SECURIY DOCS — No.12 大企業との取引

取引先セキュリティチェックに答える

エンプラ顧客に提供するための3つの観点

権限・検証記録・再検証。チェックシートの設問に対して、何を用意し何を示せばよいか



Securiy

10 PAGES · 2026.10

TEMPURA TECHNOLOGIES

この資料で扱う設問、扱わない設問

大企業（エンタープライズ）との取引では、契約前にセキュリティチェックシートへの回答を求められ、診断報告書の提出を求められることもあります。設問に「はい」と答えるために何を用意し、何を示せばよいかを整理します。

想定読者

チェックシートを受け取った開発責任者・営業

セキュリティ専任がいない SaaS・Fintech・EC 事業者。初めての大企業との商談で、回答の根拠に困っている方。

扱う設問

アクセス制御・脆弱性への対応・変更時の確認

この3分野の設問は、聞かれ方が違っても「権限」「検証記録」「再検証」の3つに整理できます。

扱わない設問

暗号化、ログイン方式、バックアップ、ログ保管、障害対応、規程・教育

これらは設定画面の記録や運用規程で答える設問です。本資料は適合の証明ではなく、回答に添える根拠の作り方を扱います。

1 チェックシートが来たとき

設問の意味は分かっても、「はい」の根拠が手元にない P.03

設問を3つの観点に仕分ける P.04

2 3つの観点と、自分でやる方法

権限: 「誰が・何を・どの条件で」を1枚で示す P.05

検証記録: 「確認した」を記録で示す P.06

再検証: 変更のたびに確かめ直す P.07

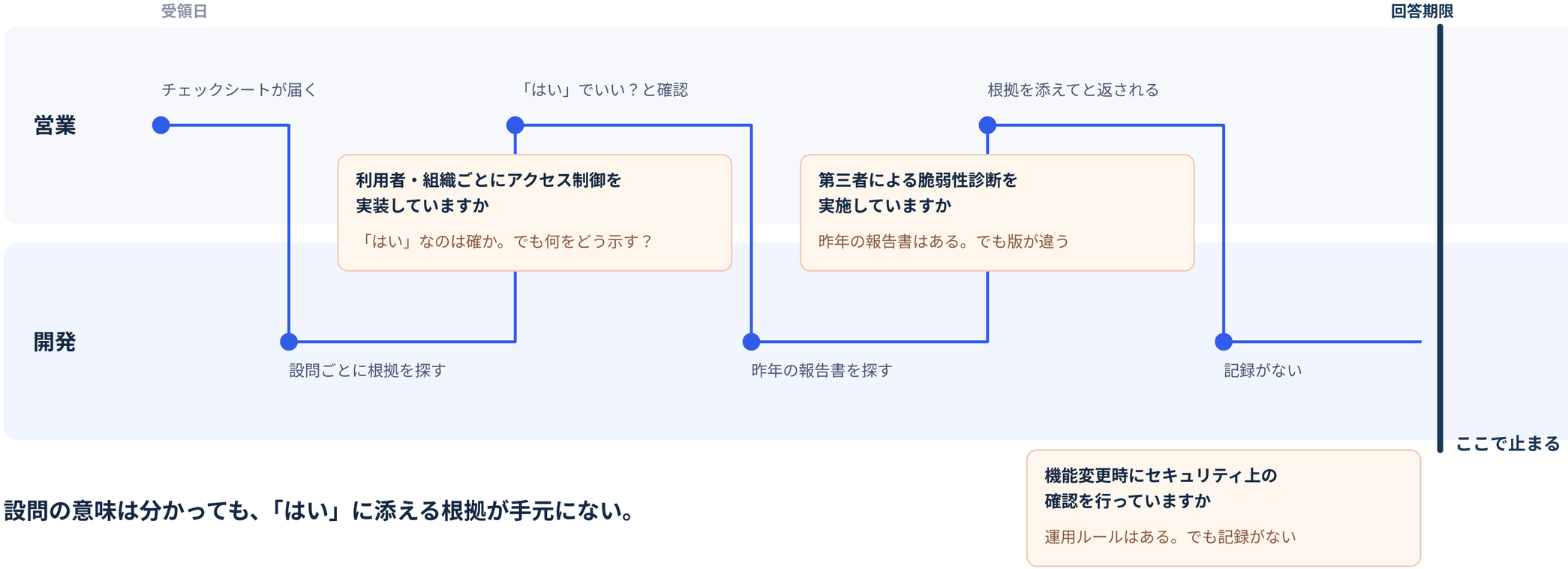
3 回答のつくり方

「はい」に根拠を添える3つの回答例 P.08

提出前の点検表 P.09

設問の意味は分かっても、「はい」の根拠がない

チェックシートは営業が受け取り、開発に回り、根拠を探すうちに回答期限が来ます。実装はしているのに、示すものがない。この資料は、その根拠の作り方を扱います。



設問の意味は分かっても、「はい」に添える根拠が手元にない。

設問を3つの観点に仕分ける

アクセス制御・脆弱性対応・変更管理の設問は、聞かれ方が違っても「権限」「検証記録」「再検証」のどれかで答えられます。
観点ごとに根拠を1つ用意しておけば、設問ごとに探し回らずに済みます。

設問の例（聞かれ方）	観点	用意するもの
利用者・組織ごとにデータが分離されていますか	権限	アクセス設計の1枚（誰が・何を・どの条件で）
管理者権限を持つ利用者を限定し、付与を記録していますか	権限	同じ1枚の「条件」欄と、権限変更の履歴
脆弱性診断を実施し、結果を保管していますか	検証記録	検証記録（対象と版・再現条件・結果・未検証範囲）
検出された脆弱性は修正済みですか	検証記録	修正差分と、修正後の再検証結果
機能追加・変更時にセキュリティ上の確認を行っていますか	再検証	リリースごとの再検証の記録（いつ・誰が・何を）
既知の脆弱性情報を定期的に確認していますか	再検証	公開情報（JVN・GitHub Advisory・CISA KEV＝悪用が確認された脆弱性の一覧など）と照合した記録
通信と保存データを暗号化していますか	対象外	証明書・暗号化設定の画面の記録
多要素認証・バックアップ・ログ保管を行っていますか	対象外	設定画面の記録と運用規程

自社で回るもの

設問を3つの観点に仕分け、観点ごとの根拠（権限の1枚・検証記録・再検証の記録）を作るところまでは自社でできます（P.05〜P.07）。

外に出す価値があるもの

検証記録のうち、まだ決めていない条件と経路（承認内容の書き換え・連携キー・AI 機能）を広く試し、未検証範囲まで書いた記録にするところ。

ここでいう検証記録は、操作ログやアクセスログの保管（これも別の設問で聞かれます）ではなく、「何を試して、どうなったか」を書いた記録のことです。

観点① 権限:「誰が・何を・どの条件で」を1枚で示す

問われているのは実装の有無ではなく、社外の人が読んで分かる形で示せるかどうかです。ロール×操作の表に「条件」を添えると、データの分離と管理者権限の設問（P.04 の上2行）に同じ1枚で答えられます。

アクセス設計の1枚（例） 誰が ／ 何を ／ どの条件で

	顧客データの閲覧	顧客データの出力	他組織の顧客データ	権限の変更
一般ユーザー	✓	条件つき 自分の担当分のみ	✕	✕
組織の管理者	✓	✓	✕	条件つき 履歴が残る
事業者のサポート担当	条件つき 顧客の許可・記録あり	✕	条件つき 期間限定・記録あり	条件つき 申請・承認・記録

✓ 許可　✕ 拒否　条件つき 承認・本人・期間・記録のどれが効くかを書く

表と実装がずれやすい3カ所

画面では止まるが、API を直接呼ぶと通る
表の「拒否」は画面と API の両方で確かめます

外部連携（API キー・OAuth）が全権限を持つ
連携用の権限を表に1行足します

一括出力や検索 API が組織をまたぐ
「出力」「検索」を「閲覧」と分けて書きます

自分でやるなら: 2つの組織で、ID を書き換えて試す

テスト用に組織 A・B を作り、それぞれに一般ユーザーを置きます。
A の利用者でログインし、画面ではなく API に組織 B の ID を指定して要求します。
拒否されれば表どおり。結果を版・日付と一緒に表へ書き込めば、
それが最初の検証記録になります。

観点② 検証記録: 「確認した」を記録で示す

「診断済みです」だけでは、「いつ・どの範囲で」と追加の質問が返ってくることがあります。いつ・どの版で・何を試し・何が拒否され・何を確かめていないか。これを書いた記録があれば、追加の質問にも答えられます。

診断報告書（手元にあるもの）

脆弱性診断報告書

昨年実施

- ⚠ 対象: 当時の版（版の記載なし）
- ☑ 診断項目に沿った検出結果の一覧
- ⚠ 結果: 指摘なし

報告書は必要です。ただし版が変わると
根拠が古くなり、「他組織のデータが見えないか」
のような業務上の重要な操作は、診断項目に
入っていないことがあります。

いまの版？

試していない操

検証記録（設問に添えるもの）

検証記録

版ごと・毎回

- ☑ 対象と版（アプリ・環境・日付）
- ☑ 再現条件（誰が・何を・どの条件で）
- ☑ 結果（拒否された ／ 通った）
- ☑ 修正差分と、修正後の再検証の結果
- ☑ 確かめていない範囲

自分でやるなら: 1操作1行の表

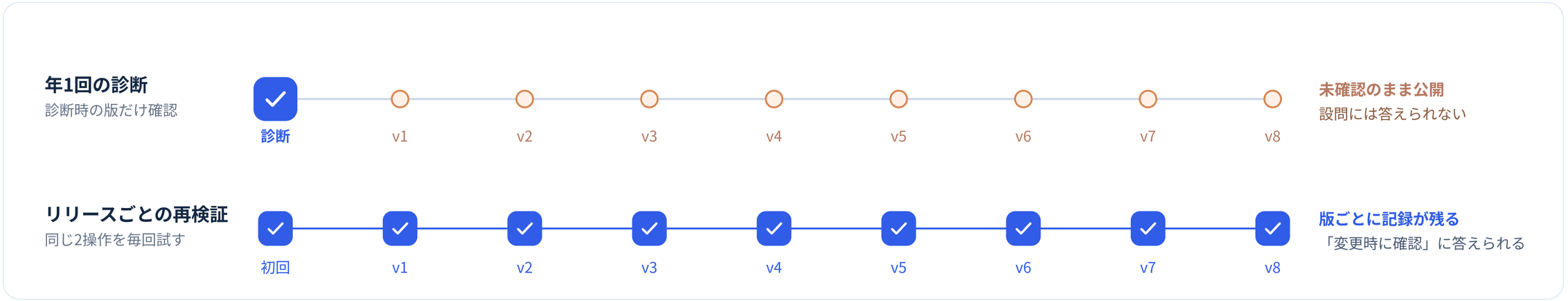
スプレッドシートに6列を作ります。

日付 ／ 対象の版
誰として ／ 何を指定して
結果（拒否・通過）
確かめていない範囲

これをそのまま回答に添付できます。
増えるのは行だけなので、
版が変わっても作り直しは要りません。

観点③ 再検証: 変更のたびに確かめ直す

年1回の診断と毎週のリリースは噛み合いません。「変更時に確認していますか」に答えるには、変更のたびに確かめ直した記録が要ります。全部を毎回やる必要はなく、事故になる操作を決めて繰り返します。



01 ／ 決める

確認する操作を2つ決める

他組織のデータ取得、承認なしの実行など、起きたら事故になる操作から。まず2つに絞るのは、毎回のリリースで欠かさず回せる量に抑えるためです。回せるようになったら増やします。

02 ／ 試す

リリース前に同じ操作を試す

前回と同じ手順を、同じテストアカウントで繰り返します。手順書にしておけば、担当が替わっても回ります。

03 ／ 残す

結果を記録に追記する

版・日付・拒否／通過を1行ずつ。前回との差分だけ書けばよいので、記録は毎回ふくらみません。公開情報（既知の脆弱性）が出たときも同じ手順です。

上の3ステップは、手順書とスプレッドシートで自社で回せます。回し方は No.24 にあります。

回答の書き方:「はい」に根拠を添える

回答欄には「はい」だけでなく、何を・いつ・どの範囲で確かめたかを一文で添えます。安全かどうかの判断は提出先が行います。
自社は、確かめた事実と範囲を示せば十分です。

設問 利用者・組織ごとにデータが分離されていますか

根拠なし	はい。マルチテナント構成で分離しています。
根拠つき	はい。すべての要求で利用者の所属組織と要求データの組織を照合しています。他組織の顧客データの取得・出力・検索の3操作が拒否されることを検証環境で確認済みです（検証記録 No.◯、対象版 v◯.◯、◯年◯月◯日）。確かめていない範囲も同じ記録に記載しています。

設問 検出された脆弱性は修正済みですか

根拠なし	はい。指摘事項はすべて対応済みです。
根拠つき	重大度「高」の指摘は修正し、修正後に同じ操作が拒否されることを確認しました。重大度「中」の指摘は、該当機能を管理者だけが使える状態に制限したうえで、◯月のリリースで修正します（修正差分・再検証結果・対応予定は別紙）。

設問 機能変更時にセキュリティ上の確認を行っていますか

根拠つき	はい。リリースのたびに、他組織データの取得と権限変更の2操作が拒否されることを確認し、記録しています。それ以外の変更はコードレビューで確認しています。

書き方の3原則

範囲・版・していないこと

範囲を書く: どの操作を確かめたか。
版と日付を書く: いつの状態の話か。
していないことも書く: 確かめていない範囲。提出先はこれで判断できます。

書かないこと

「安全です」「すべて対応済み」

確かめた事実と範囲を示せば、設問の意図には答えられます。言い切りは、後で食い違ったときに信頼を失います。

提出前の点検表:「ない」に丸が付いた行が、次に用意するもの

6つが揃えば、アクセス制御・脆弱性対応・変更時の確認に関する設問には、同じ資料を繰り返し根拠として使えます。期限まで2週間なら、まず Q1（アクセス設計の1枚）と Q6（分担と期限）を。Q2〜Q5 は「実施予定日」を書いて回答できます。

No.	揃えるもの	例	状態	担当・期限
Q1	**アクセス設計の1枚**	ロール×操作の表、「条件つき」の中身（承認・本人・期間・記録）、組織の境界	ある ／ 一部 ／ ない	
Q2	**検証記録**	対象と版、再現条件、結果（拒否／通過）、確かめていない範囲	ある ／ 一部 ／ ない	
Q3	**修正差分と再検証結果**	直した箇所、修正後に同じ操作が拒否されたことの記録	ある ／ 一部 ／ ない	
Q4	**変更時の再検証ルール**	いつ（リリース前）・誰が・どの操作を試すか。公開情報が出たときの手順	ある ／ 一部 ／ ない	
Q5	**公開情報との照合の記録**	利用中のライブラリと、JVN・GitHub Advisory・CISA KEV などを突き合わせた日付と結果	ある ／ 一部 ／ ない	
Q6	**回答の分担と期限**	営業が窓口、開発が根拠を用意。設問番号ごとに担当と提出日	ある ／ 一部 ／ ない	

自社で回るもの

Q1 は P.05 の表、Q2 は P.06 の6列の表、Q4 は P.07 の3ステップをそのまま使えます。

外に出す価値があるもの

Q2・Q3 のうち、まだ決めていない条件と経路（承認内容の書き換え・連携キー・AI機能）を広く試し、未検証範囲まで書いた記録にするところ。共同検証プログラムでは2操作から始めます。

揃っていないものから、一緒に作りませんか。

P.05 の表と P.06 の記録を作り、決めた条件で画面と API から試すところまでは自社でできます（No.09 P.09）。外に出す価値があるのは、まだ決めていない条件と経路（承認内容の書き換え・連携キー・AI 機能）を広く洗い出して試し、試さなかったものも未検証範囲として書いた、取引先に出せる記録にするところです。初回のご相談は30分・無料です。

お申し込み・お問い合わせ　**securiy-preview.pages.dev**　のフォームから
Tempura technologies株式会社

※ 本資料はチェックシートへの回答の整理を目的としたものです。ガイドラインへの適合は保証しません。すべての攻撃の網羅や「安全」の総合判定は保証しません。
継続検証システムは設計・検証段階で、提供時期は未定です。金融の規制適合審査、決済基盤全体の監査、カード不正判定は対象外です。

無料相談

30分無料相談

P.09 の点検表をもとに、揃っていないものと、初回に検証する2操作を一緒に決めます。本番の認証情報や顧客データは不要です。

相談を申し込む →

共同検証プログラムとは

検証テーマを一緒に作る企業に、初回スコープで提供する枠組み

1アプリ・1検証環境・2操作、準備完了後10営業日・合計40時間（修正案と再検証は2件・12時間まで）。費用は着手前に提示し、事例としての公開は任意です。

次に読む

資料3点セット（課題編・紹介編・活用編）

AI攻撃で不正操作が成立する理由、Securiy の仕組み、業種別の検証シーンはこちら。