APEX ENGLISH
Your voice. Global.

B2 · BUSINESS ENGLISH

UNIT 8 · B8A

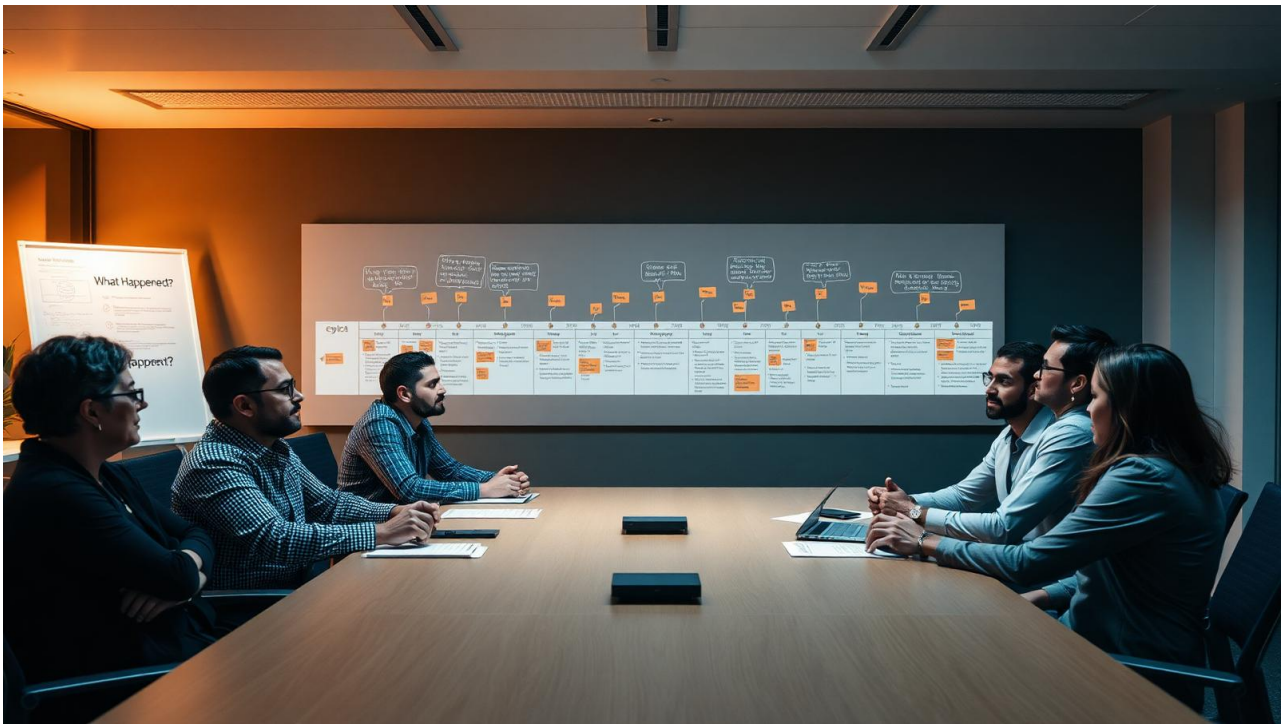
When systems fail

Theme · Tech & Security

Unit 8 · B8A

When systems fail

Theme: Tech & Security



Learning objectives

- Talk about incidents and outages using the passive naturally — not to hide who did what.
- Run a blameless post-mortem: separate what happened, why, and what we change.
- Use 'have something done' to describe fixes ordered but not done by you.

Ask the teacher — notes to bring to class

Class Book

Warm-up · Your last outage

Think of the last time a system YOU depend on went down — a login, a bank app, a payment terminal, your company's own tool. In 45 seconds out loud: what broke, how did you FIND OUT, and how did the company communicate? Then record a 30-second voice note.

How did you find out — the system, or Twitter/X?

How LONG before someone in the company said anything?

Did they explain WHAT happened, or just say 'sorry'?

Bonus: did anyone get blamed by name in public? Should they have?

Lead-in: incident or drama?

Decide: BLAMELESS (describes the system) or BLAME-LOADED (names a villain).

1. 'A configuration change was pushed at 14:02 that removed the health-check.'

---

2. 'Ravi broke the auth service again.'

---

3. 'The failover was never tested in production — we caught that in the post-mortem.'

---

4. 'This wouldn't have happened if ops actually cared.'

---

followUp: In your last incident, was the language you used blameless or blame-loaded?

BLAMELESS

BLAME-LOADED

Reading: The blameless post-mortem

When a system fails, the temptation is always the same — to find the person who pushed the wrong button and make them the story. It is emotionally satisfying and completely useless. The person who pushed the button did so because the system allowed the button to be pushed, because the review that should have caught it was skipped, because the alert that should have fired was silent, and because the runbook that should have said 'stop' didn't exist. Blame stops at the first name; a blameless post-mortem doesn't stop until it reaches the third or fourth WHY.

The grammar of the incident report is the passive, and it is used for a specific reason — not to hide the actor, but to keep the focus on the system. 'A configuration change WAS PUSHED at 14:02 that removed the health-check' is more useful than 'Ravi pushed a bad config'. The passive says: the interesting thing is what happened to the system, not who happened to be at the keyboard. This is why the whole family of passive forms — 'the database HAD BEEN restarted', 'the alert HAS NOT YET

Ask the teacher — notes to bring to class

BEEN acknowledged', 'the failover IS BEING tested', 'the root cause WILL HAVE BEEN identified by Friday' — is the working vocabulary of ops writing. When you do want a name, name a role, not a person: 'the on-call engineer' rather than 'Ravi'.

A useful post-mortem has three sections and never more: What happened (a timeline, in the passive, minute by minute); Why it happened (the chain of contributing causes, at least three deep — never one); and What we're changing (concrete actions with owners and dates — not 'we should be more careful'). The single most valuable phrase in the whole document is 'we HAVE the alert THRESHOLD RECONFIGURED by Wednesday' — using 'have something done' to make clear that the fix is owned but the person doing it is not necessarily the person writing the report. If your post-mortem doesn't produce at least three specific changes with dates, you didn't hold a post-mortem — you held a wake.

1. The point of a BLAMELESS post-mortem is…

   (A) To protect senior engineers

   (B) To keep the focus on the system, not the person, so the real chain of causes surfaces

   (C) To avoid writing anything down

2. 'A configuration change was pushed at 14:02' uses the passive because…

   (A) English grammar rules force it

   (B) The interesting thing is what happened to the system, not who happened to be at the keyboard

   (C) It sounds more formal

3. A useful post-mortem has how many sections?

   (A) One: What happened

   (B) Three: What happened, Why, and What we're changing

   (C) Seven

4. 'We'll have the alert threshold reconfigured by Wednesday' is an example of…

   (A) A passive voice past tense

   (B) 'Have something done' — the fix is owned, but not necessarily by the writer

   (C) A pronunciation drill

Vocabulary in context

Match the incident term to its plain-English meaning.

outage — a period when a service is unavailable

root cause — the underlying reason, not just the last thing that broke

post-mortem — the meeting and document after an incident, to learn from it

runbook — the step-by-step guide for handling a known kind of incident

failover — the automatic switch to a backup system when the primary fails

blast radius — how far the damage spread — which users, which regions, which systems

Ask the teacher — notes to bring to class

Vocabulary: incidents, security and post-mortem language

outage / downtime — the service is unavailable

degradation — slower or partial — not fully down

trigger (n./v.) — the last thing that set the incident off

root cause — the underlying reason — usually a chain, not one thing

contributing factor — something that made the incident worse or more likely

blast radius — the scope of impact — who, where, how much

breach — unauthorised infiltration of data or systems

vulnerability — a weakness that could be exploited

exploit (n./v.) — code or method that abuses a vulnerability

patch (v./n.) — fix a vulnerability in software

credential stuffing — attackers systematically trialling leaked passwords against your login

MFA (multi-factor auth) — second factor to prove it's really you

timeline — minute-by-minute record of what happened

runbook — the pre-written playbook for handling a known incident

on-call engineer — the person paged first when an alert fires

the five whys — keep asking 'why' until you reach a process, not a person

action items — the changes we'll make, with owners and dates

blameless (adj.) — impartial and focused on the system, rather than apportioning individual culpability

practice

Complete each sentence with ONE word / phrase.

1. The whole payments API went down for 47 minutes — a total ___.

2. The ___ ___ was a missing null-check in the auth service, but the ___ ___ ___ was that the code review had been skipped under launch pressure.

3. Attackers used a stolen password list against our login — a classic ___ ___ attack.

Ask the teacher — notes to bring to class

4. The ___ ___ was small — only Portuguese users on the mobile app were affected.

5. We're rolling out ___ across all admin accounts by end of quarter — no more password-only logins for anything sensitive.

6. The alert fired at 02:14; the ___-___ ___ acknowledged in 4 minutes — under our SLA.

Practice 2 — collocations & rewrite

Rewrite each sentence replacing the underlined phrase with ONE word/expression from this lesson's vocabulary. Keep the meaning identical.

1. The <u>underlying reason for the failure</u> was a database deadlock that wasn't caught in staging.

2. The <u>person currently handling alerts</u> was paged at 3 AM but didn't respond for ten minutes.

3. We need to <u>trigger the automatic switch to a backup</u> manually to verify it works.

4. The <u>period when the service was unavailable</u> lasted exactly 47 minutes during peak traffic.

5. The report must be <u>impartial and focused on the system</u> to ensure we actually learn from this.

Practice 3 — discursive writing (business memo)

prompt: Write a summary memo for an internal engineering post-mortem following a 45-minute payment gateway outage.

Use at least FOUR of the target expressions from this unit.

Adopt a professional register: concise, structured, executive-friendly.

Include a clear recommendation or takeaway in the final sentence.

sample: The outage on Tuesday morning was a total failure of the primary gateway for 45 minutes, resulting in a significant blast radius across all Southern European markets. While the root cause was a malformed API call, a major contributing factor was the silence of the automated alerts, which had been accidentally disabled during the previous night's maintenance. In line with our blameless culture, we focused on the system's failure to notify the on-call engineer rather than the individual error. We have

Ask the teacher — notes to bring to class

now logged several action items, including the implementation of a secondary watchdog service and a mandatory review of alert thresholds. We will have the updated runbook published by Friday. Recommendation: we must automate alert verification as part of every deployment pipeline to ensure this silence never recurs.

Practice B — Incident collocations

Match each verb to the noun it most naturally goes with in incident language.

acknowledge — an alert

page — the on-call engineer

roll back — a bad deploy

patch — a vulnerability

trigger — a failover

log — an action item

Practice C — Build a post-mortem line

Use each cluster of words to write ONE sentence you could put in a real post-mortem.

1. trigger · null-check · auth service · 14:02

2. blast radius · Portuguese · mobile · 12 minutes

3. runbook · updated · SRE lead · Wednesday

4. on-call · paged · acknowledged · 4 minutes

Practice D — Odd one out

Which word in each group does NOT belong to the same field?

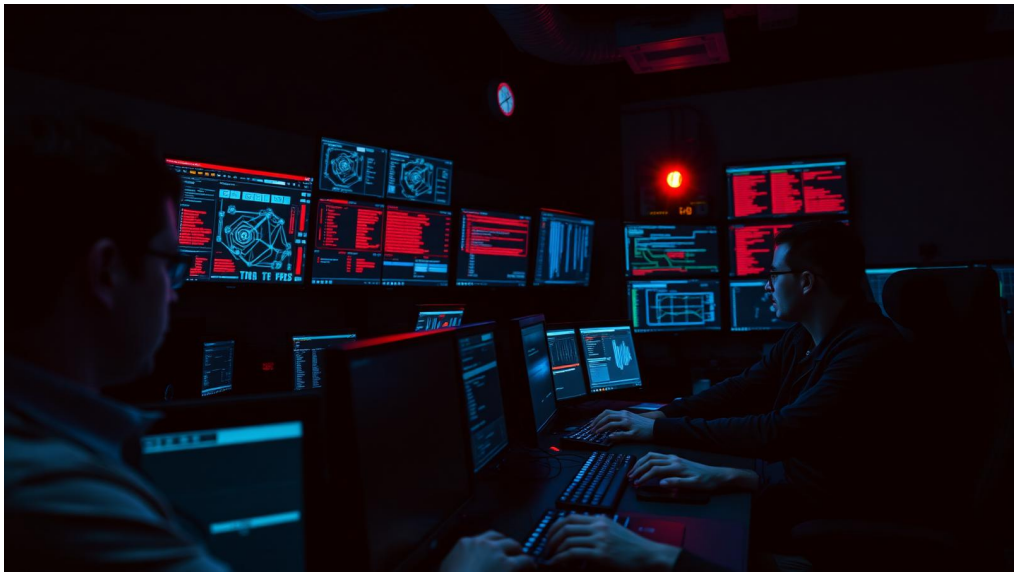
1. outage · downtime · degradation · runbook

2. breach · vulnerability · exploit · timeline

3. acknowledge · trigger · page · roll back

4. MFA · credential stuffing · patch · blameless

Ask the teacher — notes to bring to class



Grammar focus: The passive (all forms) & have something done

1. The passive across tenses — one pattern, all times — Form: BE (in the right tense) + past participle. • Present simple: 'The database IS backed up every hour.' • Present continuous: 'The failover IS BEING tested right now.' • Present perfect: 'The alert HAS BEEN acknowledged.' • Past simple: 'A bad config WAS PUSHED at 14:02.' • Past continuous: 'The service WAS BEING restarted when the second alert fired.' • Past perfect: 'The health-check HAD BEEN removed by an earlier change.' • Future / future perfect: 'The patch WILL BE deployed tonight' / 'The root cause WILL HAVE BEEN identified by Friday.' • Modals: 'The failover SHOULD BE tested weekly.' / 'This COULD HAVE BEEN prevented.'

2. Why use the passive in incident writing? — Not to hide the actor — to keep the focus on the SYSTEM. 'A configuration change WAS PUSHED at 14:02 that removed the health-check.' = the interesting thing is the state of the system. 'Ravi pushed a bad config' = the interesting thing is Ravi. Emotionally satisfying, learning-destroying. When you DO want the actor, name a ROLE, not a person: 'by the on-call engineer', 'by the deploy pipeline', 'by an automated job'.

3. HAVE something DONE — the fix is owned, not necessarily by you — Form: HAVE + object + past participle. 'We'll HAVE the alert threshold RECONFIGURED by Wednesday.' — someone will do it; the point is that we're accountable, not that we personally will type the command. 'We HAD the pipeline REVIEWED by security last quarter.' — arranged, not done by us. 'We NEED TO HAVE the runbook UPDATED before we close this incident.' Difference from 'we'll reconfigure': 'have X done' signals ownership without claiming the keyboard work — right for a lead or a manager writing a report.

4. Common traps — • 'The alert was fire' ✗ → 'was FIRED' (past participle, not base form). • 'It was happened' ✗ → 'HAPPEN' is intransitive; there's no passive. Say 'it happened'. • 'We had reconfigured the alert' ≠ 'we had the alert reconfigured'. First = we did it (past perfect). Second = we arranged for someone else to do it ('have something done'). • Overusing the passive = mushy writing. Rule of thumb: use the passive when the SYSTEM is the story; use the active when a PERSON is the story ('The on-call engineer paged the SRE lead at 02:18').

Ask the teacher — notes to bring to class

Exercise A — Rewrite in the passive

Rewrite each sentence in the passive so the SYSTEM is the subject.

1. 'Someone pushed a bad configuration at 14:02.'

2. 'We are testing the failover in staging right now.'

3. 'An earlier change had removed the health-check.'

4. 'The team will identify the root cause by Friday.'

5. 'We should test the failover every week.'

6. 'We could have prevented this with a proper code review.'

Exercise B — 'Have something done'

Rewrite each sentence using 'have something done' — signalling ownership without claiming the keyboard work.

1. 'We're going to reconfigure the alert threshold by Wednesday.' (…but ops actually does it)

2. 'Security reviewed our pipeline last quarter.'

3. 'We need to update the runbook before closing this incident.'

4. 'Legal will draft the customer notification tonight.'

Exercise C — Rewrite blame as blameless

Rewrite each blame-loaded incident line as a blameless one using the passive and, where useful, a role instead of a name.

1. 'Ravi pushed a bad config and broke prod again.'

2. 'Ops never tested the failover and now we're paying for it.'

Ask the teacher — notes to bring to class

3. 'Whoever wrote this alert should be fired — it doesn't fire when the DB is down.'

Listening: three post-mortems for the same 47-minute outage

You're the Head of Engineering. Yesterday your payments API went down for 47 minutes at Portuguese peak lunchtime — €340k of transactions lost or delayed. Three of your engineering leads are writing up the post-mortem. You're listening to their walkthroughs to decide who owns the incident write-up going forward. You're looking for who can think in SYSTEMS, use the passive without hiding, and produce action items with owners and dates.

pre

Which speaker do you predict will produce the most useful post-mortem — and what makes a post-mortem useful?

Think of an outage YOU were part of. Who got blamed — and what was the actual root cause?

gist

1. Which lead's post-mortem gets adopted, and why?

2. Which lead's post-mortem is 'find the guilty person' dressed as engineering?

3. Which lead is technically correct but produces no action?

detail

1. What is Ines's ROOT CAUSE — and how deep does she go?

2. What THREE action items does Ines commit to — with owners and dates?

3. What does Diego propose as the action item, and why does it fail?

Ask the teacher — notes to bring to class

Third listen — post-mortem craft

Focus on HOW each lead thinks. Answer from what they DO with language and structure.

1. Ines uses the passive constantly ('was pushed', 'had been removed', 'will be reconfigured'). What work is the passive doing for her?

2. Diego's post-mortem has an emotional cost beyond the outage. What is it?

3. Kwame writes correct sentences and produces zero change. Why is 'we should be more careful' the most dangerous line in a post-mortem?

Language from the audio — post-mortem moves worth stealing

Match each phrase to what it DOES.

"A configuration change was pushed at 14:02 that removed the health-check." — passive keeps the SYSTEM as subject — the state of the system is what the reader has to fix

"The second-reviewer requirement had been waived for launch week." — past perfect passive — surfaces a pre-existing process gap, not a moment-of-outage mistake

"We'll have the runbook updated to require two reviewers, SRE lead, by Wednesday." — 'have something done' + named owner + named date = an action item with real weight

"Ravi pushed the config and broke prod again." — personal blame — stops learning at the first name, guarantees the pattern repeats

"More training for Ravi and a written warning." — wrong-layer fix — leaves the system unchanged; the next engineer under pressure will do the same

"We should probably be more careful with deploys." — vague action item — satisfies the ritual, imposes no cost, immunises the org against real change

Voice A (Diego): OK so, look — this is really on Ravi. He pushed the config at 14:02 without waiting for a second reviewer. Ravi's been here two years, he should have known better. Look, everyone was under pressure for the launch, sure, but you don't just push to prod on launch day without a review. So my action items are: written warning for Ravi, and more training for the whole team on deploy discipline. That should sort it.

Ask the teacher — notes to bring to class

Voice B (Ines): OK, timeline. At 14:02 a configuration change was pushed that removed the /healthz endpoint from the load balancer. At 14:06 the on-call engineer was paged. At 14:14 the health-check misconfiguration was identified and the change was rolled back. Service was fully restored at 14:49. Now — why. WHY 1: the change had been approved with only one reviewer. WHY 2: the second-reviewer requirement had been waived for launch week under time pressure — that had been agreed the previous Monday. WHY 3, and this is the important one: the runbook did not classify auth-adjacent changes as requiring two reviewers regardless of launch pressure. That's the real gap. So — three action items. One, we'll have the runbook updated to require two reviewers on any auth-adjacent change regardless of launch phase — SRE lead, by Wednesday. Two, we'll have the /healthz alerting reconfigured so a missing endpoint pages within 60 seconds — on-call rotation lead, by Friday. Three, we'll have a monthly failover drill scheduled into the calendar — Head of Platform, first drill by end of month. No individual names in the write-up; the point is the process.

Voice C (Kwame): So, timeline is basically what Ines said — the config was pushed at 14:02, the alert was acknowledged at 14:06, the fix went in around 14:14. Look, we should be more careful with deploys, honestly. We should probably test the failover more often. And, um, we should probably look at the runbook at some point too. Yeah. That's my report.

Speaking: solo role-play — 90-second incident walkthrough

Solo task. Pick ONE incident you were part of — an outage, a security scare, a bad deploy, a broken release, a missed cron. Record yourself (voice note, up to 90 seconds) walking through it. You MUST include: (1) a 3-line TIMELINE in the passive ('X was pushed at…, the alert was fired at…, the fix was deployed at…'), (2) at least THREE 'whys' deep — trigger, contributing factor, process gap, (3) name ROLES not people ('the on-call engineer', 'the deploy pipeline') — no first names in the incident write-up, (4) ONE 'have something done' action item with a named owner and a named date.

Scene: A real recent outage you were part of (recommended)

A: You debrief.

B: Imagined listener: your Head of Engineering, on the Monday morning after.

Scene: A security scare — phishing, leaked credential, close call

A: You debrief.

B: Imagined listener: your CISO — no drama, structural fix.

Scene: A bad release / broken deploy in a product you own

A: You debrief.

B: Imagined listener: the wider team in a Friday all-hands.

Ask the teacher — notes to bring to class

Extra speaking task — 30-second 'the third why'

Compress the whole post-mortem to ONE sentence — the PROCESS-level gap that made the incident possible. Record a 30-second voice note. Format: 'The trigger was ___, but the real cause was that the ___ had been ___ — so we'll have ___ done by ___.'

Writing mini-task

Write the 4-sentence post-mortem summary (max 80 words). Sentence 1: the timeline in the passive — trigger + detection + fix. Sentence 2: the root cause, going at least three whys deep. Sentence 3: ONE 'have something done' action item with named owner and date. Sentence 4: what you'll change in the RUNBOOK — the process layer, not the person layer.

Wrap-up

One passive line from today you'll use in your next incident write-up.

One 'have something done' action item you'll assign this week.

One name you WON'T put in a post-mortem — and the role you'll use instead.

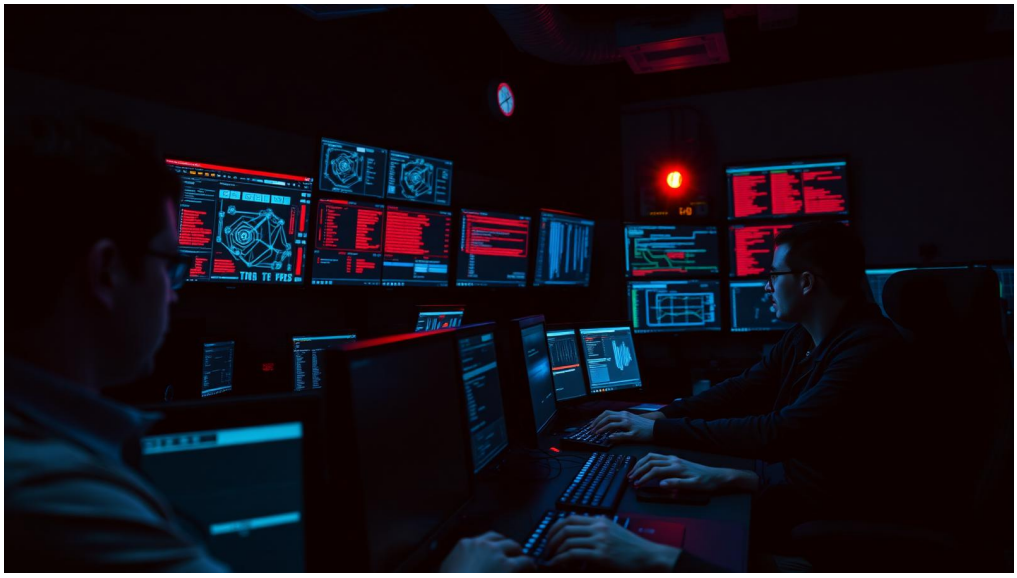
Quiz

1. The point of a BLAMELESS post-mortem is…
   (A) To protect senior engineers from consequences
   (B) To keep the focus on the system so the real chain of causes surfaces
   (C) To avoid writing a report
2. 'A configuration change was pushed at 14:02' uses the passive because…
   (A) English requires it
   (B) The interesting thing is the state of the system, not who was at the keyboard
   (C) It's more polite than naming Ravi
3. 'We'll have the runbook updated by Wednesday' is an example of…
   (A) A simple future passive
   (B) 'Have something done' — signals ownership without claiming the keyboard work
   (C) A past perfect
4. A useful post-mortem produces…
   (A) A named villain
   (B) At least three specific action items with owners and dates
   (C) A written warning
5. 'We should be more careful with deploys' is a BAD action item because…
   (A) It's grammatically wrong
   (B) It has no owner and no date — nothing to check on Friday
   (C) It uses the passive

Ask the teacher — notes to bring to class

Ask the teacher — notes to bring to class

Workbook — home practice



Exercise 1 — Incident vocabulary

Complete each sentence with ONE phrase from the box.

Word bank: outage · root cause · blast radius · runbook · on-call engineer · MFA

1. The payments API was rendered wholly unavailable for 47 minutes — a total ___.

2. The ___ ___ was a missing null-check in the auth service — but that was only the trigger.

3. The ___ ___ was negligible: only Portuguese mobile users were affected.

4. The alert fired at 02:14 and the ___-___ ___ acknowledged it in 4 minutes.

5. We're rolling out ___ across all admin accounts — no more password-only logins for anything sensitive.

6. The ___ was updated on Monday to require two reviewers on any auth-adjacent change.

Ask the teacher — notes to bring to class

Exercise 1b — Vocabulary — discursive writing

Write a formal incident summary (120–180 words) for the CTO regarding a recent system failure. Detail the causes and the remediation plan. Incorporate at least four phrases from Exercise 1: outage, root cause, blast radius, runbook, on-call engineer, or MFA.

type: writing

Use at least 4 target vocabulary items from the unit.

Maintain a formal, technical-executive register.

Use the passive voice for blameless reporting.

Structure clearly: Incident Overview, Root Cause Analysis, Corrective Actions.

model: Subject: Incident Report — Payments API Interruption To the CTO, This memorandum outlines our response to the 47-minute outage affecting the payments API on Tuesday. While the immediate trigger was a configuration push, the root cause was a deeper failure in our deployment runbook, which allowed a change to proceed without sufficient secondary review. Fortunately, the blast radius was limited to users in the EMEA region, but the incident highlighted a critical vulnerability in our recovery speed. The on-call engineer responded within four minutes, yet the rollback was delayed by manual verification steps. Moving forward, we will have our automated failover scripts audited and require MFA for all production environment access to prevent unauthorised changes. By hardening these processes, we aim to eliminate this class of failure entirely and ensure high availability for our customers.

Exercise 2 — The passive across tenses

Rewrite each sentence in the passive so the SYSTEM is the subject.

1. 'Someone pushed a bad configuration at 14:02.'

2. 'We are testing the failover in staging right now.'

3. 'An earlier change had removed the health-check.'

4. 'The team will identify the root cause by Friday.'

5. 'We should test the failover every week.'

6. 'We could have prevented this with a proper code review.'

Exercise 3 — Have something done

Rewrite each sentence using 'have something done' to signal ownership without claiming the keyboard work.

Ask the teacher — notes to bring to class

1. 'We're going to reconfigure the alert threshold by Wednesday.' (…but ops actually does it)

2. 'Security reviewed our pipeline last quarter.'

3. 'We need to update the runbook before closing this incident.'

4. 'Legal will draft the customer notification tonight.'

Exercise 4 — Blameless or blame-loaded?

Decide whether each incident line is BLAMELESS (describes the system) or BLAME-LOADED (names a villain).

1. 'A configuration change was pushed at 14:02 that removed the health-check.'
   (A) BLAMELESS
   (B) BLAME-LOADED
2. 'Ravi broke the auth service again.'
   (A) BLAMELESS
   (B) BLAME-LOADED
3. 'The failover was never tested in production — we caught that in the post-mortem.'
   (A) BLAMELESS
   (B) BLAME-LOADED
4. 'This wouldn't have happened if ops actually cared.'
   (A) BLAMELESS
   (B) BLAME-LOADED
5. 'The second-reviewer requirement had been waived for launch week.'
   (A) BLAMELESS
   (B) BLAME-LOADED
6. 'Whoever wrote this alert should be fired.'
   (A) BLAMELESS
   (B) BLAME-LOADED

Exercise 5 — Rewrite blame as blameless

Rewrite each blame-loaded line as a blameless incident sentence using the passive and, where useful, a role instead of a name.

Ask the teacher — notes to bring to class

1. 'Ravi pushed a bad config and broke prod again.'

2. 'Ops never tested the failover and now we're paying for it.'

3. 'Whoever wrote this alert should be fired — it doesn't fire when the DB is down.'

Exercise 6 — Three post-mortems for the same 47-min outage

You'll hear three engineering leads walk through the same outage. Decide who should own the incident write-up going forward.

1. Which lead's post-mortem is adopted, and why?

2. Which lead's post-mortem is 'find the guilty person' dressed as engineering?

3. Which lead is technically correct and produces zero change?

Voice A (Diego): This is really on Ravi. He pushed the config at 14:02 without waiting for a second reviewer. Everyone was under launch pressure, sure, but you don't push to prod on launch day without a review. Action items: written warning for Ravi, more training for the whole team.

Voice B (Ines): Timeline. At 14:02 a configuration change was pushed that removed the /healthz endpoint. At 14:06 the on-call engineer was paged. At 14:14 the change was rolled back. Service was fully restored at 14:49. Why. WHY 1: the change had been approved with only one reviewer. WHY 2: the second-reviewer requirement had been waived for launch week. WHY 3: the runbook did not classify auth-adjacent changes as always requiring two reviewers. Three action items. One, we'll have the runbook updated — SRE lead, by Wednesday. Two, we'll have the /healthz alerting reconfigured to page within 60 seconds — on-call lead, by Friday. Three, we'll have a monthly failover drill scheduled — Head of Platform, first drill by end of month.

Voice C (Kwame): Timeline is what Ines said. We should be more careful with deploys, honestly. We should probably test the failover more often. And, um, we should look at the runbook at some point too. Yeah.

Ask the teacher — notes to bring to class

Exercise 7 — Write your 4-sentence post-mortem

Pick a real incident from the last 6 months. Write the 4-sentence summary (max 80 words). Sentence 1: timeline in the passive. Sentence 2: root cause, at least three whys deep. Sentence 3: ONE 'have something done' action item with owner and date. Sentence 4: what changes in the RUNBOOK, not what a person will 'try to do better'.

Ask the teacher — notes to bring to class

Answer key — Class Book

Lead-in: incident or drama?

1. BLAMELESS — describes the system's state, no name attached.
2. BLAME-LOADED — names a person, no cause, no fix.
3. BLAMELESS — points at the process gap.
4. BLAME-LOADED — attacks a team, teaches nothing.

Reading: The blameless post-mortem

1. (B) To keep the focus on the system, not the person, so the real chain of causes surfaces
2. (B) The interesting thing is what happened to the system, not who happened to be at the keyboard
3. (B) Three: What happened, Why, and What we're changing
4. (B) 'Have something done' — the fix is owned, but not necessarily by the writer

Vocabulary: incidents, security and post-mortem language

1. outage / downtime
2. root cause / contributing factor
3. credential stuffing
4. blast radius
5. MFA
6. on-call engineer

1. The root cause was a database deadlock that wasn't caught in staging.
2. The on-call engineer was paged at 3 AM but didn't respond for ten minutes.
3. We need to trigger the failover manually to verify it works.
4. The outage lasted exactly 47 minutes during peak traffic.
5. The report must be blameless to ensure we actually learn from this.

1. The trigger was a missing null-check in the auth service, pushed at 14:02.
2. The blast radius was limited to Portuguese mobile users over a 12-minute window.
3. We'll have the runbook updated by Wednesday, owned by the SRE lead.
4. The on-call engineer was paged at 14:06 and acknowledged the alert within 4 minutes.

1. runbook — the other three describe the state of the service; a runbook is a document.
2. timeline — the other three are security terms; a timeline belongs to post-mortem structure.
3. trigger — the other three are actions the on-call takes; a trigger is what STARTED the incident.
4. blameless — the other three are technical/security terms; blameless describes the tone of the write-up.

Grammar focus: The passive (all forms) & have something done

1. 'A bad configuration WAS PUSHED at 14:02.' (Or: '…was pushed at 14:02 by the deploy pipeline.')
2. 'The failover IS BEING TESTED in staging right now.'
3. 'The health-check HAD BEEN REMOVED by an earlier change.'
4. 'The root cause WILL HAVE BEEN IDENTIFIED by Friday.' (Or: 'will be identified by Friday.')
5. 'The failover SHOULD BE TESTED every week.'
6. 'This COULD HAVE BEEN PREVENTED with a proper code review.'

1. 'We'll HAVE the alert threshold RECONFIGURED by Wednesday.'
2. 'We HAD our pipeline REVIEWED by security last quarter.'
3. 'We need to HAVE the runbook UPDATED before we close this incident.'
4. 'We'll HAVE the customer notification DRAFTED by legal tonight.'

1. Model: 'A bad configuration WAS PUSHED to production at 14:02, removing the health-check. The change HAD BEEN APPROVED without the standard second reviewer under launch-week pressure — a contributing factor we're addressing in the runbook.'
2. Model: 'The failover HAD NOT BEEN TESTED in production for six months. We'll HAVE weekly failover drills SCHEDULED starting next Monday, owned by the SRE lead.'
3. Model: 'The alert AS CONFIGURED does not fire on complete DB unavailability — only on high latency. This CONTRIBUTED to the 12-minute detection delay. We'll HAVE the alert threshold RECONFIGURED by Wednesday, owned by the on-call rotation lead.'

Ask the teacher — notes to bring to class

Listening: three post-mortems for the same 47-minute outage

1. Voice B (Ines). Uses the passive to keep the focus on the system ('a configuration change was pushed', 'the health-check had been removed'), digs three whys deep (skipped review → launch pressure → no second-reviewer requirement in the runbook), names ROLES not people, and produces three action items with owners and dates.
2. Voice A (Diego). Names Ravi four times ('Ravi pushed the config', 'Ravi should have known'), stops at the first WHY, and proposes 'more training for Ravi' as the action item. The system-level failure — no second-reviewer requirement — is never named. Two months from now the same thing happens with someone else.
3. Voice C (Kwame). Reads the timeline accurately, uses the passive well ('the alert was acknowledged at 14:06'), but the 'action items' are 'we should be more careful', 'we should test more', 'we should probably look at the runbook'. No owners, no dates, no fix. The document is a wake, not a post-mortem.
1. Immediate trigger: 'a configuration change was pushed at 14:02 that removed the /healthz endpoint from the load balancer'. WHY 1: 'the change had been approved with only one reviewer'. WHY 2: 'the second-reviewer requirement had been waived for launch week under time pressure'. WHY 3: 'the runbook did not classify auth-adjacent changes as requiring two reviewers regardless of launch pressure'. So the action is at the RUNBOOK level, not the reviewer level — that's what makes the post-mortem useful.
2. 1) 'We'll HAVE the runbook UPDATED to require two reviewers on any auth-adjacent change regardless of launch phase — SRE lead, by Wednesday.' 2) 'We'll HAVE the /healthz alerting RECONFIGURED so a missing endpoint pages within 60 seconds — on-call rotation lead, by Friday.' 3) 'We'll HAVE a monthly failover drill SCHEDULED into the calendar — Head of Platform, first drill by end of month.' All three: named change, named owner, named date.
3. 'More training for Ravi and a written warning.' It fails because it fixes the wrong layer — the system will still allow single-reviewer pushes of auth-adjacent changes under launch pressure. Two months later a different engineer, under a different launch pressure, does the same thing. The pattern repeats; the training is theatre.
1. It keeps the SYSTEM as the subject of every sentence in the timeline. The reader's attention stays on 'what state was the system in?' — which is where the fix has to happen. If Ines wrote 'Ravi pushed the config', the reader's brain would immediately want to know what to do about Ravi — a dead end. The passive is a form of thinking, not just a form of politeness.
2. It teaches the engineering team that when things go wrong, individuals get named and punished. Next time an engineer sees something odd right before a deploy, they'll STAY QUIET — because raising it might make them the story. Diego's post-mortem doesn't just fail to fix this outage; it manufactures the conditions for the next one.
3. Because it satisfies the ritual — the report exists, the meeting was held — without imposing any cost or commitment on anyone. There is nothing to check on Friday. Six months later the outage is 'a known thing that keeps happening', and the team stops taking post-mortems seriously. Vague action items are worse than none — they immunise the organisation against real ones.

Quiz

1. (B) To keep the focus on the system so the real chain of causes surfaces
2. (B) The interesting thing is the state of the system, not who was at the keyboard
3. (B) 'Have something done' — signals ownership without claiming the keyboard work
4. (B) At least three specific action items with owners and dates
5. (B) It has no owner and no date — nothing to check on Friday

Ask the teacher — notes to bring to class

Answer key — Workbook

Exercise 1 — Incident vocabulary

1. outage
2. root cause
3. blast radius
4. on-call engineer
5. MFA
6. runbook

Exercise 2 — The passive across tenses

1. 'A bad configuration WAS PUSHED at 14:02.'
2. 'The failover IS BEING TESTED in staging right now.'
3. 'The health-check HAD BEEN REMOVED by an earlier change.'
4. 'The root cause WILL BE IDENTIFIED by Friday.' (or 'will have been identified')
5. 'The failover SHOULD BE TESTED every week.'
6. 'This COULD HAVE BEEN PREVENTED with a proper code review.'

Exercise 3 — Have something done

1. 'We'll HAVE the alert threshold RECONFIGURED by Wednesday.'
2. 'We HAD our pipeline REVIEWED by security last quarter.'
3. 'We need to HAVE the runbook UPDATED before we close this incident.'
4. 'We'll HAVE the customer notification DRAFTED by legal tonight.'

Exercise 4 — Blameless or blame-loaded?

1. (A) BLAMELESS
2. (B) BLAME-LOADED
3. (A) BLAMELESS
4. (B) BLAME-LOADED
5. (A) BLAMELESS
6. (B) BLAME-LOADED

Exercise 5 — Rewrite blame as blameless

1. Model: 'A bad configuration WAS PUSHED to production at 14:02, removing the health-check. The change HAD BEEN APPROVED without the standard second reviewer under launch-week pressure.'
2. Model: 'The failover HAD NOT BEEN TESTED in production for six months. We'll HAVE weekly failover drills SCHEDULED starting next Monday, owned by the SRE lead.'
3. Model: 'The alert AS CONFIGURED does not fire on complete DB unavailability. We'll HAVE the alert threshold RECONFIGURED by Wednesday, owned by the on-call rotation lead.'

Exercise 6 — Three post-mortems for the same 47-min outage

1. Voice B (Ines) — passive keeps the SYSTEM as subject, digs three whys deep to a runbook-level gap, names ROLES not people, produces three action items with owners and dates ('we'll have X done by Wednesday, owned by SRE lead').
2. Voice A (Diego) — names Ravi four times, stops at the first WHY, proposes 'more training for Ravi' as the fix. The system-level gap (no second-reviewer requirement) is never named.
3. Voice C (Kwame) — accurate timeline, correct passive, but 'we should be more careful' has no owner, no date, no fix. The document is a wake, not a post-mortem.

Ask the teacher — notes to bring to class