

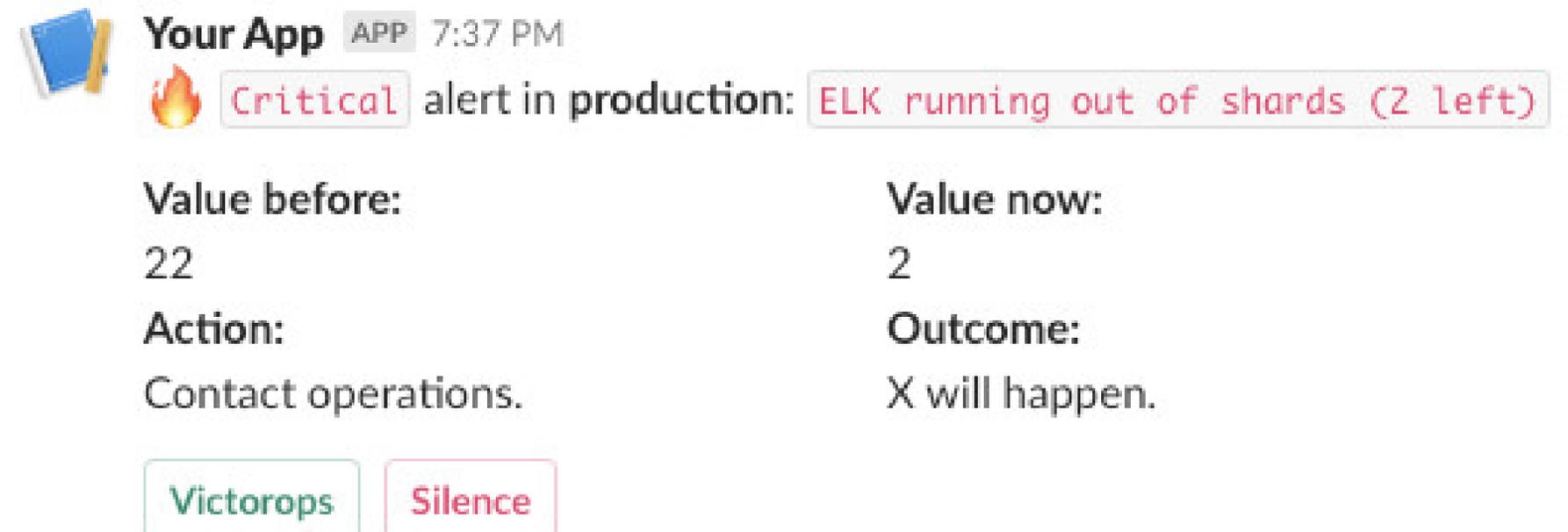
Почему моргают алерты как это исправить

Необходимые технические и организационные изменения



Что такое алерт

- Способ оповещения команды о проблемах
- ~~Способ мониторинга работы приложения~~



Your App APP 7:37 PM
Critical alert in production: **ELK running out of shards (2 left)**

Value before:
22

Action:
Contact operations.

Victorops Silence

Value now:
2

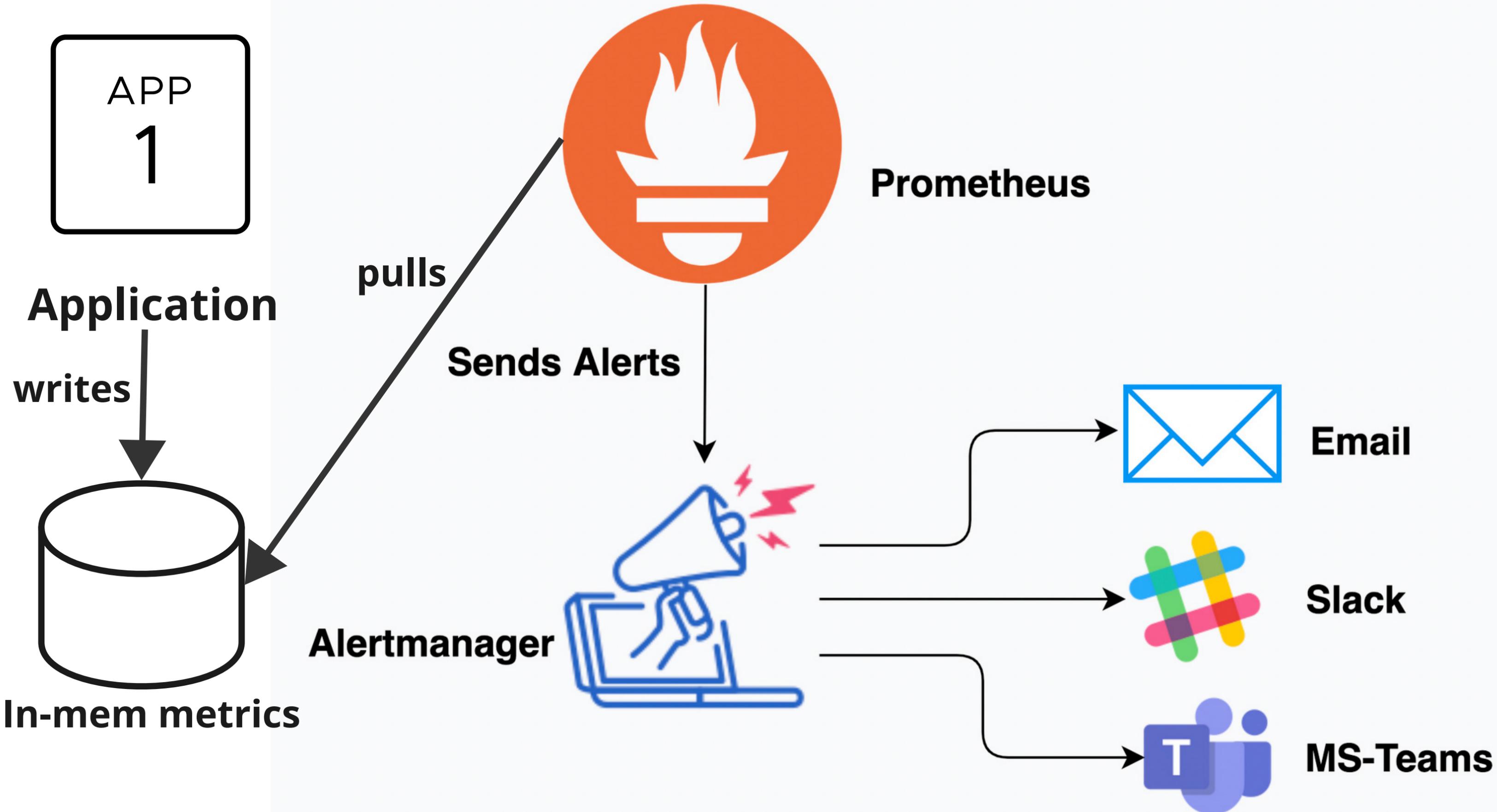
Outcome:
X will happen.

Проблема на проде сейчас

Большая проблема на проде скоро

время ответа 10сек

Кончается место в базе



С чего начинается проработка алерта:

- Поиск проблемной ситуации которую надо закрыть
- Проще всего это проводить разбор инцидентов или сделать ретроспективу и составить список того что могло помочь
- Составить список рисков для приложения, что может пойти не так
- Прочитать про 4 золотых метрики от google



Prometheus

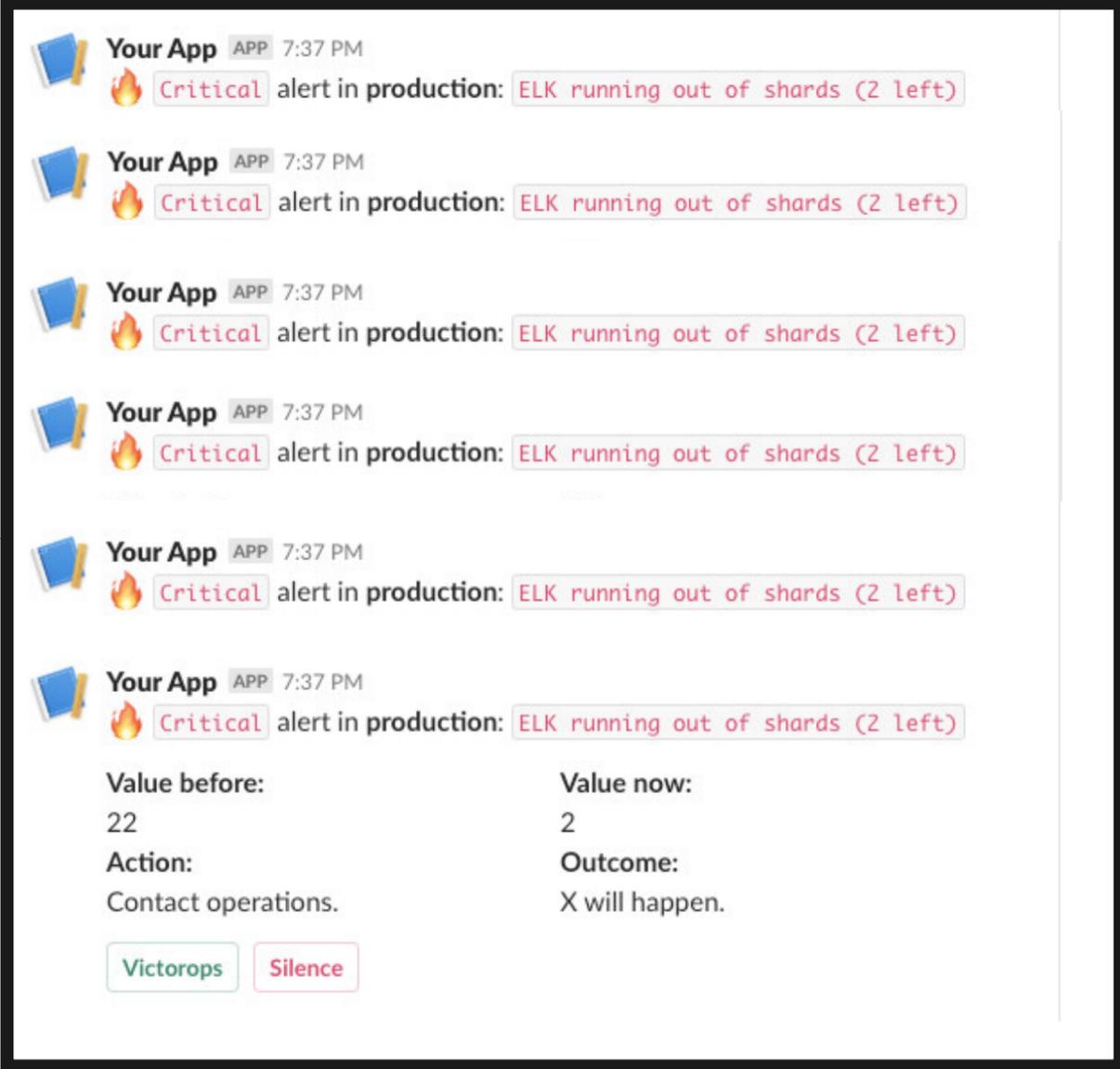


AlertManager

Казалось бы все настроили

На этом моменте
все статьи
кончаются

Так выглядит канал
с алертами каждые 5 минут



The screenshot shows a Slack channel with a repeating alert message. The message is a critical alert from 'Your App' at 7:37 PM, stating 'alert in production: ELK running out of shards (2 left)'. Below the message, there is a table with the following data:

Value before:	Value now:
22	2
Action:	Outcome:
Contact operations.	X will happen.

At the bottom of the alert, there are two buttons: 'Victorops' and 'Silence'.

На практике: жиза любой команды

Проблемы

- Микроинсульт у разработчиков на сообщения в канале
- Приходит много разных алертов, хотя вроде все работает и нет проблем
- Когда все не работает алерты тоже приходят и их игнорируют по привычке
- По алертам разработчики "убеждаются что код работает", оно же что-то пишет



Цель

Что должны получить

- Алерты сигнализируют о реальных проблемах
- Любой разработчик может решить проблему, есть инструкция что делать
- В логах релевантная информация
- Сентри не является помойкой



Практика: Пример алерта на использование CPU

```
- alert: HighCpuUsage
  expr: 100 - (avg(irate(node_cpu_seconds_total{mode="idle"}[1m])) * 100) > 95
  labels:
    severity: error
  annotations:
    summary: "High cpu usage in app"
    description: "High avg cpu usage"
```

Практика: 1. Этот алерт выстрелит слишком поздно

```
- alert: HighCpuUsage
  expr: 100 - (avg(irate(node_cpu_seconds_total{mode=
  "idle"}[1m])) * 100) > 95
  labels:
    severity: error
  annotations:
    summary: "High cpu usage in app"
    description: "High avg cpu usage"
```

Практика: 1. Этот алерт выстрелит слишком поздно

```
- alert: HighCpuUsage
  expr: 100 - (avg(irate(node_cpu_seconds_total{mode="idle"}[1m])) * 100) > 70
  labels:
    severity: error
  annotations:
    summary: "High cpu usage in app"
    description: "High avg cpu usage"
```

Увеличили границу, теперь что-то можно успеть сделать

Практика: 2. Этот алерт когда на всех серверах кончится CPU

```
- alert: HighCpuUsage
  expr: 100 - (avg(irate(node_cpu_seconds_total{mode=
  "idle"}[1m])) * 100) > 70
  labels:
    severity: error
  annotations:
    summary: "High cpu usage in app"
    description: "High avg cpu usage"
```

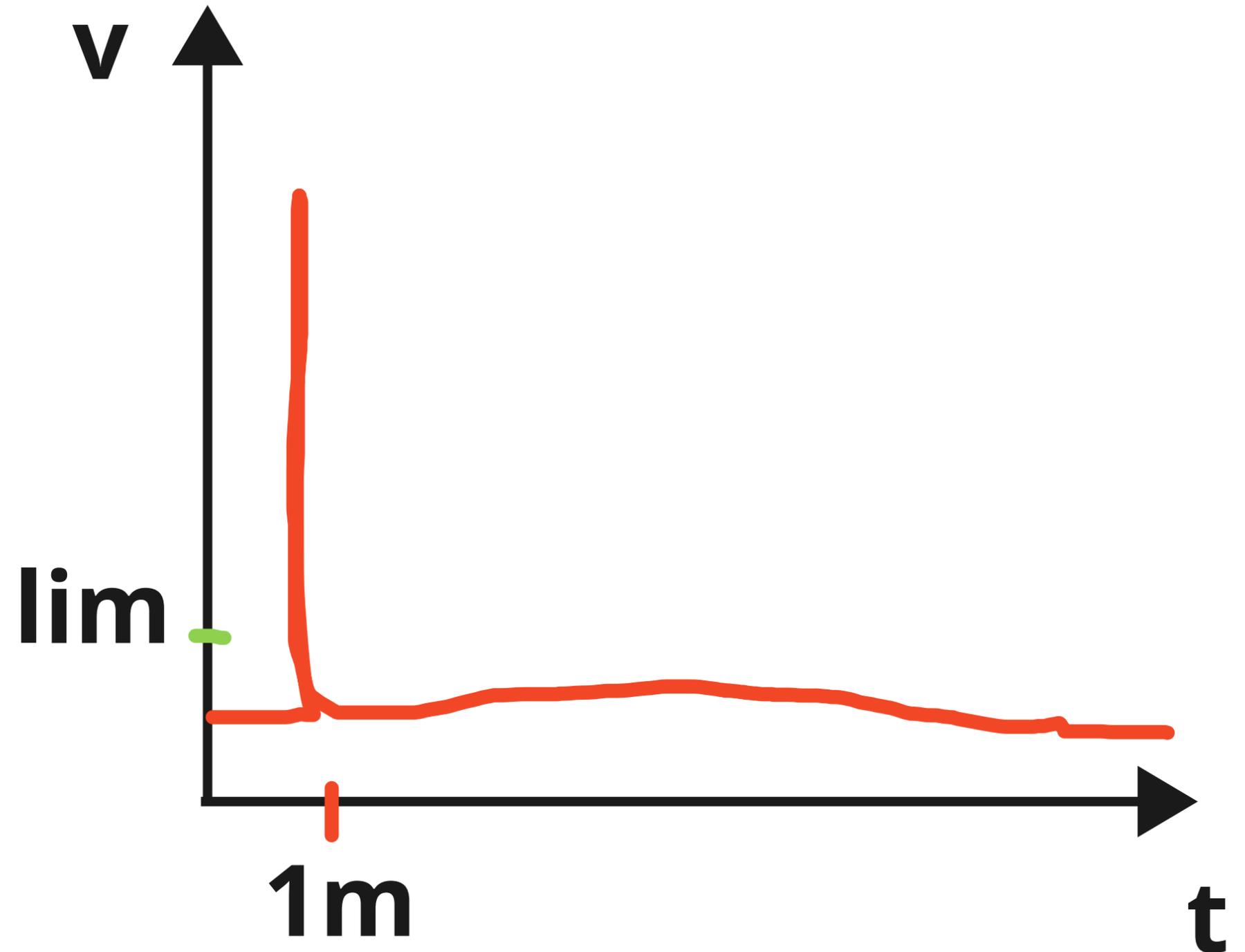
Практика: 2. Этот алерт когда на всех серверах кончится CPU

```
- alert: HighCpuUsage
  expr: (avg by (instance)
  (irate(node_cpu_seconds_total{mode="idle"}[1m])) * 100) >
  70
  labels:
    severity: error
  annotations:
    summary: "High cpu usage on {{ .instance }}"
    description: "Cpu usage on {{ .instance }} > 70%"
```

Теперь сработает когда первый сервер выпадет за границу
и можно успеть что-то сделать

Практика: 3. Этот алерт будет моргать на коротких пиках

- Алерт срабатывает
- Хотя деградация была менее 1m



Практика: 3. Этот алерт будет моргать на коротких пиках

```
- alert: HighCpuUsage
  expr: (avg by (instance)
    (irate(node_cpu_seconds_total{mode="idle"}[1m])) *
    100) > 70
  for: 3m
  labels:
    severity: error
  annotations:
    summary: "High cpu usage on {{ .instance }}"
    description: "Cpu usage on {{ .instance }} >
      70%"
```

Теперь условие должно выполняться минимум 3 минуты

Практика: 4. Что делать?

```
- alert: HighCpuUsage
  expr: (avg by (instance)
    (irate(node_cpu_seconds_total{mode="idle"}[1m])) *
    100) > 70
  for: 3m
  labels:
    severity: error
  annotations:
    summary: "High cpu usage on {{ .instance }}"
    description: "Cpu usage on {{ .instance }} >
70%"
```

Ну высокое и высокое, что алерты то слать

Практика: 4. Что делать?

Каждый алерт должен включать в себя информацию: (потенциально ссылка на confluence или другие системы)

- Кто и когда должен решать проблему
 - От команды в рабочее время (дежурный?)
 - От компании в ночные проблемы (инцидент менеджер?)
- Надо ли эскалировать?
 - Никто не ответил за 3-5 минут, не поставил реакции, пингуем лида?
- Что делать, базово?
 - Не перезапускать просто так, если проблема есть на кешах можно долго жить!
 - Посмотреть недавние релизы сервиса и релизы зависимостей
 - Ссылки куда еще посмотреть, другие графики в графанае, что там с трафиком, мы отвечаем?
 - Принимать решение об откате или редеплое?



Практика: 4. Что делать?

```
- alert: HighCpuUsage
  expr: (avg by (instance)
    (irate(node_cpu_seconds_total{mode="idle"}[1m])) *
    100) > 70
  for: 3m
  labels:
    severity: error
  annotations:
    summary: "Cpu usage on {{ .instance }} > 70%"
    description: "How to resolve <Playbook>
<grafana>"
```

Практика: 5. Не используйте irate

Используется для посекундного увеличения и берет только 2 самых недавних значения

```
- alert: HighCpuUsage
  expr: (avg by (instance)
  (irate(node_cpu_seconds_total{mode="idle"}[1m])) *
  100) > 70
  for: 3m
  labels:
    severity: error
  annotations:
    summary: "Cpu usage on {{ .instance }} > 70%"
    description: "How to resolve <Playbook>
    <grafana>"
```

<https://www.robustperception.io/avoid-irate-in-alerts/>

Практика: 5. Не используйте irate

Используются все значения диапазона

```
- alert: HighCpuUsage
  expr: (avg by (instance)
  (rate(node_cpu_seconds_total{mode="idle"}[1m])) *
  100) > 70
  for: 3m
  labels:
    severity: error
  annotations:
    summary: "Cpu usage on {{ .instance }} > 70%"
    description: "How to resolve <Playbook>
    <grafana>"
```

<https://www.robustperception.io/avoid-irate-in-alerts/>

Процесс работы с алертами

- Не забивать на прилетающие алерты, сразу заводить задачи на доработку
- Достичь "Zero alert policy", когда все хорошо алерты не должны приходить!
- Обновлять и поддерживать playbook, дополнять ссылки, на важные бизнес процессы добавлять алерты и инструкции что делать
- Превратить ненужные алерты и логи в метрики и графики

