RocketBase
Fullstack-Developer-Team

MUSTER

SOFTWARE-CHECK · SICHERHEITSDURCHSICHT

Auftragsportal Beispielbetrieb GmbH

nach der Systematik des OWASP ASVS, dazu Geschäftslogik, Ausfall von Gegenstellen, Auslieferung und Betrieb

PRÜFGEGENSTAND
Portal mit Terminbuchung und Online-Zahlung, TypeScript/Node, PostgreSQL

PRÜFSTAND
Repository vom 12.10., Migrationen bis 064

PRÜFER
Marten Prieß, RocketBase
13. bis 16.10.

1	5	12	0
kritisch	hoch	mittel	Lücken von außen

Musterbericht. Betrieb, System und Befunde sind ausgedacht und keinem Kunden entnommen. Aufbau, Prüfverfahren, Maßstab und Schreibweise entsprechen unseren echten Berichten.

ABLAUF

Erst verstehen, dann prüfen

Ein Scanner liest Code. Ich will vorher wissen, was die Software leisten soll, wer sie benutzt und was schiefgehen darf und was nicht. Deshalb beginnt jeder Check mit dem Menschen, der sie gebaut hat.

Vorher
kostenlos

1

Kurzes Telefonat

15 MINUTEN · LEITUNG

Was ist gebaut, was steht an, etwa ein Go-live? Passt es in einen Check, machen wir den Termin zum Zeigen aus.

Tag 1
etwa 1,5 Stunden

2

Die Software gezeigt bekommen

MIT DEM ENTWICKLER ODER TEAM

Ihr zeigt mir das System, so wie es benutzt wird. Ich frage viel, um das Zielbild zu verstehen, bevor ich eine Zeile Code lese.

- Wer arbeitet damit?
- Wo fließt Geld?
- Welche Systeme hängen dran?
- Was darf nie passieren?
- Was ist bewusst so entschieden?

Tag 1–3
bei mir

3

Code sichten und messen

ZWEI DURCHGÄNGE, LAUFENDE INSTANZ

Einmal mit eurer Dokumentation, einmal bewusst ohne. Was sich messen lässt, wird gemessen: gleichzeitige Zugriffe, ausfallende Gegenstellen, die Zustandsmeldung.

Tag 3
kurz, per Telefon oder Chat

4

Rückfragen zu den Befunden

MIT DEM ENTWICKLER · VORAB-STICHPUNKTE

Bevor etwas im Bericht steht, frage ich nach: Ist das Absicht? Gibt es dafür einen Grund, den ich nicht sehe? Was dabei nicht hält, fliegt raus. Die wichtigsten Punkte bekommt ihr vorab, damit das Team schon anfangen kann.

Tag 4–5
etwa 1,5 Stunden

5

Dokumente und Abschlusstermin

LEITUNG UND TEAM

Management-Summary, Bericht, Papier zum Betrieb und bei Bedarf Fragen an eure Dienstleister. Im Termin führe ich die wichtigsten Befunde live vor, und wir legen die Reihenfolge gemeinsam fest.

4 bis 5 Werktage vom Zeigen bis zum Abschlusstermin. Ihr braucht dafür etwa drei Stunden eures Teams, den Rest mache ich.

FÜR DIE LEITUNG

Sicher gebaut, aber noch nicht betriebsfest

Ein Angreifer von außen kommt nach meiner Prüfung nicht an Kundendaten. Die Risiken liegen dort, wo das Portal auf Zahlungsanbieter und Warenwirtschaft trifft, bei gleichzeitigen Buchungen und im Weg, auf dem Änderungen live gehen.

Die Risiken

KRITISCH

Bezahlte Buchungen werden storniert

Ist der Zahlungsanbieter eine Stunde nicht erreichbar, storniert das Portal jede Buchung, deren Zahlung in der Zeit bestätigt werden sollte. Der Kunde hat bezahlt, der Termin ist weg.

HOCH

Termine und Erstattungen doppelt

Kommen zwei Anfragen gleichzeitig, nimmt das Portal beide an. Ein Termin geht zweimal weg, eine Erstattung zweimal hinaus. Gemessen, nicht vermutet.

HOCH

Ein Schlüssel öffnet alle Kundendaten

Der Zugang der Warenwirtschaft läuft nie ab und liefert die Daten aller Kunden, nicht nur der offenen Aufträge.

HOCH

Live ohne zweiten Menschen

Der KI-Assistent setzt die Freigabe selbst. Ein Ausfall nach einer Auslieferung fällt nicht auf, weil die Zustandsmeldung die Datenbank nicht prüft.

VOR DEM GO-LIVE

≈ 3 Entwicklertage

Stornierung nur bei echter Ablehnung, Sperren bei Buchung und Erstattung, Pflicht-Review vor jeder Auslieferung, eine Zustandsmeldung, die einen Ausfall erkennt.

ALLE KRITISCHEN UND HOHEN BEFUNDE

≈ 1 Entwicklerwoche

Danach im laufenden Betrieb die mittleren Befunde und die Dokumentation auf den Stand des Codes.

Was in den IT-Betrieb übergehen sollte

Freigabe von Auslieferungen · Schnittstellenschlüssel mit Ablaufdatum · Alarmierung mit klarer Zuständigkeit · eine erprobte Rücksicherung der Datenbank.

Zur Entscheidung, kein Befund: Erstattungen gehen ohne Betragsgrenze und ohne zweite Freigabe in die Bankdatei. Das ist so gewollt, sollte aber bewusst entschieden sein.

So läuft euer Software-Check

Festpreis je System, Bericht nach 4 bis 5 Werktagen. Preis und Rückruf auf der Seite zum Check.

rocketbase.io/software-check

0 · EINORDNUNG UND GRENZEN

Was geprüft wurde, und wie

Ein **Review, kein Penetrationstest und kein Testat.** Die Kapitel des OWASP ASVS geben den Rahmen, damit nachvollziehbar bleibt, was geprüft wurde. Eine Zertifizierung nach einer ASVS-Stufe beansprucht der Bericht nicht. Fünf Felder gehen über den ASVS hinaus, weil dort bei mit KI gebauter Software die schwersten Befunde liegen.

ASVS V2–V4

Anmeldung, Sitzung, Rechte

Kann Kunde A die Daten von Kunde B sehen? Kommen Rollen aus der Datenbank?

ASVS V5, V6, V13

Eingaben, Schnittstellen, Geheimnisse

Alles parametrisiert? Wer kann mit einem Schlüssel was abrufen, und läuft er ab?

ASVS V11 · gemessen

Geschäftslogik, gleichzeitige Zugriffe

Zwei Buchungen auf den letzten Platz? Eine Zahlung zweimal hinaus?

über ASVS hinaus · gemessen

Ausfall von Gegenstellen

Was passiert, wenn Zahlungsanbieter oder Warenwirtschaft nicht antworten?

über ASVS hinaus

Datenmodell

Grenzwerte in der Datenbank? Indizes? Laufen alle Migrationen auf leerer Datenbank?

ASVS V1, V7 · und mehr

Auslieferung und Betrieb

Prüft ein Mensch vor dem Live-Gang? Erkennt die Zustandsmeldung einen Ausfall?

ASVS V8 · DSGVO

Datenschutz und Löschung

Löschkonzept? Was steht in den Logs? Welche Daten gehen an Dritte?

ASVS V10, V14

Abhängigkeiten, Versionen

Bekannte Schwachstellen? Laufzeit noch mit Sicherheitspatches?

über ASVS hinaus

Dokumentation gegen Code

Stimmt, was Runbook und Übergabe sagen? Wer übernimmt, liest zuerst die Doku.

Verfahren. Code, Migrationen, Infrastruktur und Pipelines zweimal gelesen: einmal mit der Dokumentation, einmal nur entlang des Codes, Kommentare als Behauptung. Eine lokale Instanz, gegen die gleichzeitige Anfragen und ausfallende Gegenstellen gemessen wurden. Abhängigkeiten mit npm audit, OSV und Trivy. Jeder Befund zum Schluss noch einmal von der Fundstelle aus nachgeprüft.

Nicht geprüft: die Warenwirtschaft selbst, die reale Cloud-Konfiguration, produktive Daten, kein aktiver Angriff, keine Last.

MASSSTAB DER EINSTUFUNG

KRITISCH trifft viele Kunden gleichzeitig oder beschädigt Geld und Daten dauerhaft, oft unbemerkt

HOCH deutlicher Schaden im Einzelfall oder nach dem Go-live nur teuer zu beheben

MITTEL begrenzte Wirkung, auf Dauer nicht tragbar. Niedrige Befunde sind nicht aufgeführt.

1 · ERGEBNIS IN KÜRZE

Die klassischen Stellen halten. Die Befunde liegen woanders.

Anmeldung, Zugangskontrolle, Geheimnisse und Einschleusung sind ohne Befund. Eine Lücke, über die ein anonymer Angreifer an Kundendaten kommt, wurde nicht gefunden. Das Portal verlässt sich aber darauf, dass jede Gegenstelle antwortet und nie zwei Anfragen gleichzeitig dieselben Daten anfassen.

18 Befunde — davon 4 neu unter den kritischen und hohen, 1 als Entscheidung im Code, 1 schon geführt



1 kritisch · 5 hoch · 12 mittel — dazu 4 Stellen, an denen die Dokumentation vom Code abweicht

Kapitel 3

Ausfall der Gegenstellen

Eine Zeitüberschreitung zählt wie eine Ablehnung. Nach drei davon wird storniert.

Kapitel 4

Gleichzeitige Zugriffe

Buchung und Erstattung lesen, entscheiden und schreiben ohne Sperre.

Kapitel 6

Auslieferung

Live ohne menschliche Prüfung, kein Alarm, kein Rückweg.

2 · WAS HÄLT

Was ausdrücklich so bleiben darf

Kein Lob, sondern Bestand. Diese Stellen könnt ihr verteidigen, wenn der nächste Berater alles neu bauen will.

✓ Anmeldung vollständig
Signatur, Aussteller, Zielgruppe festgelegt; ohne Konfiguration kein Start. `auth/verify.ts`

✓ Zugänge standardmäßig zu
Gemessen: alle 46 geschützten Routen ohne Nachweis abgelehnt.

✓ Geheimnisse und Einschleusung
Nichts im Code oder Frontend, alle Abfragen parametrisiert.

✓ Geldbeträge in ganzen Cent
Vom Formular bis zur Bankdatei, ohne Fließkomma.

✓ Übergabe an die Warenwirtschaft
Zustandsautomat, eingefrorener Beleg. Vorlage für eigene Reparaturen. `erp/uebergabe.ts`

✓ Entscheidungen nummeriert
Mit Datum und Begründung. Erst das macht ein Review in wenigen Tagen möglich.

3 · AUSFALL VON GEGENSTELLEN · 4 · GLEICHZEITIGE ZUGRIFFE

Vor dem Go-live

3.1.1 Ein Ausfall beim Zahlungsanbieter storniert bezahlte Buchungen

KRITISCH · ASVS V11 · gemessen · Entscheidung im Code

FUNDSTELLE `payment/bestaetigung.ts:88-131` · `buchung/stornieren.ts`

BEOBACHTUNG Jedes Ergebnis außer einer Bestätigung zählt als Fehlversuch. Zeitüberschreitung, 503 und 429 wiegen so schwer wie eine echte Ablehnung. Nach drei Fehlversuchen im 20-Minuten-Takt wird storniert und der Termin freigegeben. Der Kunde bekommt eine Stornomail für einen bezahlten Termin.

GEMESSEN

```
$ Gegenstelle antwortet 503, Takt 20 min
offene Buchungen: 14 → 0 nach 61 min
Zahlungen beim Anbieter: 14 weiter belastet
```

EMPFEHLUNG Nur eine ausdrückliche Ablehnung zählt. Transportfehler, 5xx und 429 setzen den letzten Fehler und lösen einen Betriebsalarm aus.

4.1.1 Zwei Buchungen auf den letzten freien Termin werden beide angenommen

HOCH · ASVS V11 · gemessen · neu

FUNDSTELLE `buchung/anlegen.ts:40-72`

BEOBACHTUNG Liest den freien Platz, entscheidet im Speicher, schreibt ohne Sperre. Die nächtliche Bereinigung löscht die doppelte Buchung, ohne einen der Kunden zu informieren.

GEMESSEN

```
$ Integrationstest, echte Datenbank
erwartet: 1 angenommen, 1 abgelehnt
tatsächlich: 2 angenommen
```

EMPFEHLUNG Zeilensperre beim Lesen, Schreiben in derselben Transaktion. Vorbild: `lager/reservierung.ts:18-35`, die es schon richtig macht. Der Test liegt dem Bericht bei.

4.1.2 Eine Erstattung kann zweimal in die Bankdatei gelangen

HOCH · ASVS V11 · teilweise geführt

FUNDSTELLE `erstattung/zahllauf.ts:157`

BEOBACHTUNG Liest freigegebene Erstattungen ohne Sperre und markiert sie erst nach dem Hochladen. Zwei Zahlläufe kurz nacheinander, etwa durch einen Doppelklick, enthalten dieselben Erstattungen.

EMPFEHLUNG Sperre im Zahllauf, Festschreibung vor dem Hochladen, `CHECK (betrag > 0)` in der Datenbank.

5 · ZUGÄNGE · 6 · AUSLIEFERUNG UND BETRIEB

Vor dem Go-live, Fortsetzung

5.1.1 Ein Schlüssel der Warenwirtschaft liefert die Daten aller Kunden

HOCH · ASVS V4 · neu

FUNDSTELLE `api/erp-export.ts:22-61`

BEOBACHTUNG Ein einziger, nie ablaufender Schlüssel gibt Name, Adresse und Buchungen aller Kunden heraus, nicht nur der offenen Aufträge. Er liegt zusätzlich beim Dienstleister der Warenwirtschaft.

EMPFEHLUNG Eigener, wechselbarer Schlüssel je Gegenstelle mit Ablaufdatum; Ausgabe nur der Aufträge, die übergeben werden sollen.

6.1.1 Änderungen gehen ohne zweiten Menschen live

HOCH · ASVS V1 · neu

FUNDSTELLE `.github/workflows/automerge.yml`

BEOBACHTUNG Zusammengeführt wird, sobald ein Label gesetzt ist, und dieses Label setzt der KI-Assistent selbst. Ausliefern lässt sich jeder Zweig, nicht nur der Hauptzweig.

EMPFEHLUNG Pflicht-Review durch eine Person auf dem Hauptzweig, Auslieferung nur von dort mit grünem Testlauf.

6.1.2 Die Zustandsmeldung sagt „ok“, auch wenn die Datenbank weg ist

HOCH · ASVS V7 · gemessen

FUNDSTELLE `api/health.ts`

BEOBACHTUNG Prüft nur, ob der Server antwortet, und entscheidet genau damit nach jeder Auslieferung, ob die neue Version live bleibt. Eine Alarmierung gibt es nicht.

GEMESSEN

```
$ Datenbank stoppen, dann
GET /api/health → 200 {"status":"ok"}
```

EMPFEHLUNG Abfrage gegen die Datenbank mit kurzem Zeitlimit, bei Fehler 503. Alarm mit fester Zuständigkeit, Auslieferung mit Rückweg.

3 BIS 7 · WEITERE BEFUNDE

Im laufenden Betrieb beheben

MITTEL **Übergabe an die Warenwirtschaft ohne Wiederholung.** Gescheiterte Aufträge bleiben auf `uebergabe_offen`, nichts holt sie nach. neu

MITTEL **Mailversand ohne Zeitlimit** in derselben Anfrage wie die Buchung. Ein langsamer Mailanbieter hält die Buchung offen.

MITTEL **Keine Grenzwerte in der Datenbank.** Null CHECK-Bedingungen in 64 Migrationen.

MITTEL **Fremdschlüssel ohne Index:** 31 von 38. Mit wachsenden Daten werden Löschen und Auswertungen langsam.

MITTEL **Rolle aus dem Token statt aus der Datenbank.** Wer gesperrt wird, behält seine Rechte bis zu 12 Stunden.

MITTEL **Begrenzung der Anfragen installiert, aber nirgends eingebunden.** neu

MITTEL **214 Integrationstests laufen in der Pipeline nie,** weil dort keine Datenbank bereitsteht.

MITTEL **Laufzeitumgebung seit April ohne Sicherheitspatches.** Das meldet kein Werkzeug.

MITTEL **Kein Löschkonzept;** Einwilligungen lassen sich ändern und löschen.

MITTEL **Die Schwärzung im Logging zielt auf Feldnamen, die der Code nicht verwendet.** Der Test ist grün, weil er genau diese Schreibweise prüft.

8 · DOKUMENTATION GEGEN CODE

Wo die Doku etwas anderes sagt

DOKUMENT	SAGT	CODE TUT
`docs/runbook.md:41`	Nachtlauf holt gescheiterte Übergaben nach	liest nur neue Aufträge
`docs/betrieb.md:12`	Sicherung mit Test der Rücksicherung	Rücksicherung nirgends beschrieben
`docs/sicherheit.md:30`	Anfragen werden begrenzt	nicht eingebunden
`README.md:8`	Tests laufen bei jeder Änderung	Integrationstests übersprungen

9 · EMPFOHLENE REIHENFOLGE

In drei Schritten zum Go-live und darüber hinaus

SOFORT

Vor dem Go-live

einzeln klein, schützen am meisten

1. Fehlversuche beim Zahlungsanbieter nach Art unterscheiden, Alarm statt Storno **3.1.1**
2. Sperren bei Buchung und Zahllauf, Grenzwerte in der Datenbank **4.1.1, 4.1.2**
3. Pflicht-Review, Auslieferung nur vom Hauptzweig **6.1.1**

KURZFRISTIG

Wenige Tage

geringer Aufwand

4. Zustandsmeldung mit Datenbank, Alarm, Rückweg **6.1.2**
5. Eigener Schlüssel mit Ablaufdatum, nur offene Aufträge **5.1.1**
6. Rolle aus der Datenbank, Anfragebegrenzung einbinden

MITTELFRISTIG

Laufender Betrieb

größerer Umfang

7. Übergaben nachholen, Zeitlimits für alle Gegenstellen
8. Integrationstests mit Datenbank, Indizes
9. Löschkonzept, unveränderliche Einwilligungen
10. Dokumentation auf den Stand des Codes

Zur Entscheidung beim Beispielbetrieb, kein Befund. Erstattungen gehen ohne Betragsgrenze und ohne zweite Freigabe in die Bankdatei. Eine zweite Freigabe ab einem festen Betrag wäre die vorsichtigere Wahl.

10 · SCHLUSSBEMERKUNG

Das Auftragsportal ist kein Sanierungsfall. Die Kapitel, in denen Prüfungen dieser Art sonst die schwersten Mängel finden, sind ohne Befund. Die Befunde liegen dort, wo das Portal auf andere Systeme trifft und wo zwei Dinge gleichzeitig passieren. Mit wenigen Testkunden fällt das nicht auf. Im Betrieb sieht es niemand mehr.

ANHANG A · PAPIER ZUM BETRIEB

Was man im Code nicht sieht

Mailweg

Eine Unzustellbarkeitsmeldung mit der Buchungsnummer im Text landet als Kundenantwort am Vorgang. Der Fall sieht bearbeitet aus. Er ist es nicht.

Backup

Es gibt eins. Hat schon einmal jemand zurückgespielt? Ein voller Rücksprung löscht alles seither Gebuchte.

Protokoll der Vorgänge

Festgehalten wird, was sich geändert hat, nicht was gescheitert ist. „Warum kam keine Bestätigung?“ bleibt unbeantwortet.

Alarmierung

Wer bekommt nachts eine Nachricht, wenn das Portal steht, und was tut er dann?

ANHANG B · FRAGEN AN EUREN DIENSTLEISTER

Zum Weitergeben

Was passiert, wenn dieselbe Buchung zweimal bei der Warenwirtschaft ankommt? eher Dienstleister

Warum ich frage: Das Portal wiederholt bei Zeitüberschreitung. Erkennt die Warenwirtschaft das nicht, entsteht ein zweiter Auftrag.

Lässt sich der Schnittstellenschlüssel ohne Ausfall wechseln? eher Dienstleister

Warum ich frage: Er öffnet heute die Daten aller Kunden (5.1.1). Ein Wechsel muss geplant sein, bevor er nötig ist.

Wer erfährt von Störungen beim Zahlungsanbieter? vermutlich ihr

Warum ich frage: Wer die Statusseite abonniert, weiß vor dem ersten Kundenanruf Bescheid.

Was zum Software-Check gehört

Management-Summary und Bericht wie in diesem Muster, Vorab-Stichpunkte, ein Abschlusstermin von etwa anderthalb Stunden mit Live-Nachweisen, das Papier zum Betrieb und bei Bedarf Fragen an eure Dienstleister. Festpreis je System, Preis und Ablauf stehen auf der Seite zum Check.

Preis und Rückruf: rocketbase.io/software-check

Scannen für Preis und Rückruf

Marten Prieß · marten@rocketbase.io