Blameless Incident Resolution

Modern incident response challenges

As systems increase in complexity, resolving incidents becomes harder. Repetitive tasks take up precious time, and lapses in communication can create rifts in your process. You need to get everyone on the same page, and quickly.

Site Reliability Engineering can help

Site reliability engineering, or SRE, is an approach to creating more reliable systems. SRE encourages best practices for incident response. According to Google, these best practices include:
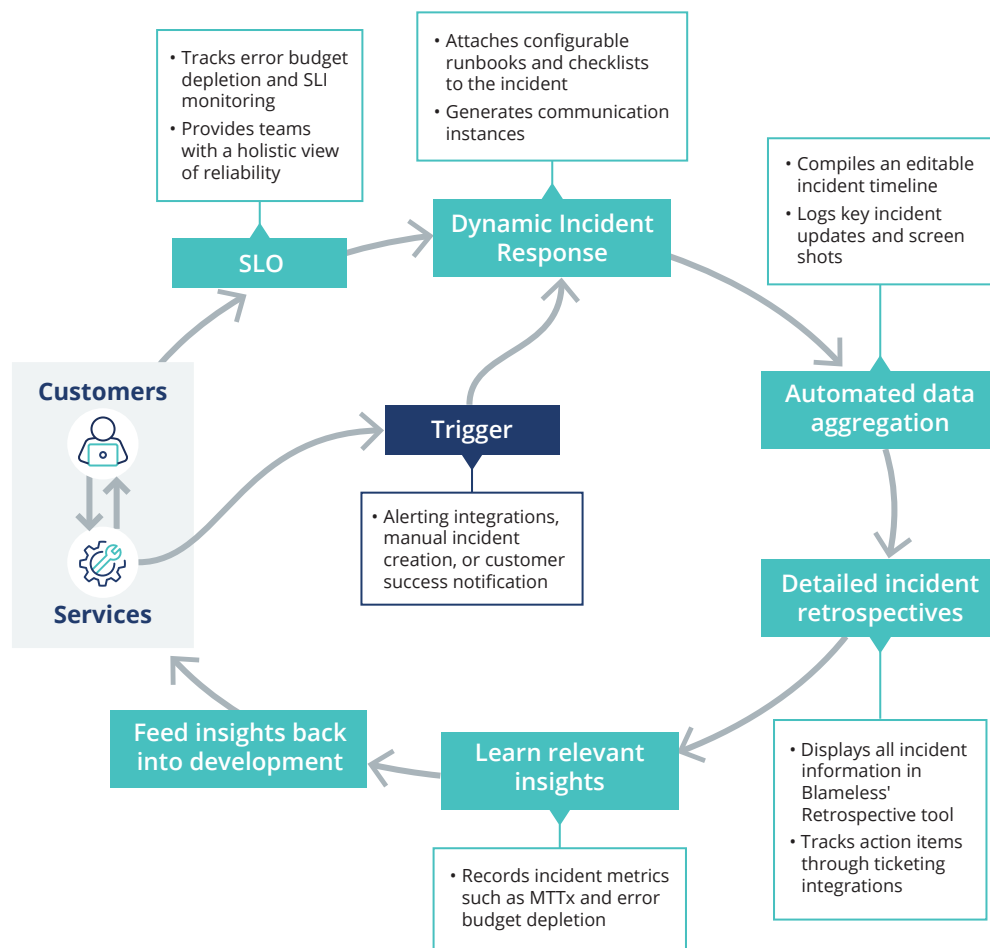
- Maintaining a clear line of command.
- Designating clearly defined roles.
- Keeping a working record of debugging and mitigation as you go.
- Declaring incidents early and often.

Where does Blameless fit in?

Understanding best practices is one thing, but operationalizing them is another. Blameless helps you:

- Automate manual, repetitive parts of the incident response process with our intelligent chatbot
- Centralize incident context and establish guardrails with customizable checklists and runbooks
- Capture details in real time to eliminate the toil of aggregating the right data and context after the incident

Blameless Reliability Lifecycle

- Tracks error budget depletion and SLI monitoring
- Provides teams with a holistic view of reliability

SLO

- Attaches configurable runbooks and checklists to the incident
- Generates communication instances

Dynamic Incident Response

- Compiles an editable incident timeline
- Logs key incident updates and screen shots

Customers

Services

Trigger

- Alerting integrations, manual incident creation, or customer success notification

Automated data aggregation

Detailed incident retrospectives

- Displays all incident information in Blameless' Retrospective tool
- Tracks action items through ticketing integrations

Feed insights back into development

Learn relevant insights

- Records incident metrics such as MTTx and error budget depletion

> "The improved coordination, follow-up tracking, and visibility help us actually address what caused the incident in the first place, and prevent it from happening again."
>
> -Staff Site Reliability Engineer Tenzin Wangdhen, Iterable