻度の数字だけで決めないということの中身です。

この表で確かめていないこと

BOD 26-04 の中間の区分の日数は、確かめられていません。 CISA のページでは該当の表が画像で埋め込まれていて、本文に日数が載っていません。確かめられたのは、最短が3日であることと、最も低い区分が「次の大きな更新のとき」であることの2つだけです。**中間の区分に何日と書いてある表は、当社は見ていません。**

この表から、1つ落としました。 民間の脅威報告が「外に面した物と認証の仕組みは72時間以内」を推奨している、という記述を見かけることがあります。当社も当初この表に入れていましたが、**原本に辿り着けなかったので落としました。** 公式のページは前の年の版のままで、二次の記事のうち1本は「報告の本文で確かめられなかった」と自ら書いています。

数字を載せるかどうかは、その数字が正しそうかではなく、原本に辿れたかで決めています。 72時間という数字が間違っているという話ではありません。当社が確かめられなかった、という話です。

回す頻度

2つに分けます。左が当社が実際に回している頻度、右は増やすときの案です。 どの文書の要求でもありません。

何を	当社の実際	増やすときの案
検知	変更のたび。外に出す前	加えて、公的な一覧の更新を1日1回取り込む
判定	検知で増えた分が出たとき	曜日を決めて、まとめて見る回を作る
振り返り	四半期	月に1回
経営への報告	四半期	四半期のまま
外部の診断	年に1回から数回	変えない。役割は「仕組みが回っているかの確かめ」

当社は毎日は回していません。 変更のたび、外に出す前、四半期の3つです。右の列は「人の時間が取れるなら、ここから増やす」という案で、当社がやっていることではありません。

判定にかかる時間は書いていません。**当社の実測が無いからです。** 機械の側は1件あたり数秒ですが、人が1件を見る時間は、件数にも、その組織の決め方にもよります。

外部の診断を減らす提案ではありません。役割を変える提案です。**検知で見るのは「出てきた物を捌けているか」、年に数回の診断で見るのは「捌く仕組みそのものが働いているか」です。**

30日・90日・それ以降

やることは、30日が5つ、90日までが5つ、それ以降が4つです。合わせて14です。

最初の30日：一覧と期限を作る

#	やること
1	外に面した物の一覧を作る（製品名・版・担当者・組み込まれた部品まで）
2	脆弱性の情報をどこから受け取るかを決める
3	期限の物差しを自社で1枚に決める（例＝「悪用あり×外から届く」「悪用あり×内部だけ」「悪用なし」の3つの区分）
4	止めるか続けるかを決める人と、つなぎの策の候補を先に書く
5	自動で更新できる所は、自動更新を有効にする

90日まで：突き合わせを機械にして、記録を残す

#	やること
1	一覧と公的な一覧の突き合わせを機械にして、当たったら通知する
2	開発の流れに、依存関係の検査と静的な検査を入れる。危ない物が出たら止める
3	外からの受け身の検査を、月1回と変更の後に。能動的な検査は検証用の環境だけ
4	この台帳を作り、期限の超過が見える形にする
5	外に出ている資産の発見を、一覧の漏れの点検に使う

それ以降：AI の下書きと振り返り

#	やること
1	AI の修正案と影響の要約は、下書きまで。承認と取り込みは人が行い、試験を通す。新しい依存先は、人が実在と版を確かめる
2	AI のツールそのものも、委託先として確かめる（渡すコードやデータの範囲・保存期間・権限・止められるか）
3	月に1回振り返る。期限を守れた割合は「資産の重要度×深刻度」で見る。例外を見直す
4	AI のツールを起動する場所を決める。外から届いたファイルが混ざる場所で起動しない

4つ目は、2026年10月に海外のセキュリティ会社の報告として報じられた手口に対応するものです。悪意のあるページがダウンロードのフォルダに指示のファイルを置き、そのフォルダで起動した開発支援の AI が、それを「この案件の指示」として読んでしまう、という形です。修正版が出る種類の問題ではなく、起動する場所を分ける運用で防ぎます。

AI に任せる範囲と、任せない範囲

工程	機械に任せる	人が決める
資産の把握	外に面した物の発見。製品名と版の一覧の更新	担当者。業務上の重要度。止めてよい時間帯
検知	公的な一覧との突き合わせ。依存関係の検査	検知の対象に入れる範囲
判定	期限の候補の計算。情報の要約	自社への影響。期限の確定。例外を認めるか
修正	修正案の下書き。影響範囲の説明	本番に当てる承認。止めるか続けるか。受け入れる理由と期限
確認	修正後の再検査。期限の超過の通知	「閉じる」の判定。例外の見直し。経営への報告

英国の NCSC は2026年4月の文書で、AI のツールは問題を見つける方が、対応を確実に実行するより得意な場合がある（can be better at）と書いています。あわせて、**防御する側は AI のツールの出力をどう捌くかを考える必要がある**とも書いています。この台帳は、その「どう捌くか」を置く場所です。

AI が自動で直すようになる、とは書きません。この分野の一次の文書は、どれも人の確認を前提にしています。

記入の例

この値を推すものではありません。書き方の例です。固有名（サーバ名・委託先名・鍵の置き場所）は書かず、記号で書きます。

#	列	例
1	識別子	CVE-20XX-XXXXX
2	見つけた日	11月4日
3	どこで見つけたか	脅威情報の取り込み（公的な一覧の更新）
4	外から届くか	インターネットから直接届く
5	悪用されているか	公的な一覧に載っている
6	攻撃を自動化できるか	できる
7	影響の範囲	全部取られる
8	優先度	すぐ動く

#	列	例
9	期限	11月7日（数え始め＝11月4日。公的な一覧に載った日）
10	対応の種類	つなぐ（修正版が出ていないため、対象Aへの経路を閉じる）
11	決めた人	情報システムの責任者
12	決めた日	11月4日
13	閉じた日・見直す日	未。11月11日に見直す（修正版が出たら直してから閉じる）

受け入れる場合の例も1つ置きます。

#	列	例
1	識別子	CVE-20XX-YYYYY
4	外から届くか	届かない（検証用の環境だけ）
5	悪用されているか	報告が無い
8	優先度	記録して見直す
10	対応の種類	受け入れる
11	決めた人	情報システムの責任者
13	閉じた日・見直す日	2月4日に見直す。理由＝検証用の環境だけで使っており、本番に出す計画が無い。本番に出すことが決まった時点で、見直しを前に倒す

受け入れた件に**見直す日を書かないと、受け入れが永久になります。** 受け入れは判断で、放置は判断ではありません。この2つが台帳の上で同じ見た目になると、見分けられなくなります。

検査の作り方（7つの型）

台帳は、検知の後ろに置く物です。その前にある「検知の仕組み」をどう作るかも、同じくらい効きます。当社が自分の仕組みで使っている型を7つ出します。**どれもツールを買えば手に入るものではなく、書き方と置き場所の決め方です。**

#	型	中身
1	**見るべき物の一覧を、検査とは別に人が書いて持つ**	見るべき物の一覧を、検査とは別に人が書いて持つ。検査はその一覧を読んで回る。一覧にあるのに検査が無い物は、穴として出てくる。**検査の緑は「この検査に引っかからなかった」という意味で、それ以上ではない**
2	**既知は警告に落とし、新しく出たものだけ止める**	基準の件数をファイルに置き、そこから増えたときだけ止める。**全部で止めると赤が日常になり、赤が読まれなくなる**
3	**検査が走らなかったら「判定不能」と出す**	0件と書かない。**外とつながらない環境で「脆弱性0件」と出すのが、いちばん危ない誤りになる**
4	**手元で緑は、本番で緑の証明にならない**	手元でだけ動く書き方をすると、共用の環境では毎回落ちる。**手元の結果を本番の結果として扱わない**
5	**1つずつ見る検査と、突き合わせる検査は別物**	1つずつ見る検査が全部緑でも、2つの物の間でずれていることがある。**突き合わせる検査が無いと、ずれは最後まで出てこない**
6	**外に出す前の検査は1本にまとめる**	当日の朝に「いくつ思い出す」を人の記憶に任せない。**1つでも落ちたら出さない**
7	**AI は判定の下書きまで。承認と記録は人**	期限の候補の計算・情報の要約・影響する版の読み取り・修正案の下書きは任せる。**自社への影響・期限の確定・例外を認めるかは人が決めて、この台帳に残す**

なぜ1つ目を先頭に置くか

検査を増やすより先に、何を見ていないかを知る方が効きます。

検査が数十本あっても、ある種類の脆弱性を見る検査が1本も入っていなければ、全部緑のまま見落とします。**そのとき赤は出ません。** 出ないことが、見ていることの証明になってしまいます。

だから、見るべき物の一覧を、検査とは別に人が書いて持ちます。 一覧に載っていない物は「見ていない」と扱います。新しい検査を足すときは、まず一覧に1行足します。

無償のツールだけで組む手順（7つ）

ここまでは「何を決めるか」でした。ここからは「どう組むか」です。 当社が実際にこの順で足しました。手順を7つ並べます。

1日1つで進められます。どれも無償で、登録も要りません。最後まで行かなくても、1つ目で止めて構いません。 止めた所までは動きます。

#	やること	入れる物（無償）	出来上がる物	止まる条件
1	**見るべき物の一覧を、検査とは別に人が書いて1枚作る**	テキストファイル1枚	「いま何を見ているか」の一覧	一覧に無い物は「見ていない」と扱う
2	**依存の既知の脆弱性を見る**	パッケージ管理ツールの監査機能（`npm audit` 等）	勧告の一覧（識別子・重大度・修正版の有無）	検査が走らなければ「判定不能」。0件と書かない
3	**既知の件数を基準ファイルに置く**	テキストファイル1枚	基準より増えたときだけ止まる形	全部で止めない（赤が日常になると読まれなくなる）
4	**CVE の別名と CVSS を足す**	OSV の公開 API（`api.osv.dev`）	勧告に CVE 番号と CVSS が付いた一覧	取れなければ「判定不能」。空欄のまま進めない
5	**悪用が確認された一覧と突き合わせる**	CISA KEV の公開 JSON	「悪用されているか」の列が埋まった一覧	取れなければ「載っていない」とは書かない
6	**公開している面を1本ずつ叩く**	公開しているはずの URL の一覧＋取得ツール	開けない物・開きすぎている物の一覧	期待と違う行が1つでもあれば人が見る
7	**この台帳に1件入れて、1件閉じる**	この Excel	決めた人と決めた日が入った1行	「受け入れる」に見直す日が無ければ閉じない

この順にした理由

1日目が一覧なのは、ここが抜けると残りが全部無意味になるからです。 検査を何本足しても、見ていない種類があれば全部緑のまま見落とします。**そのとき赤は出ません。**

手順2から6で、判定の4つの要素のうち3つが埋まります。 悪用されているか（手順5）、影響の範囲（手順4の CVSS）、外から届くか（手順6）。**残る1つ（攻撃を自動化できるか）は、いまは人が見ています。**

手順6（公開している面）を後ろに置いたのは、当社がこの順で足したからです。 先に依存を見て、公開面は後から足しました。**逆の順でも構いません。**

かかった時間（当社の実測）

機械の側は1件あたり数秒です。 2026年11月6日の実測＝依存の勧告3件に CVE の別名と CVSS を足し、悪用の一覧（その日の版で1768件）と突き合わせるまでが**2.2秒**。

増えるのは人の時間だけです。だから頻度を上げるかどうかは、機械の都合ではなく、人の時間が取れるかで決まります。

まだ自動にしていないもの

4つあります。台帳の自動起票／構成の一覧（SBOM）の出力／悪用される確率（EPSS）の取り込み／公開している面の自動発見。 この4つは、やっていないのではなく、**人が手で担っています。**

有償のサービス基盤は、この4つを最初から持っています。そこが、お金で短くできる差です。

AI に渡す指示の例

前の節で「検知をどう組むか」は出しました。ここは「AI に何をどう渡すか」です。 当社が実際に使っている指示から、固有名と自社の構成を外して起こしました。

渡す物の形（3つだけ）

渡す物	中身	渡さない物
検査の出力	勧告の識別子・重大度・当たっている依存の位置	鍵・接続情報・環境変数の名前
依存の一覧	名前と版だけ	本番の構成図・入口の名前
これまでに決めたことの記録	台帳の過去の行	未修正の件の詳細

指示文の骨組み

```
あなたの役割＝判定の下書きを作る。決めるのは人。

渡す物＝検査の出力／依存の一覧／これまでに決めたことの記録。

返す形＝台帳の1列目から9列目まで。各列は、次のどれかで埋める。
  ・出力にそのまま書いてある値
  ・出力から計算できる値（計算の式も書く）
  ・分からない＝「不明」と書く（推測で埋めない）

やらないこと＝
  ・本番に当てること
  ・「受け入れる」と決めること
  ・検査の基準そのものを書き換えること
  この3つは、承認が下りても実行しない。手順と確認点を書いて止まり、人が実行するのを待つ。

数を書くときは、どこから数えたかを併記する。数えていない数は書かない。
```

返させる形（5つ）

返させる物	受け取り方
勧告の要約	原文の URL と併記。要約だけで判断しない
影響する版かどうかの読み取り	「該当／非該当／不明」の3つから1つ
台帳の1〜9列目の下書き	空欄を許す。埋まっていない列は人が埋める
修正の差分案	当てない。読むだけ
根拠の行	どの出力の何行目から言っているか

やらせない3つを、手順ではなく権限で止める

本番に当てる。「受け入れる」と決める。検査の基準を書き換える。

この3つは、**承認しても AI に実行させません。** 手順書で縛るのではなく、**実行できる権限を渡しません。** 手順書は、急いでいるときに飛ばされます。権限が無ければ、飛ばしようがありません。

「受け入れる」を人に残しているのは、**それが期限と担当者を付けない唯一の選択肢だから**です。いちばん楽な方の判断は、人が名前を書いて選ぶことにしています。

この節の限界

この指示文で、当社と同じ結果が出るとは限りません。 モデルも、渡す物の形も、組織によって違います。**骨組みとして使ってください。**

当社は、AI の出力を人が確かめずに台帳へ入れたことはありません。 そこを省くと、この台帳は「AI が書いた記録」になります。

AI に組ませるときの渡し方（補足）

前の2つの節で「何を決めるか」と「何を入れるか」は出しました。ここは「どう動かすか」です。 手順を AI（Claude Code など）に組ませるときの渡し方を書きます。

A. 手順1で作る一覧の見本

最初の1枚です。この5行で足りています。 置き場所とファイル名は何でも構いません。**ここだけは AI に頼まず、人が書きます。**

見る物	いつ見るか	どうやって見るか
依存パッケージ	変更のたび	監査機能で見る
公開している URL	変更のたびと、出す前	1本ずつ叩く
秘匿情報の混入	変更のたびと、出す前	配る物の中身を開いて語を探す
TLS と応答ヘッダ	四半期	人が見る
外から見える範囲	変更の後	人が見る（受け身の検査）

この3列のまま、表計算でもテキストでも構いません。 大事なのは置き方ではなく、**検査とは別に人が書くこと**です。

検査はこの一覧を読んで回ります。 一覧に行があるのに検査が無ければ、そこが穴です。**自分で1行足して、それを見る検査が無いことを確かめてください。**

B. 手順ごとの「AI に頼む一文」と「人が確かめる一文」

頼む一文は、そのまま貼れます。確かめる一文は、人が自分の手でやります。

手順	AI に頼む一文	人が確かめる一文
1	**AI には頼みません。人が書きます**	書いた一覧に自分で1行足して、それを見る検査が無いことを確かめる
2	「依存の監査を走らせ、勧告の識別子・重大度・修正版の有無を表にして」	**わざと古い依存を1つ入れて、赤になるか**
3	「いまの件数を基準ファイルに書き、増えたときだけ止まる形にして」	基準の数を1つ減らして、止まるか
4	「各勧告を OSV の公開 API で引き、CVE の別名と CVSS を足して」	取れなかった行が「不明」になっているか（空欄で通っていないか）
5	「CISA KEV の公開 JSON を取り、CVE で突き合わせて」	**外とつながらない状態で走らせ、「判定不能」と出るか**（0件と出たら誤り）
6	「この URL の一覧を1本ずつ叩き、期待と違う行だけ出して」	一覧に存在しない URL を1本足して、赤になるか
7	「この台帳の1〜9列目の下書きを作って。分からない列は『不明』と書いて」	10〜12列（対応の種類・決めた人・決めた日）を**人が埋める**

C. 渡さない物と、前提

渡さない物＝鍵・接続情報・環境変数の名前・本番の構成図・入口の名前。**AI に渡す指示の例の節と同じ線です。**

前提は3つあります。

ひとつ。**手順2の監査機能は、使っている言語の物に置き換えてください。** 当社は npm の監査機能を使っていますが、他の言語にも同じ役割の物があります。無ければ手順2を飛ばして手順6から始めても構いません。

ふたつ。**手元で緑は、本番で緑の証明になりません。** 検査に自分の環境の場所を直接書くと、手元だけ通ります。検査のファイル自身の位置から相対で解決する書き方にしてください。

みっつ。**AI の出力を、人が確かめずに台帳へ入れないでください。** そこを省くと、この台帳は「AI が書いた記録」になります。**確かめる役は、作った役と分けてください。**

この節の限界

当社の構成で動いた形です。 同じ指示で同じ結果が出るとは限りません。

頼む一文で作ったら、そのつど「人が確かめる一文」をやってください。 作る前には確かめられません（まだ検査が無いので、赤も出ません）。**作った直後に、わざと赤を出して確かめます。**

赤が出ない検査は、通っているのか、見ていないのかが分かりません。 そこを確かめずに次の手順へ進むと、最後まで「全部緑」のまま、何も見ていない仕組みが出来上がります。

この台帳の限界

この台帳は、判定する人の時間を増やしません。

検知を毎日に増やすなら、判定に使える時間も増やす必要があります。台帳は、増えた件数を捌くためのツールではなく、**捌いた結果を残す場所**です。残す場所が無いと、同じ判断を何度もやり直すことになります。

決める人が1人しかいない組織では、この台帳は「その1人が決めた記録」にしかなりません。それでも、半年後に理由を思い出せる価値はあります。

この資料について

合同会社SecuROI が配っています。登録は要りません。出典（合同会社SecuROI・www.securoi.com）を残し、趣旨を変えなければ、複製・転送・書き換えは自由です。

当社は検知のツールを売っていません。この台帳は、検知の後ろに置く物です。
