

Комбо-факапы

Эффект бабочки-факапницы

Контур

Григорий Кошелев

Комбо-факапы

Комбо-факапы

Комбо-факап — факап, вызванный набором багов и событий

Комбо-факапы

Комбо-факап — факап, вызванный набором багов и событий, которые по отдельности не приводят к факапу

Комбо-факапы

Комбо-факап — факап, вызванный набором багов и событий, которые по отдельности не приводят к факапу или не наносят значительного ущерба

Эффект бабочки-факапницы



Эффект бабочки-факапницы



Эффект бабочки-факапницы



Зачем всё это



Комбо-факапы, или Эффект бабочки-факапницы

Контур

Григорий
Кошелев



БАЗА

БАЗА

Аксиома 1. Hardware ломается

БАЗА

Аксиома 1. Hardware ломается

Аксиома 2. ~~Peopleware~~ люди ошибаются

БАЗА

Аксиома 1. Hardware ломается

Аксиома 2. ~~Peopeware~~ люди ошибаются

Следствие 1. Software содержит ошибки

БАЗА

Аксиома 1. Hardware ломается

Аксиома 2. ~~Peop~~leware люди ошибаются

Следствие 1. Software содержит ошибки

Следствие 2. Невозможно избежать всех факапов

Не баг, а фича

Не баг, а фича

```
DateTime dt = DateTime.Parse(  
    "12/25/35",  
    CultureInfo.InvariantCulture);
```

Не баг, а фича

```
DateTime dt = DateTime.Parse(  
    "12/25/35",  
    CultureInfo.InvariantCulture);
```

.NET 7 – 25 декабря 1935

Не баг, а фича

```
DateTime dt = DateTime.Parse(  
    "12/25/35",  
    CultureInfo.InvariantCulture);
```

.NET 7 – 25 декабря 1935

.NET 8 – 25 декабря 2035

Не баг, а фича

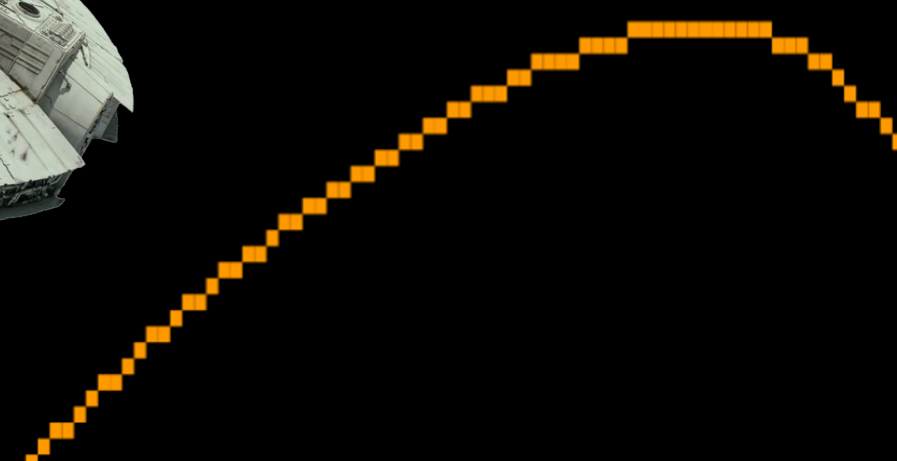
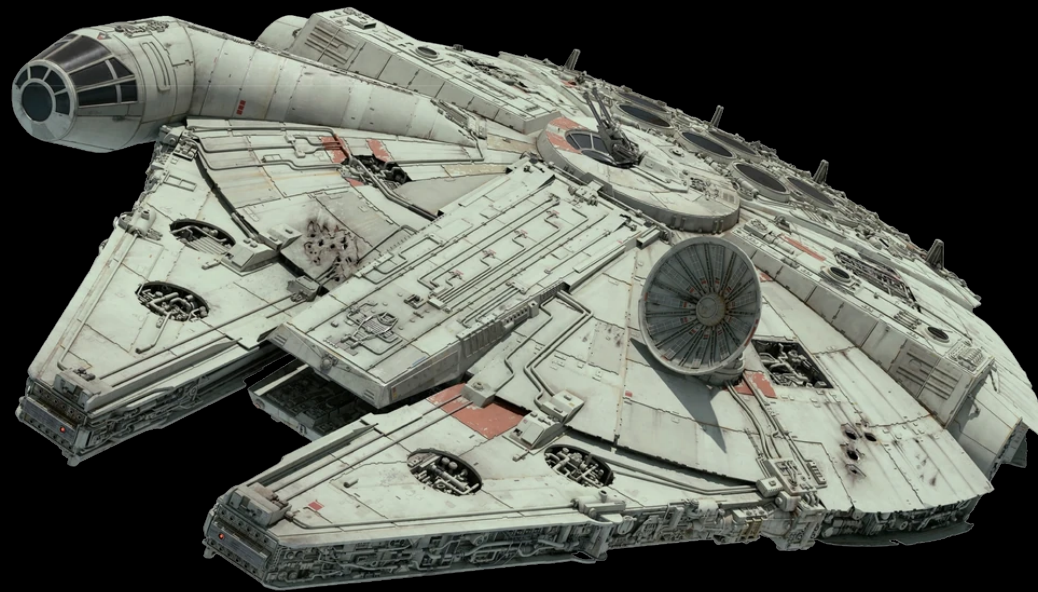
```
DateTime dt = DateTime.Parse(  
    "12/25/35",  
    CultureInfo.InvariantCulture);
```

.NET 7 – 25 декабря 1935

.NET 8 – 25 декабря 2035

```
GregorianCalendar.TwoDigitYearMax = 2029 → 2049;
```


CrowdStrike's Back



CrowdStrike's Back

19 июля 2024, 04:09 UTC — ?



CrowdStrike's Back

19 июля 2024, 04:09 UTC — ?

8.5 млн систем



CrowdStrike's Back

19 июля 2024, 04:09 UTC — ?

8.5 млн систем
10 млрд \$



CrowdStrike's Back

19 июля 2024, 04:09 UTC — ?

8.5 млн систем
10 млрд \$



CrowdStrike's Back

CrowdStrike Falcon

CrowdStrike's Back

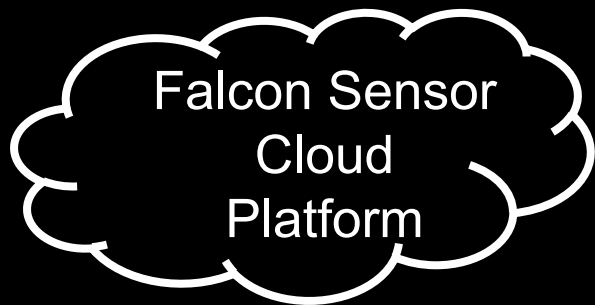
CrowdStrike Falcon

Falcon Local Sensor



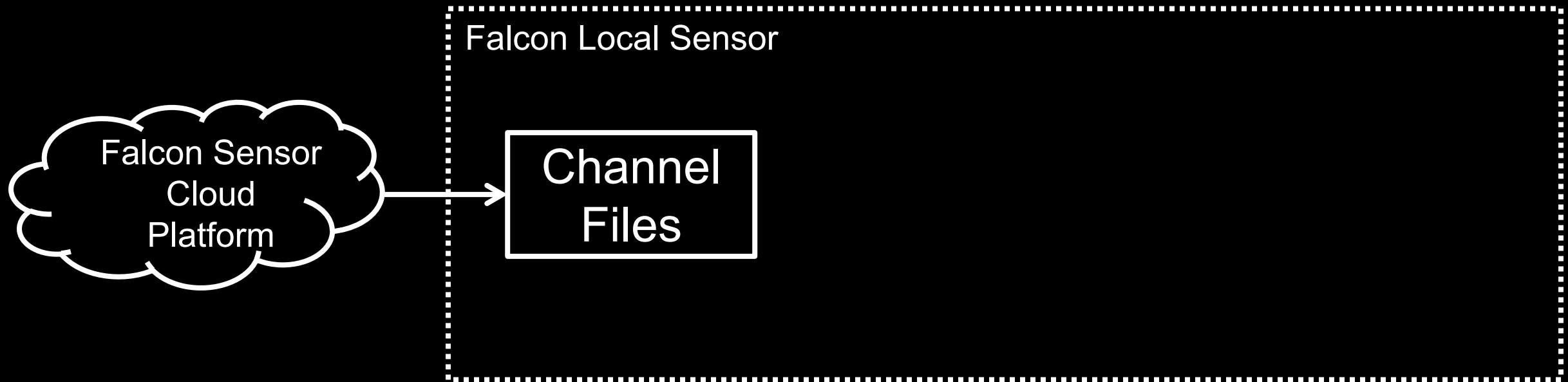
CrowdStrike's Back

CrowdStrike Falcon



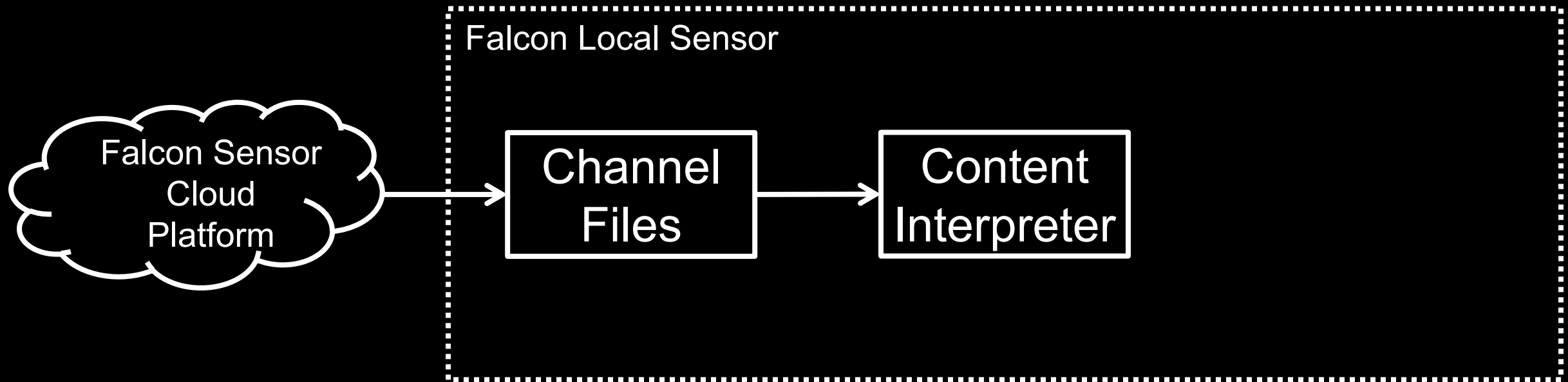
CrowdStrike's Back

CrowdStrike Falcon



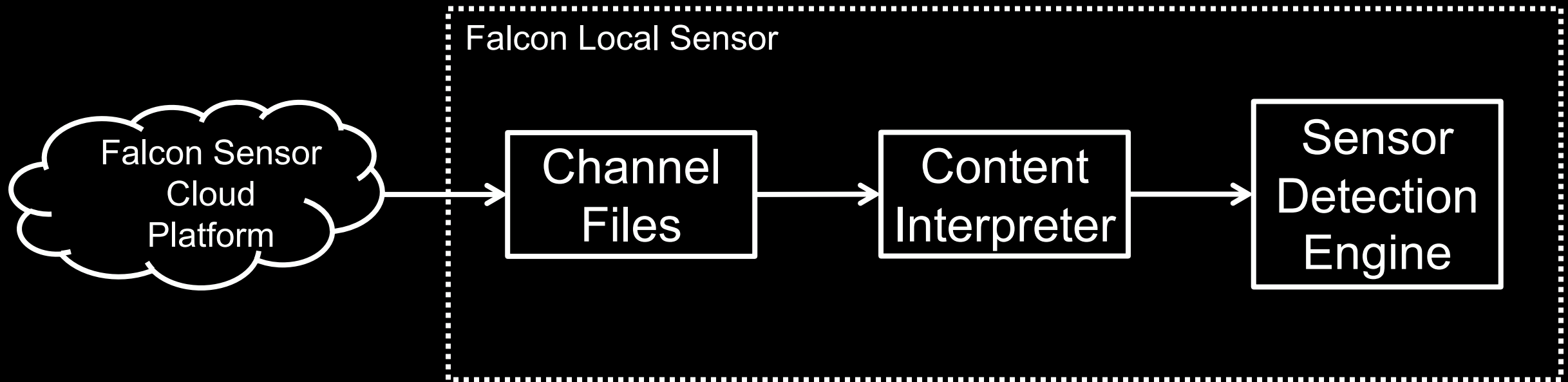
CrowdStrike's Back

CrowdStrike Falcon



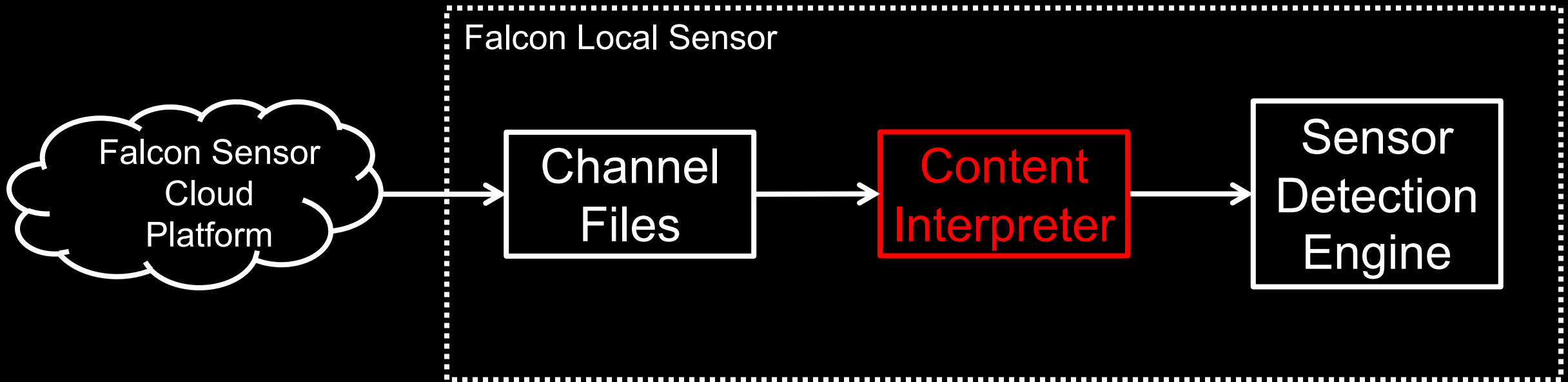
CrowdStrike's Back

CrowdStrike Falcon

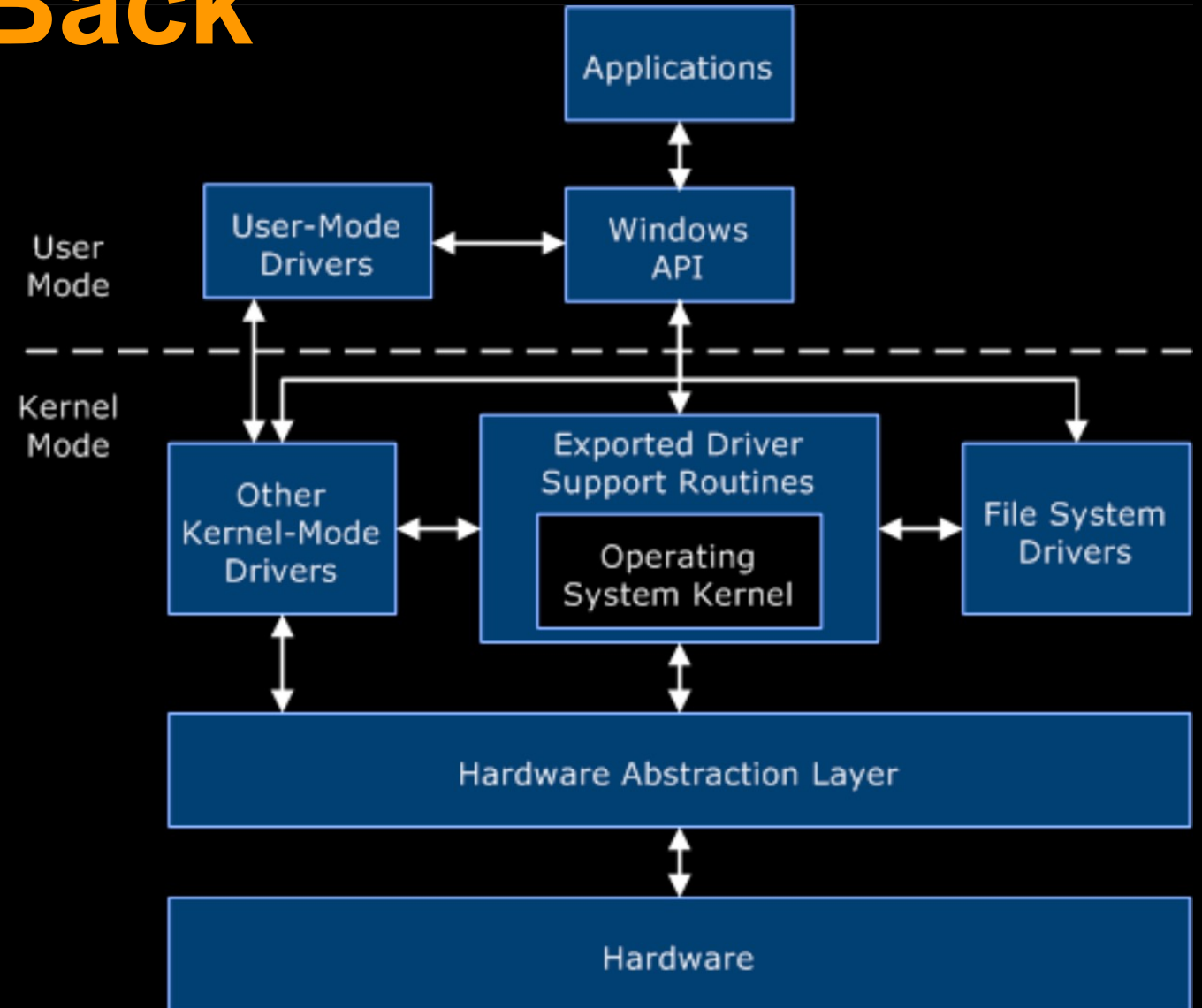


CrowdStrike's Back

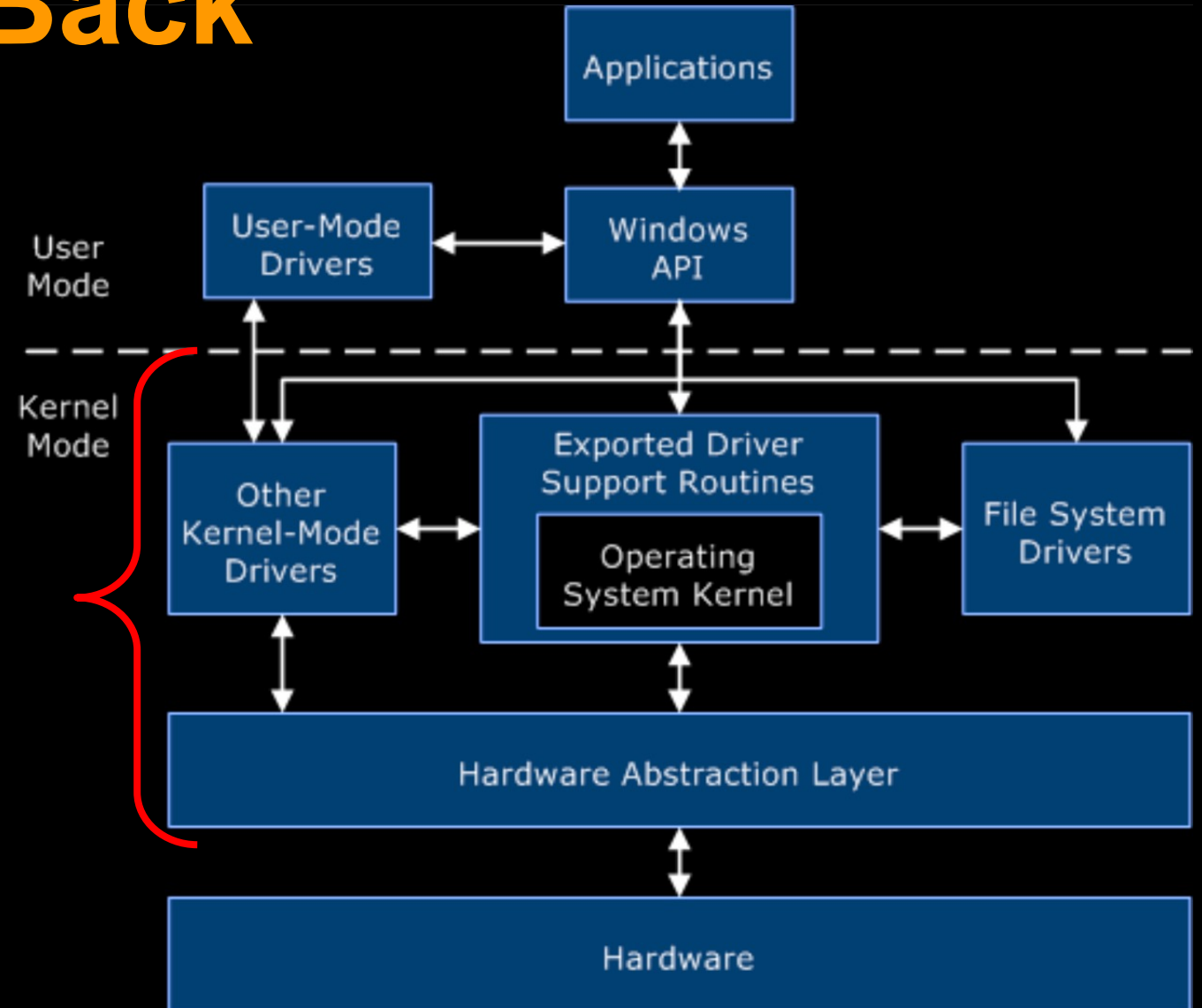
CrowdStrike Falcon



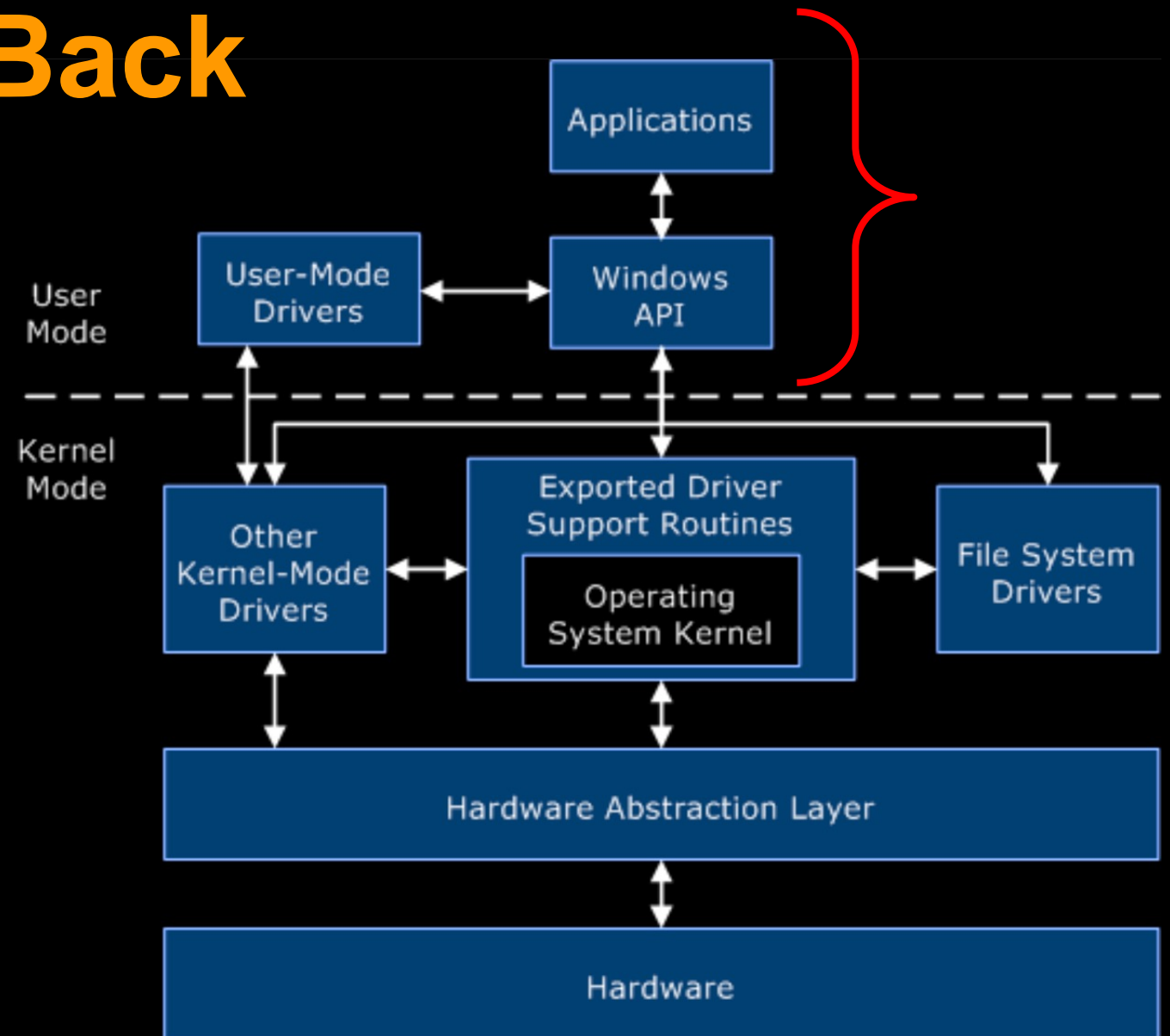
CrowdStrike's Back



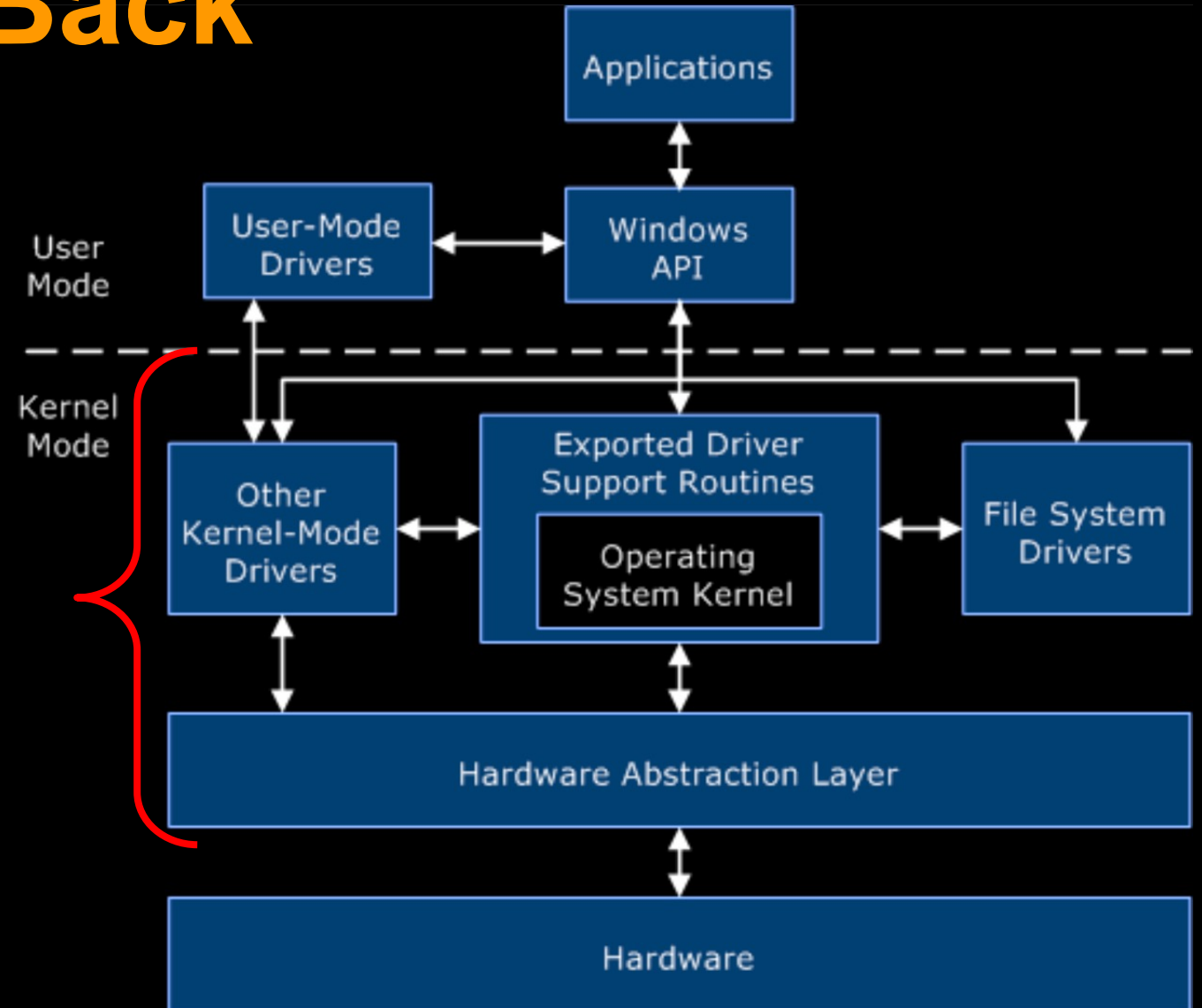
CrowdStrike's Back



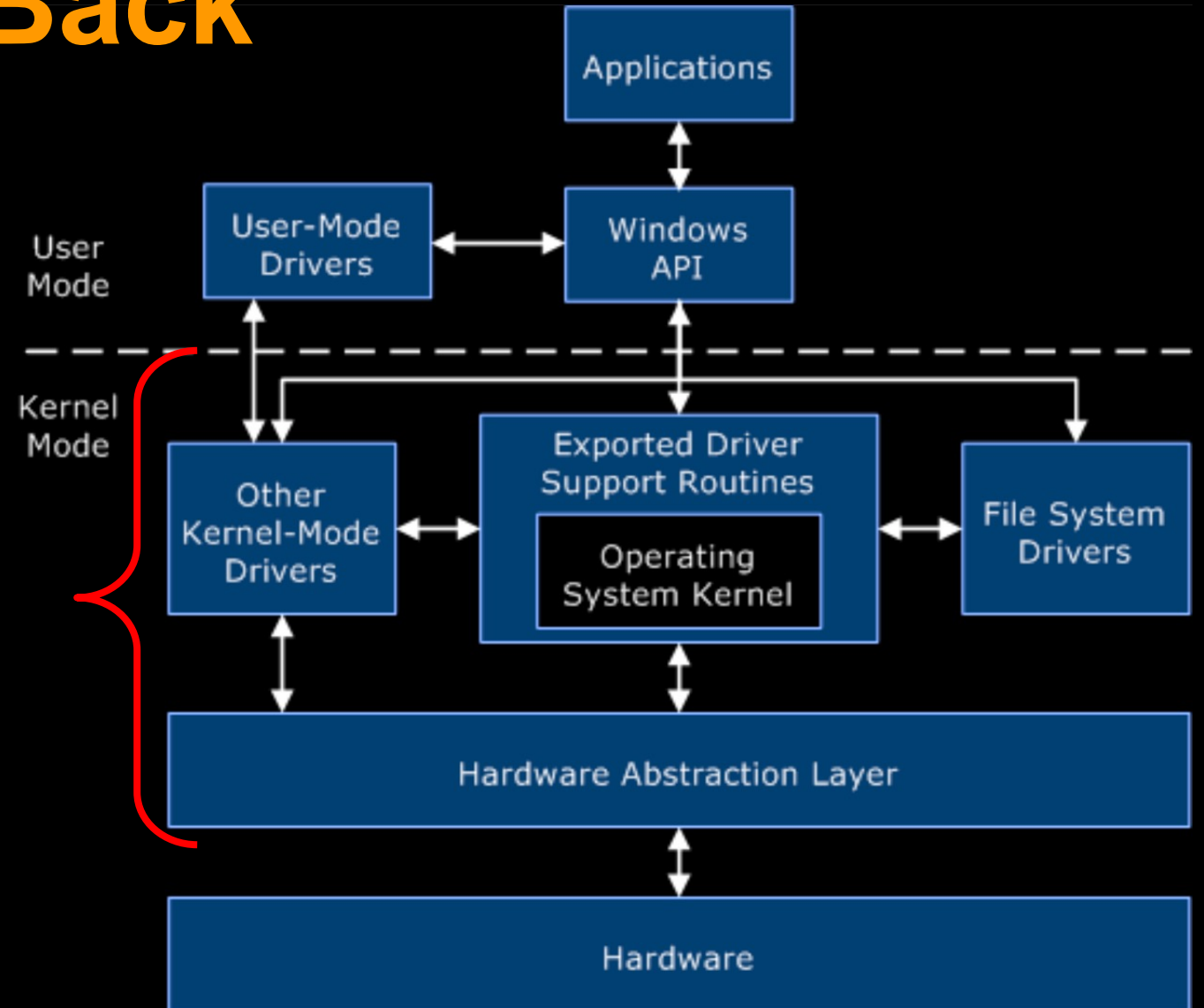
CrowdStrike's Back



CrowdStrike's Back

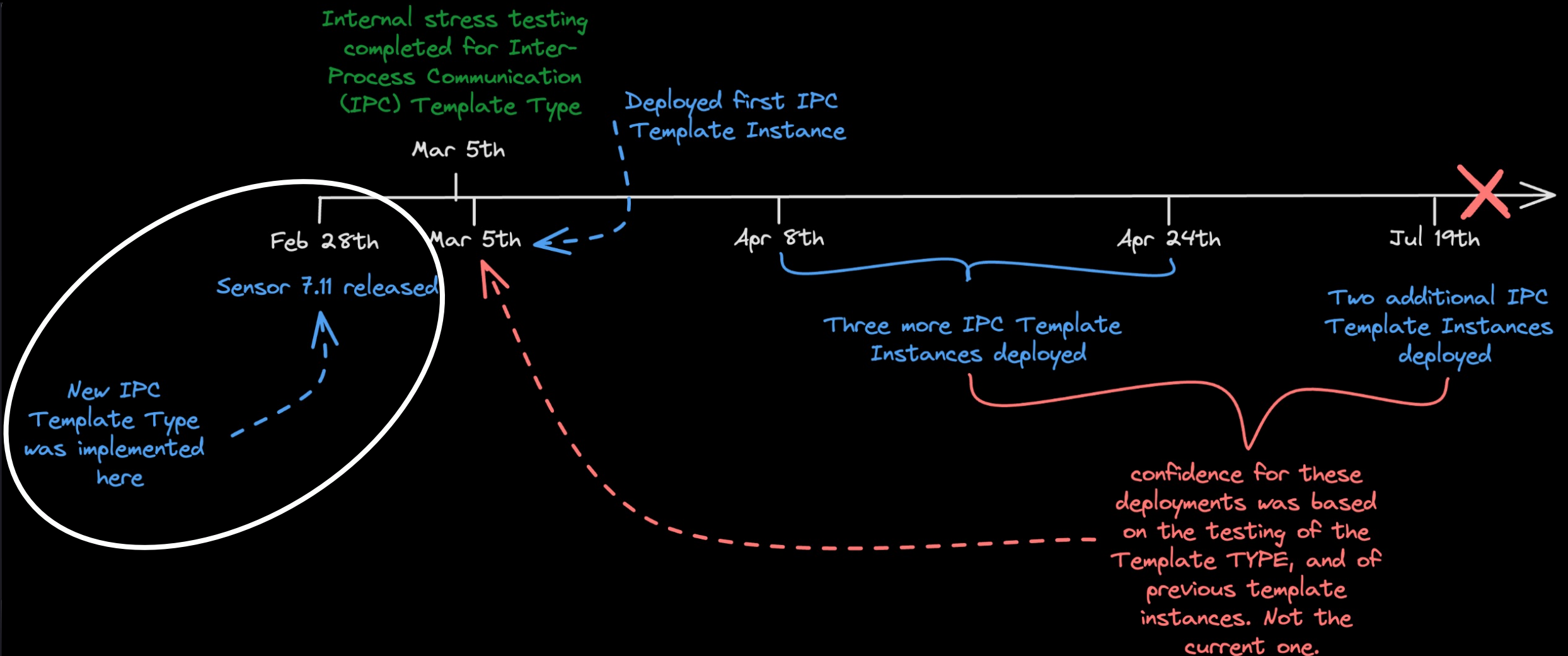


CrowdStrike's Back

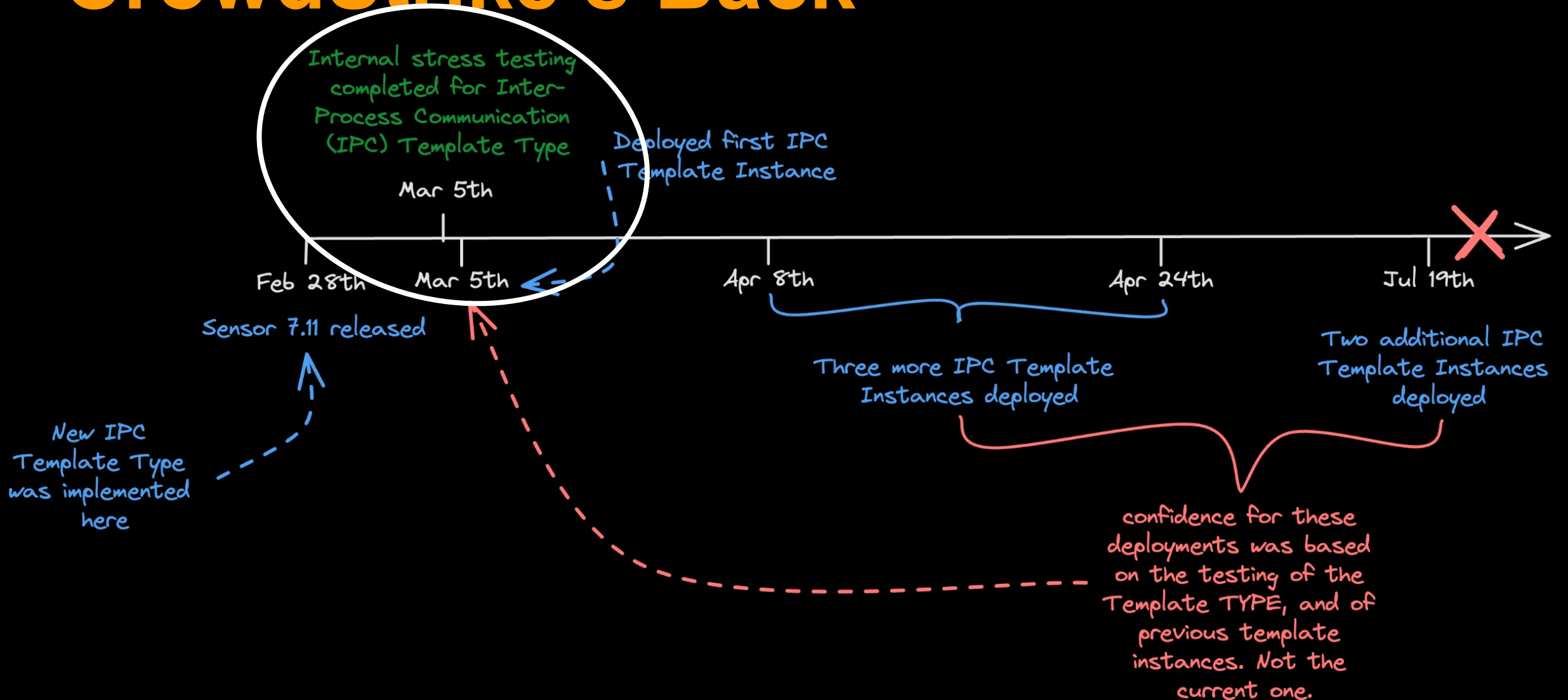


CrowdStrike's Back

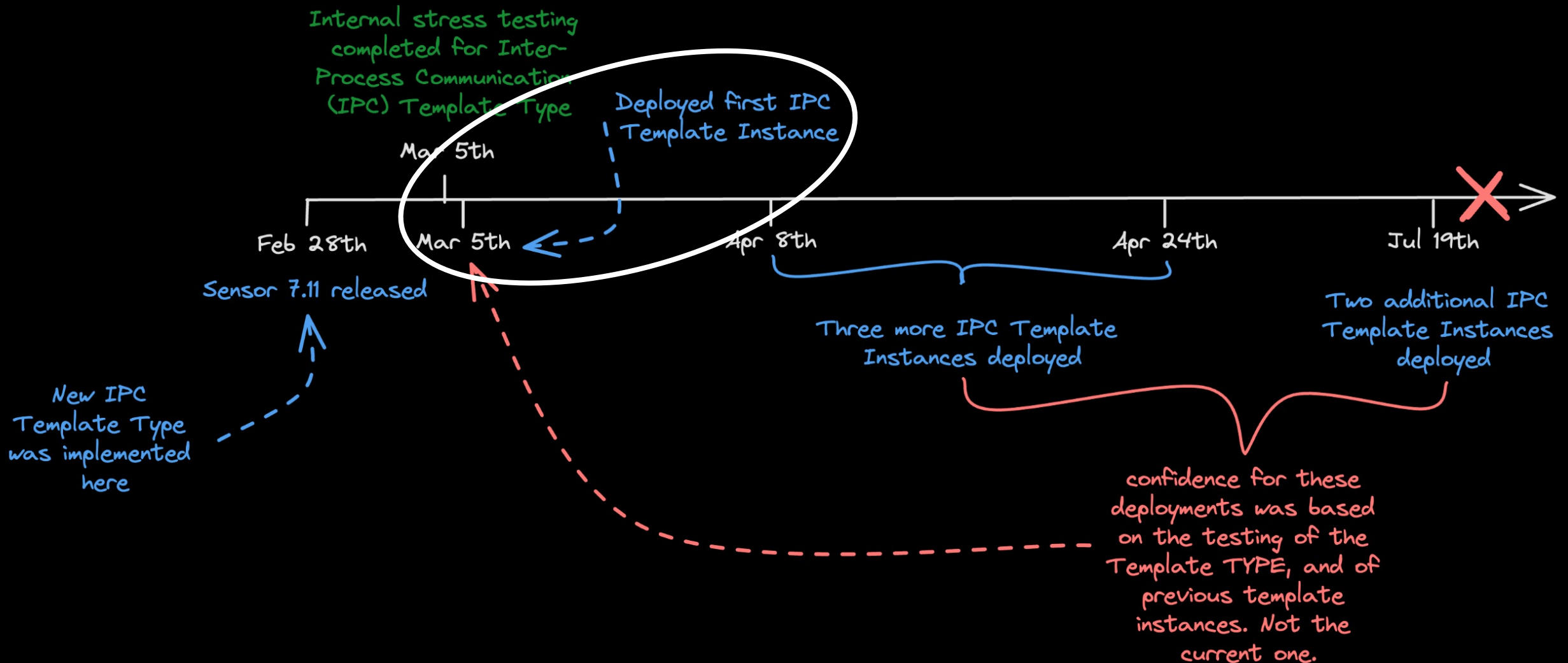
CrowdStrike's Back



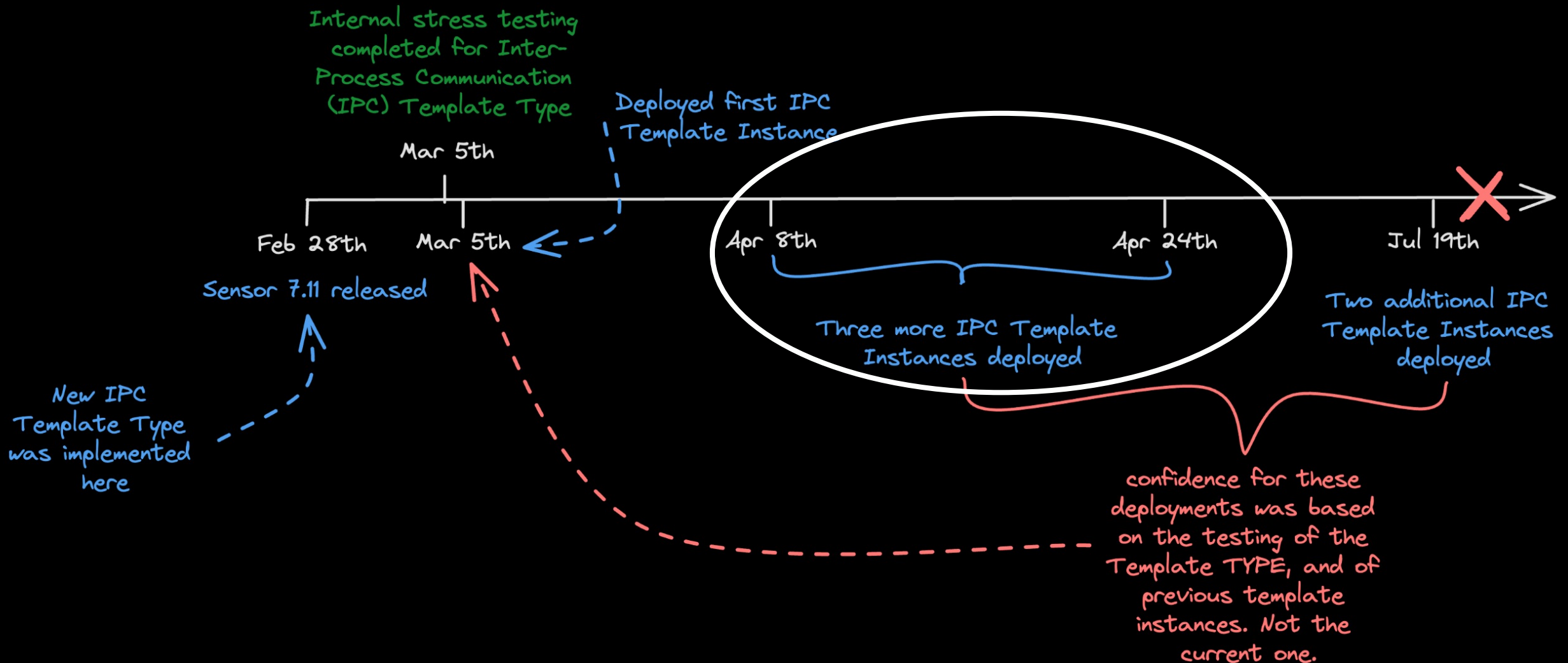
CrowdStrike's Back



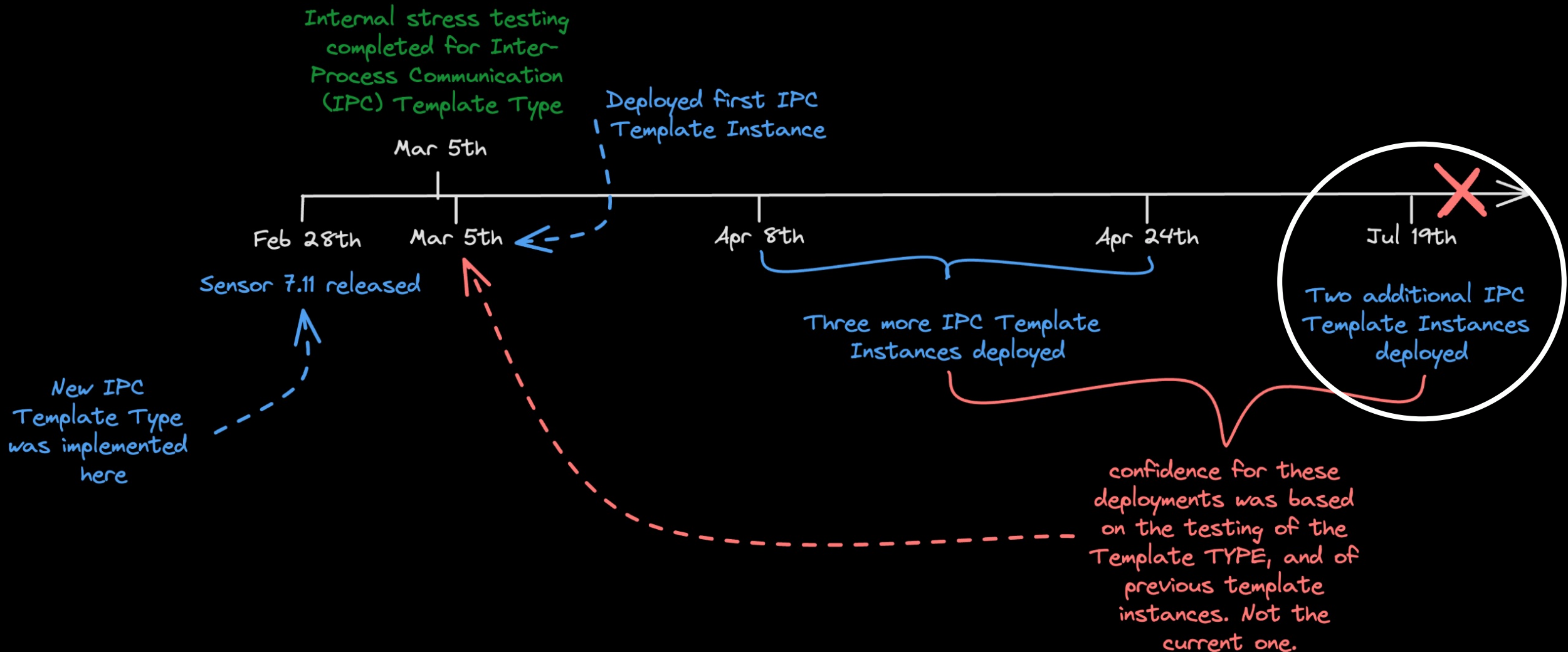
CrowdStrike's Back



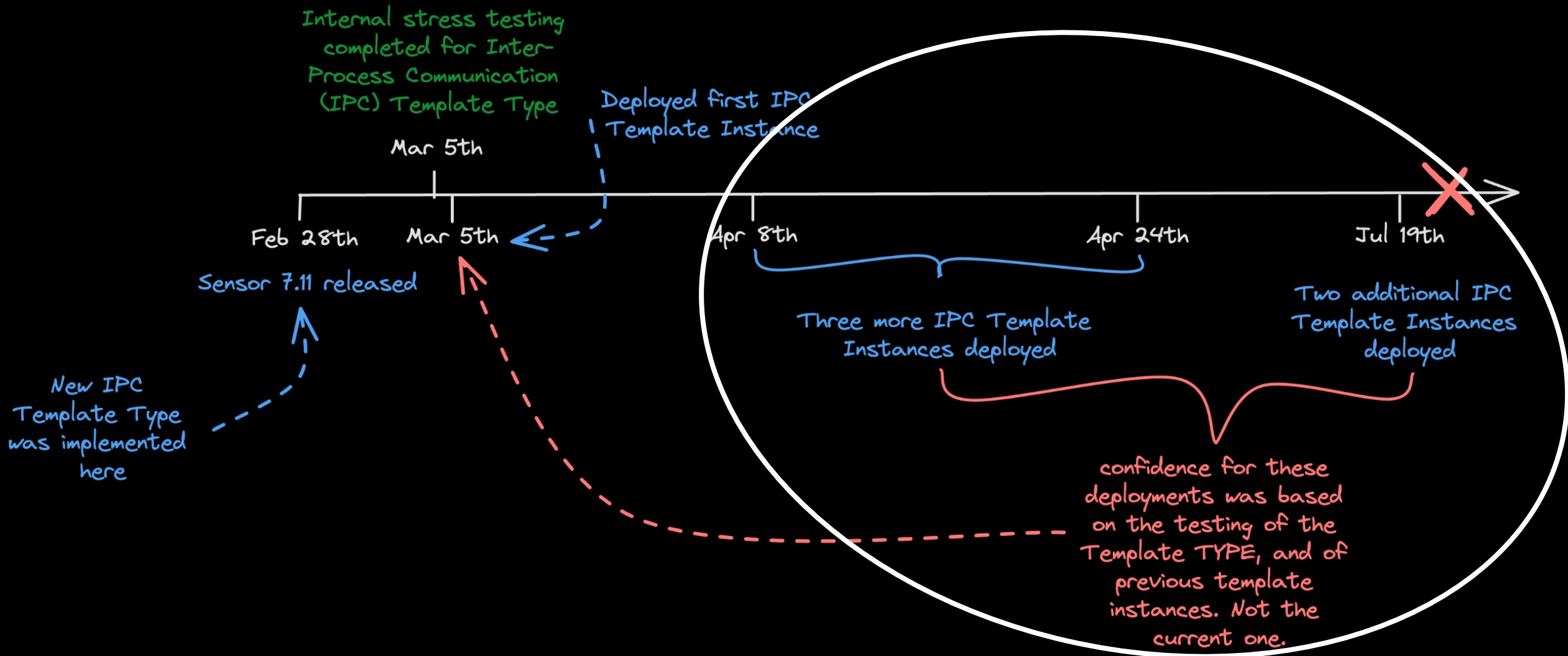
CrowdStrike's Back



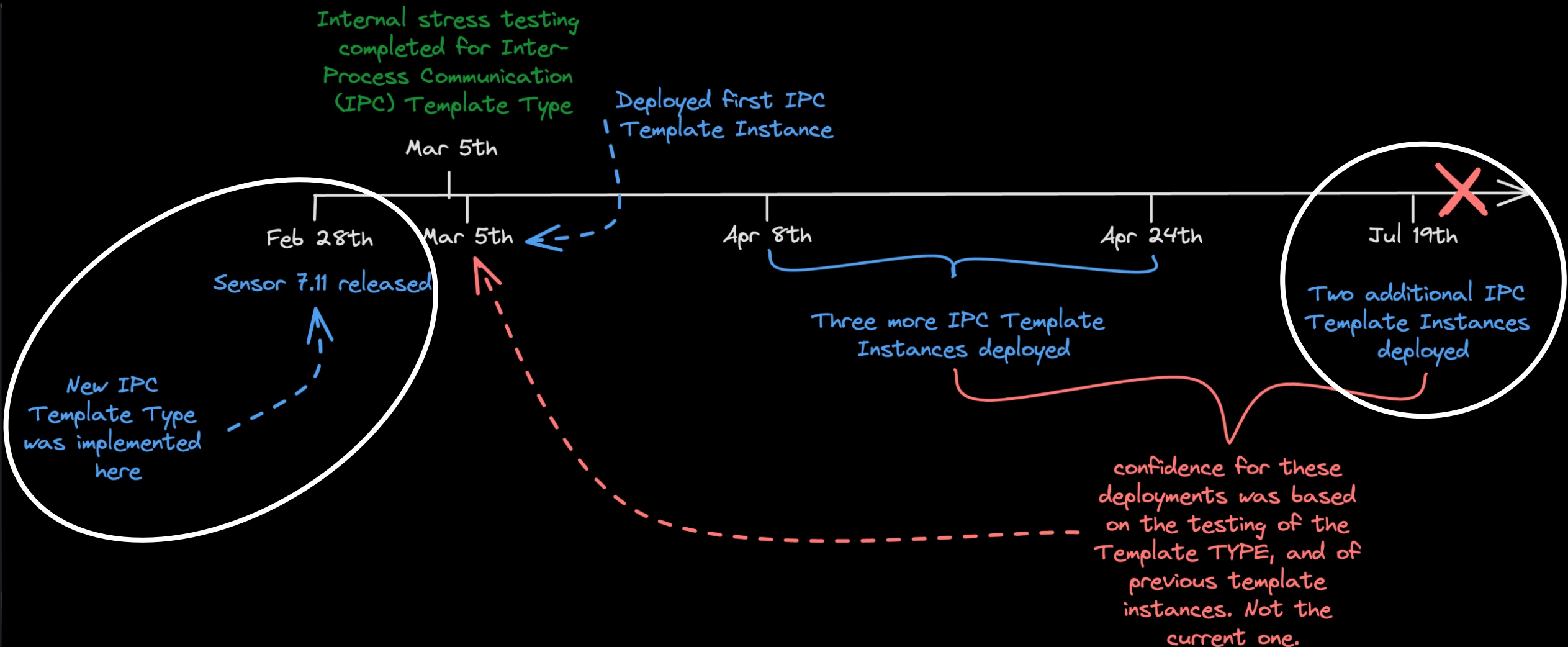
CrowdStrike's Back



CrowdStrike's Back



CrowdStrike's Back



CrowdStrike's Back

Выводы CrowdStrike

CrowdStrike's Back

Выводы CrowdStrike
— Лучше работать

CrowdStrike's Back

Выводы CrowdStrike
— Лучше работать

* Локальное dev-тестирование

CrowdStrike's Back

Выводы CrowdStrike
— Лучше работать

- * Локальное dev-тестирование
- * Тестирование обновлений и откатов конфигураций

CrowdStrike's Back

Выводы CrowdStrike
— Лучше работать

- * Локальное dev-тестирование
- * Тестирование обновлений и откатов конфигураций
- * Fuzzing и Fault Injection

CrowdStrike's Back

Выводы CrowdStrike
— Лучше работать

- * Локальное dev-тестирование
- * Тестирование обновлений и откатов конфигураций
- * Fuzzing и Fault Injection

- * Улучшение облачного Content Validator

CrowdStrike's Back

Выводы CrowdStrike
— Лучше работать

- * Локальное dev-тестирование
- * Тестирование обновлений и откатов конфигураций
- * Fuzzing и Fault Injection

- * Улучшение облачного Content Validator
- * Улучшение обработки ошибок в Content Interpreter

CrowdStrike's Back

Выводы CrowdStrike

— Лучше работать

— Поэтапная раскатка

Microsoft Azure

Microsoft Azure

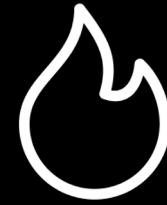
18 июля 2024, 21:40 UTC — 19 июля 2024, 22:00 UTC

Microsoft Azure

18 июля 2024, 21:40 UTC — 19 июля 2024, 22:00 UTC

Ошибка в конфигурировании Azure Storage в регионе Central US привела к недоступности файловой подсистемы у виртуалок

Microsoft Azure



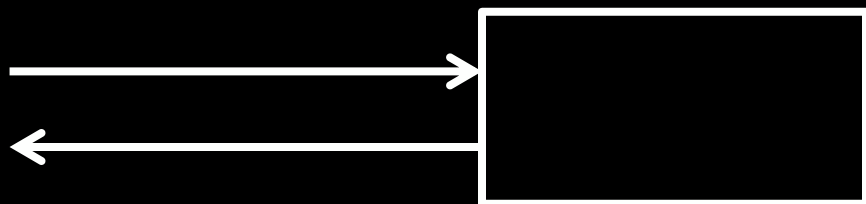
18 июля 2024, 21:40 UTC — 19 июля 2024, 22:00 UTC

Ошибка в конфигурировании Azure Storage в регионе Central US привела к недоступности файловой подсистемы у виртуалок

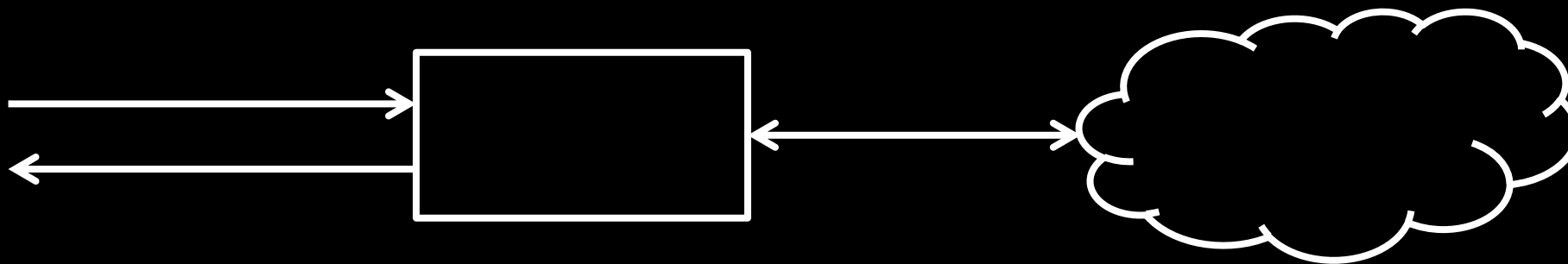
Немая канарейка



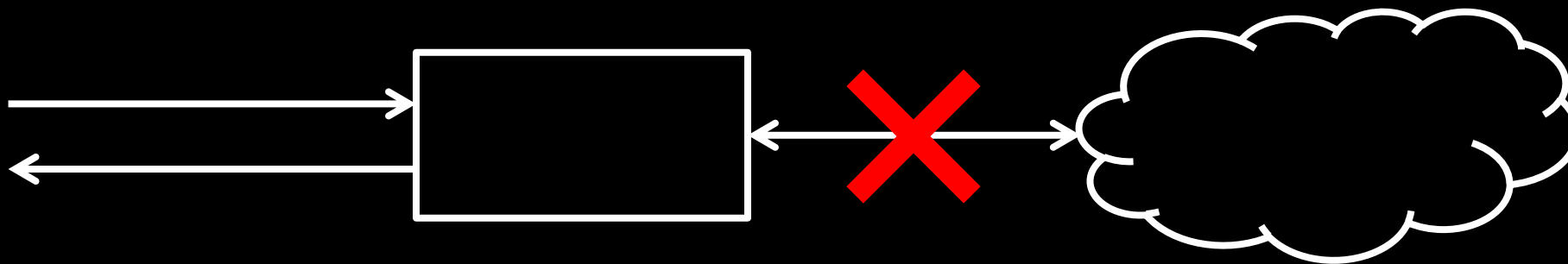
Немая канарейка



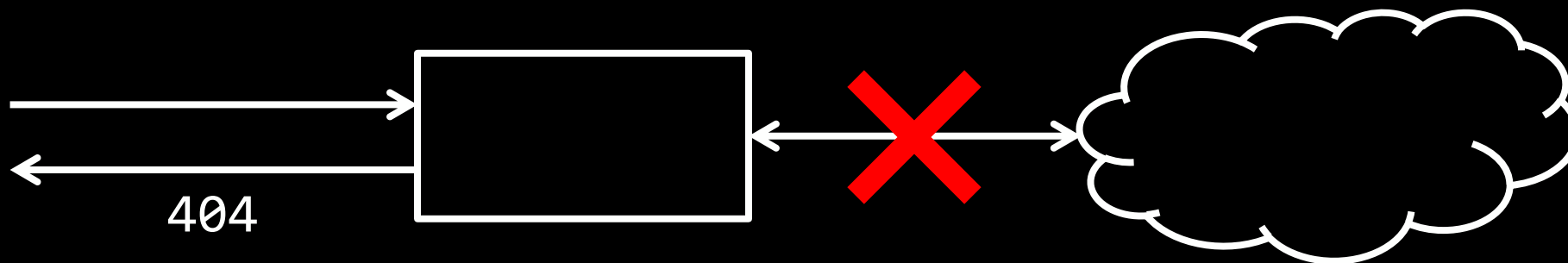
Немая канарейка



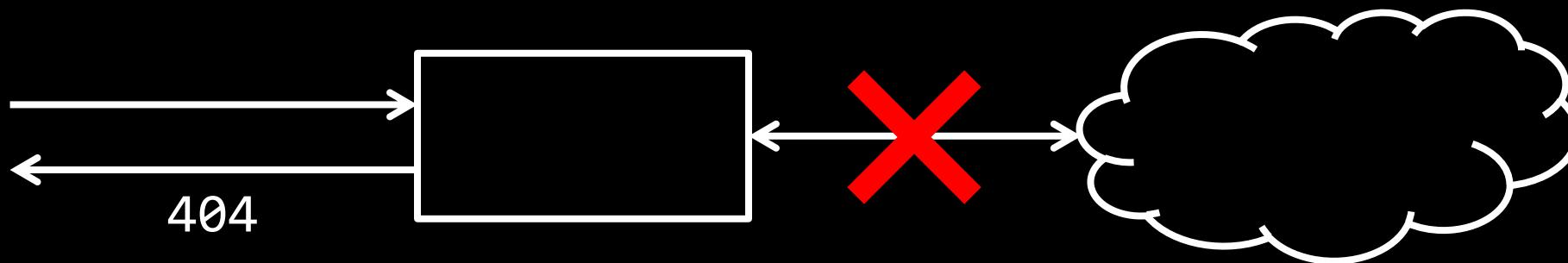
Немая канарейка



Немая канарейка



Немая канарейка



Немая канарейка



Это тоже комбо-факап

Немая канарейка



Это тоже комбо-факап

— канарейка молчала

Немая канарейка



Это тоже комбо-факап
— не протестировали изменение настройки

— канарейка молчала

Немая канарейка



Это тоже комбо-факап

— не протестировали изменение настройки

— релизили вечером

— канарейка молчала

Немая канарейка



Это тоже комбо-факап

— не протестировали изменение настройки

— релизили вечером

— не проверили корректность после обновления

— канарейка молчала

Немая канарейка



Это тоже комбо-факап

— не протестировали изменение настройки

— релизили вечером

— не проверили корректность после обновления

— ошибки затерялись

из-за высокого фона ошибок

— канарейка молчала

Отложенный релиз



Отложенный релиз



В конфиг сервиса добавили комментарии



Отложенный релиз

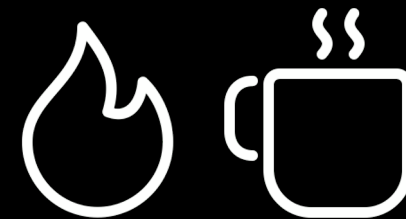


В конфиг сервиса добавили комментарии



Через год (!) при обновлении сервиса
всё сломалось

Отложенный релиз



В конфиг сервиса добавили комментарии



Через год (!) при обновлении сервиса
всё сломалось

CrowdStrike's Back

Выводы CrowdStrike

— Лучше работать

— Поэтапная раскатка

CrowdStrike's Back

Выводы++

CrowdStrike's Back

Выводы++

— Причины BSOD — это ОНИ

CrowdStrike's Back

Выводы++

— Причины BSOD — это ОНИ

Опасные Необратимые Изменения

CrowdStrike's Back

Выводы++

- Причины BSOD — это ОНИ
- Disaster Recovery Plan

CrowdStrike's Back

«Надо было драйверы на Rust писать»
(кто-то в Интернете)

**Я заставлю
сервис
исчезнуть**



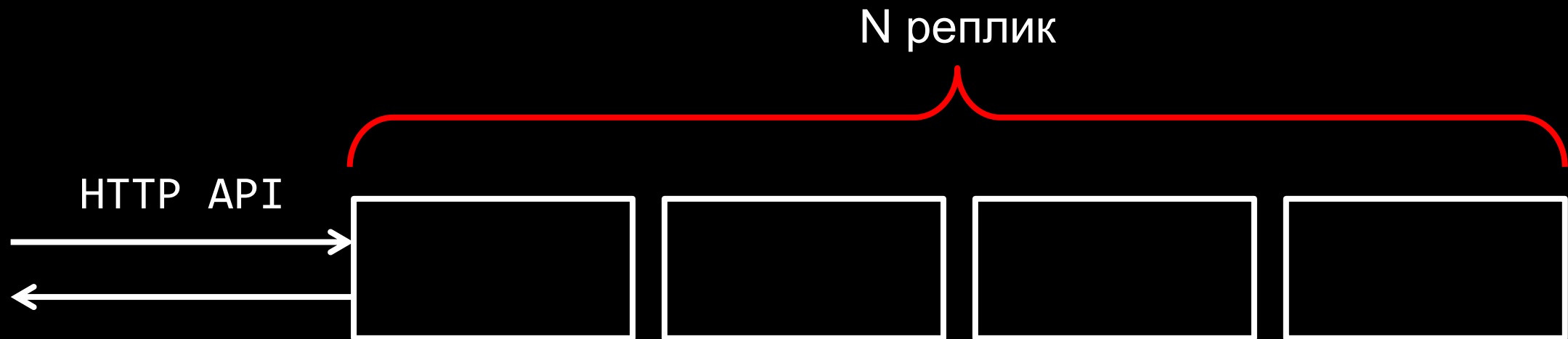
Я заставлю сервис исчезнуть



Я заставлю сервис исчезнуть



Я заставлю сервис исчезнуть



Я заставлю сервис исчезнуть



Наступает ночь...

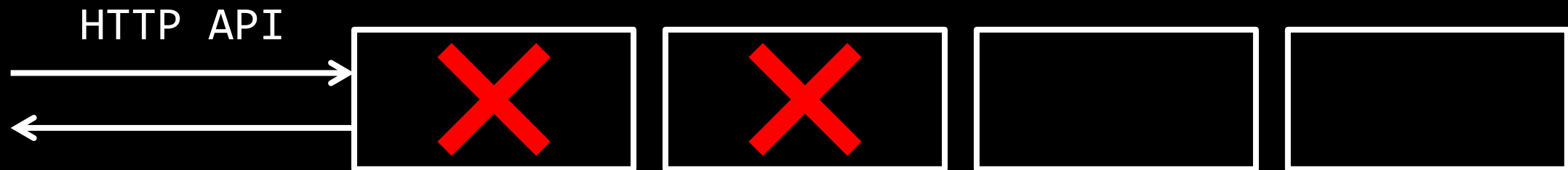
Я заставлю сервис исчезнуть



Наступает ночь...

в течение нескольких минут падает вся топология сервиса

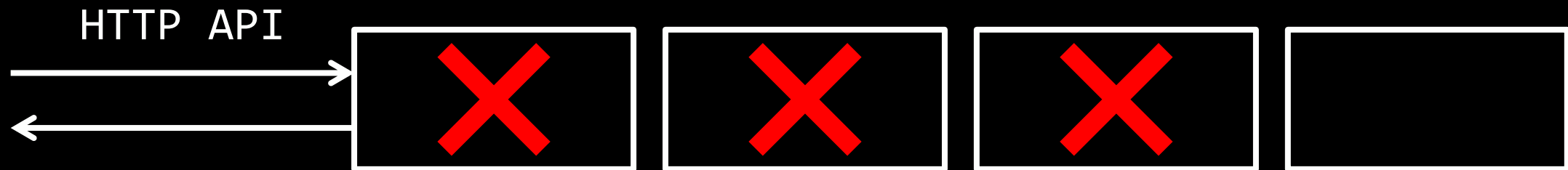
Я заставлю сервис исчезнуть



Наступает ночь...

в течение нескольких минут падает вся топология сервиса

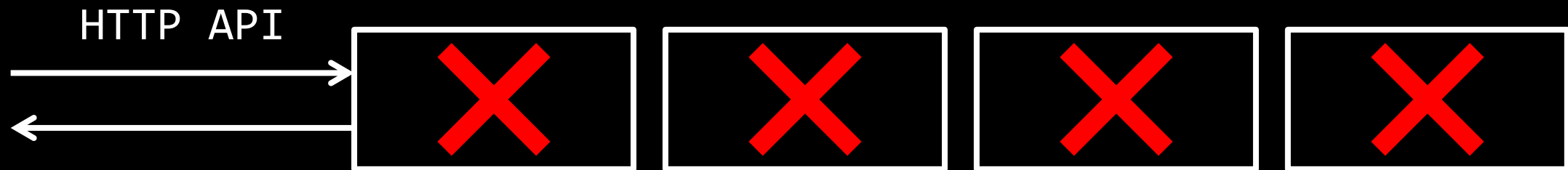
Я заставлю сервис исчезнуть



Наступает ночь...

в течение нескольких минут падает вся топология сервиса

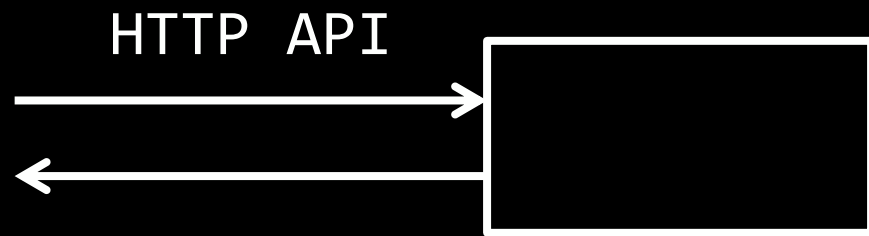
Я заставлю сервис исчезнуть



Наступает ночь...

в течение нескольких минут падает вся топология сервиса

Я заставлю сервис исчезнуть



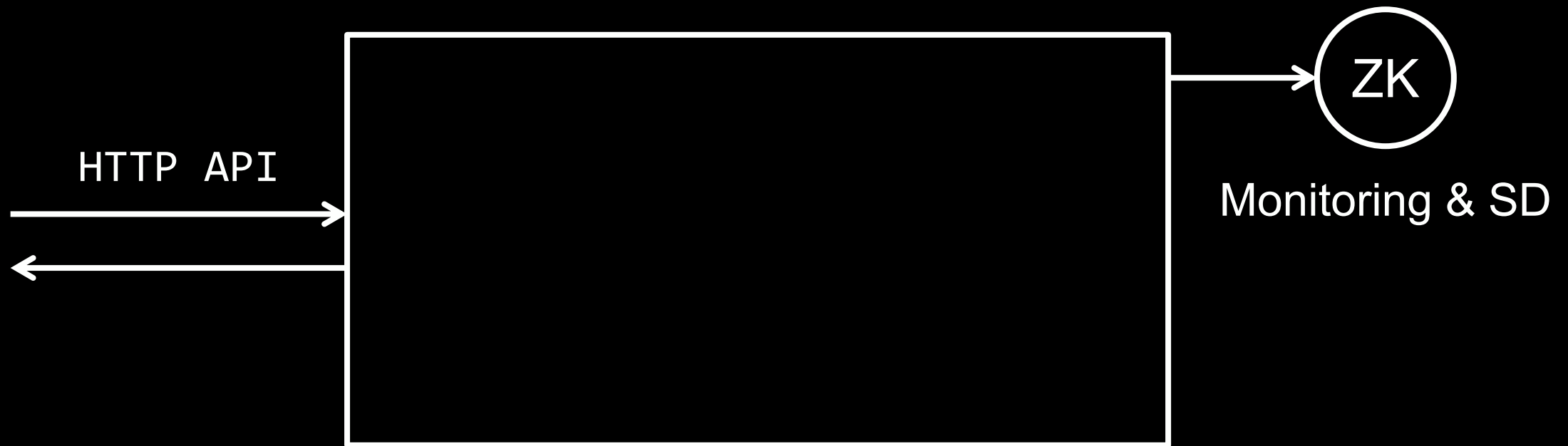
Я заставлю сервис исчезнуть



Я заставлю сервис исчезнуть



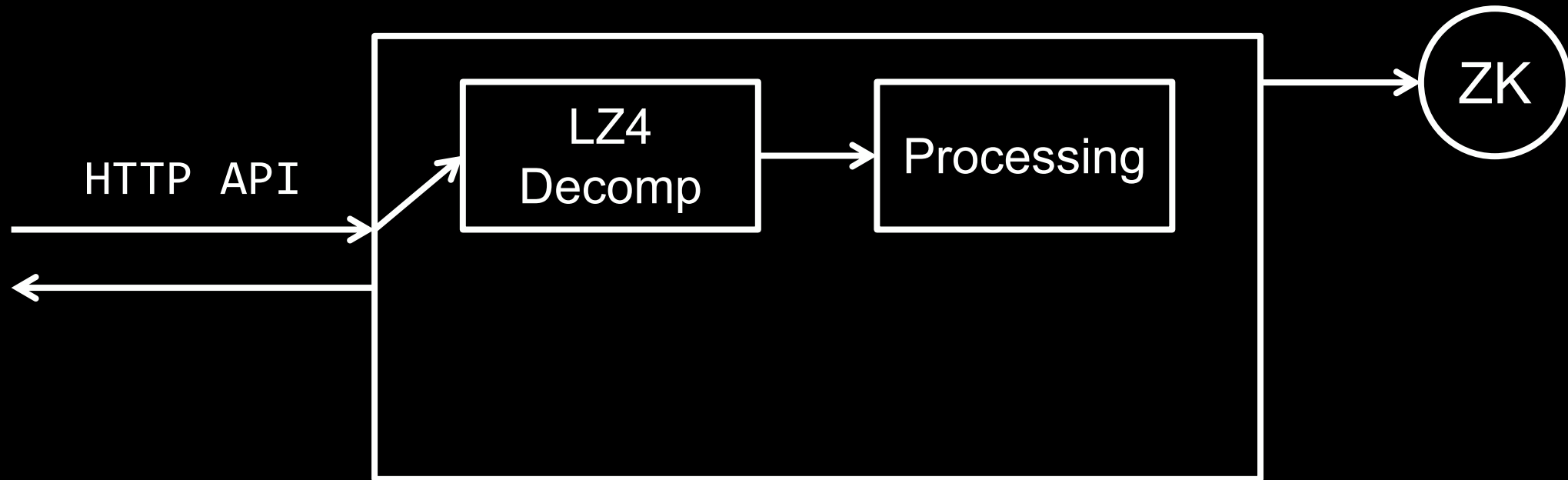
Я заставлю сервис исчезнуть



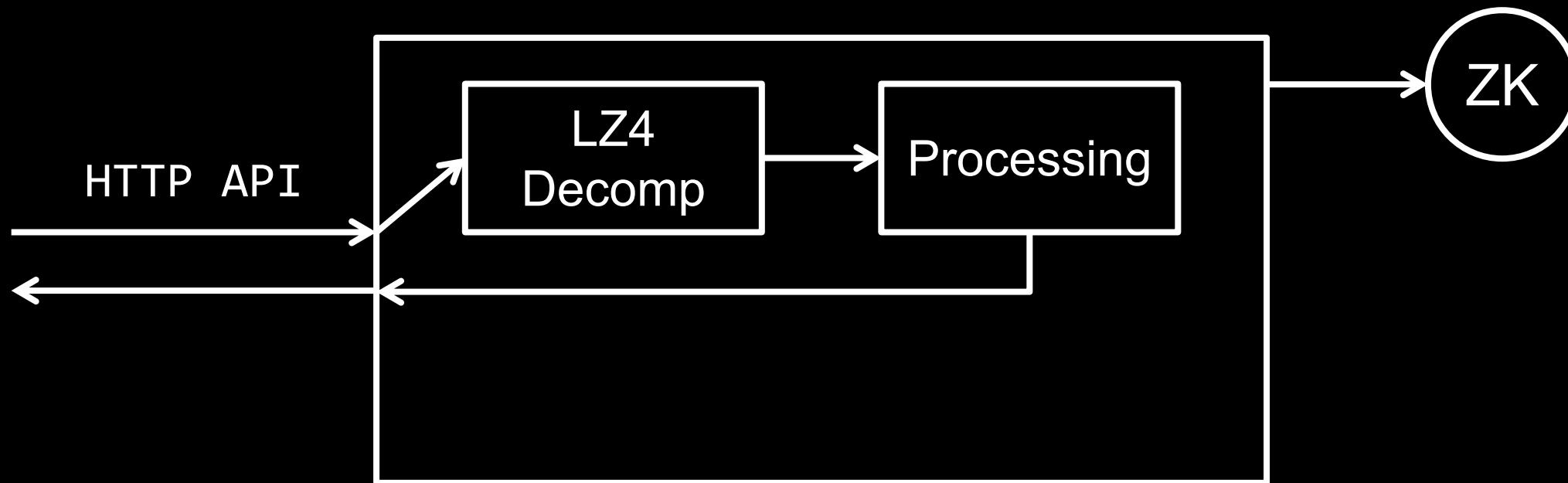
Я заставлю сервис исчезнуть



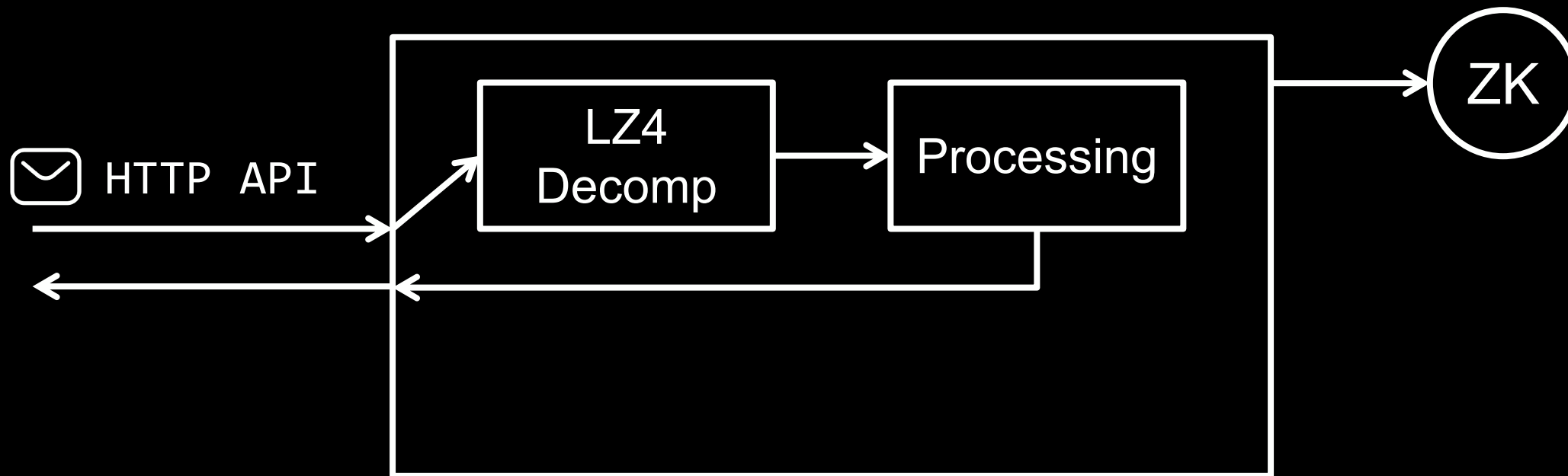
Я заставлю сервис исчезнуть



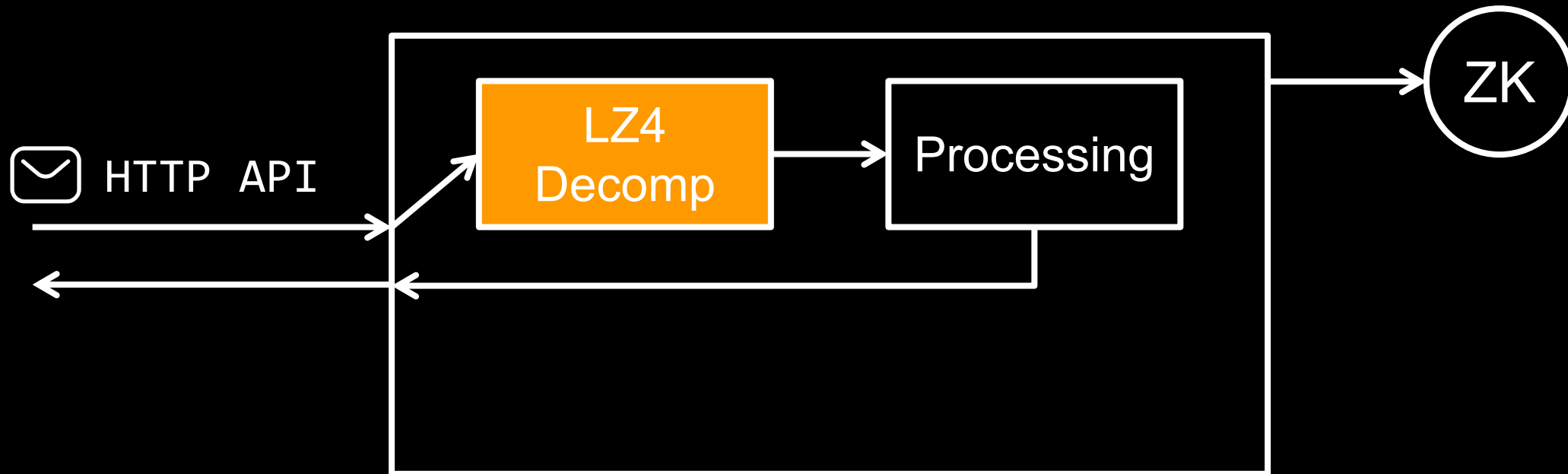
Я заставлю сервис исчезнуть



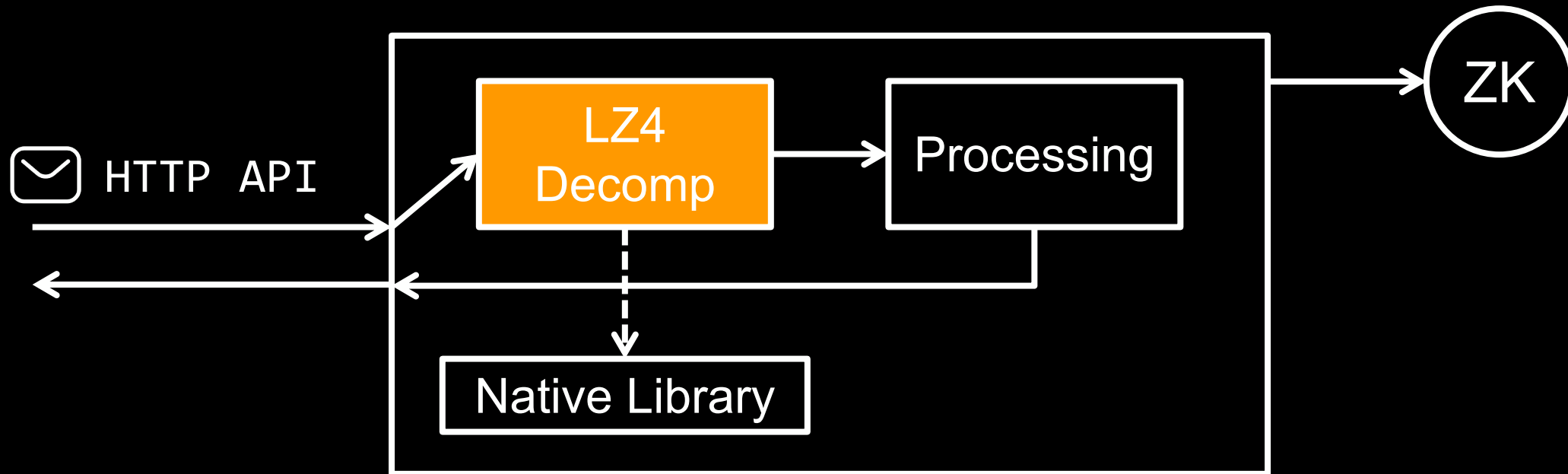
Я заставлю сервис исчезнуть



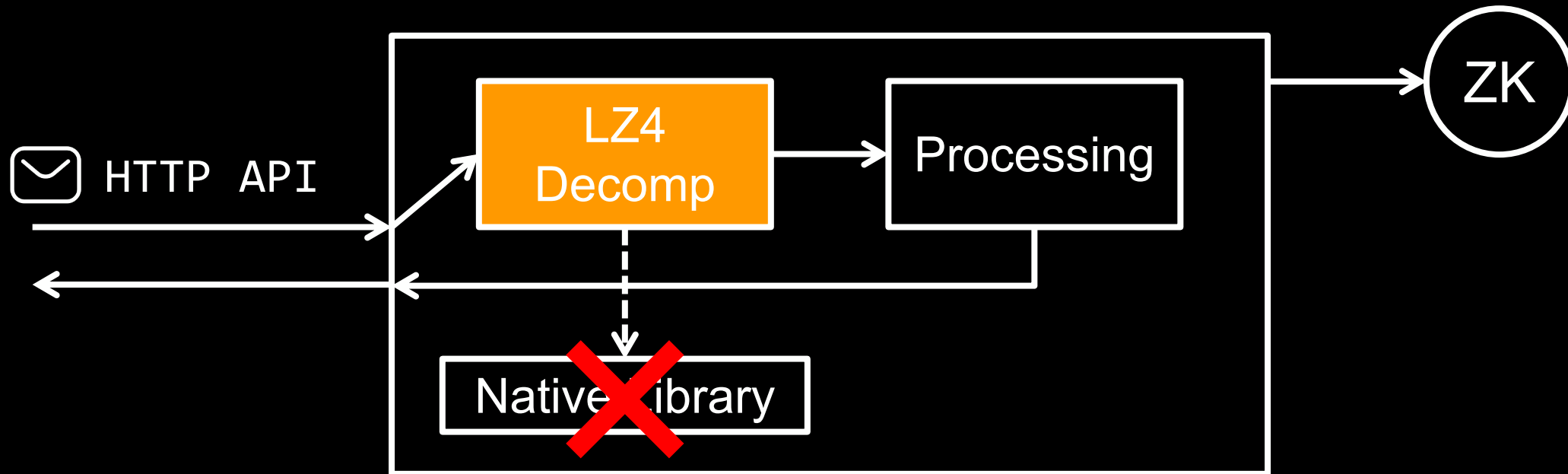
Я заставлю сервис исчезнуть



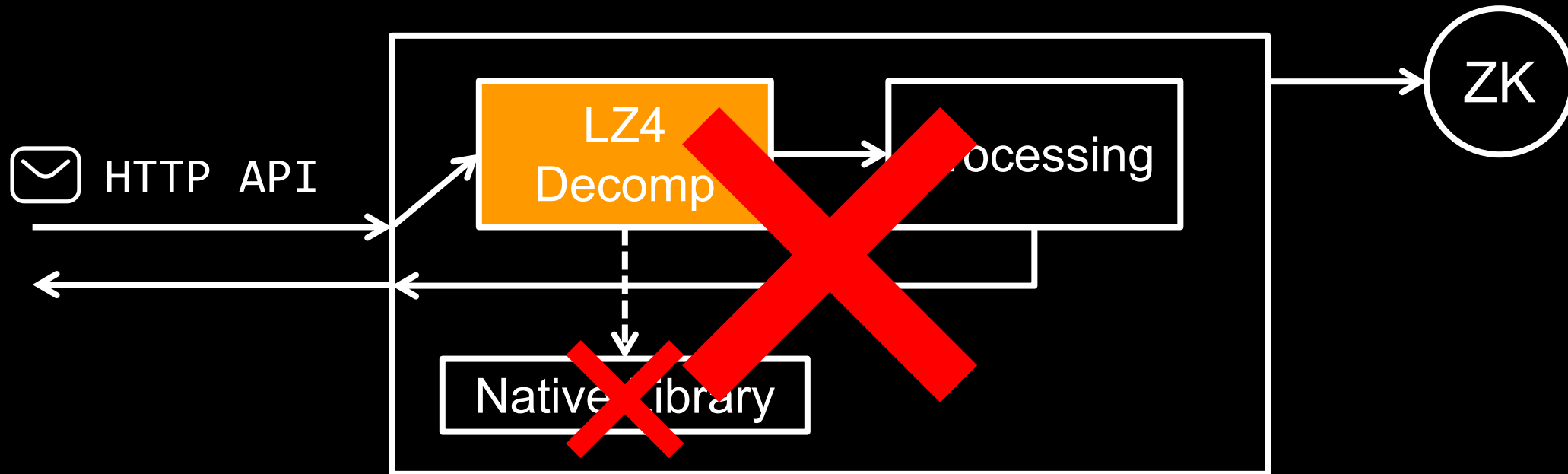
Я заставлю сервис исчезнуть



Я заставлю сервис исчезнуть



Я заставлю сервис исчезнуть

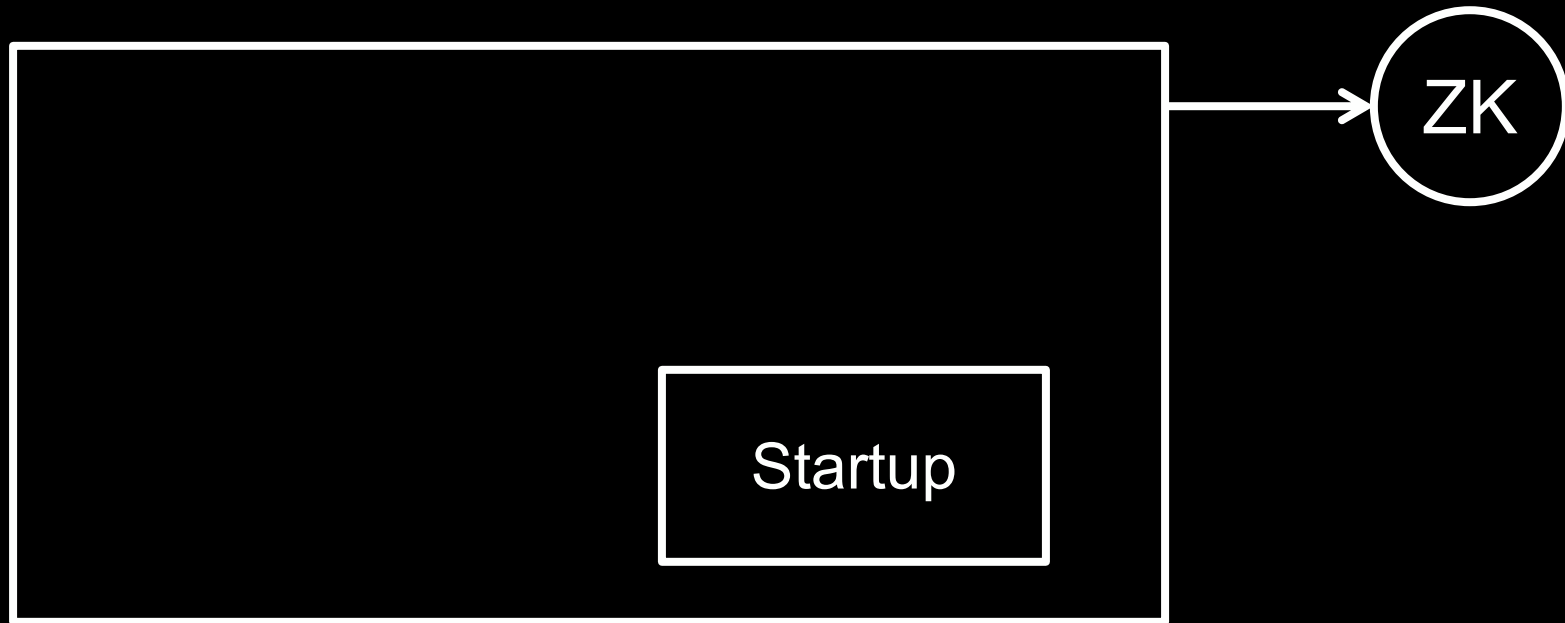


Я заставляю сервис исчезнуть



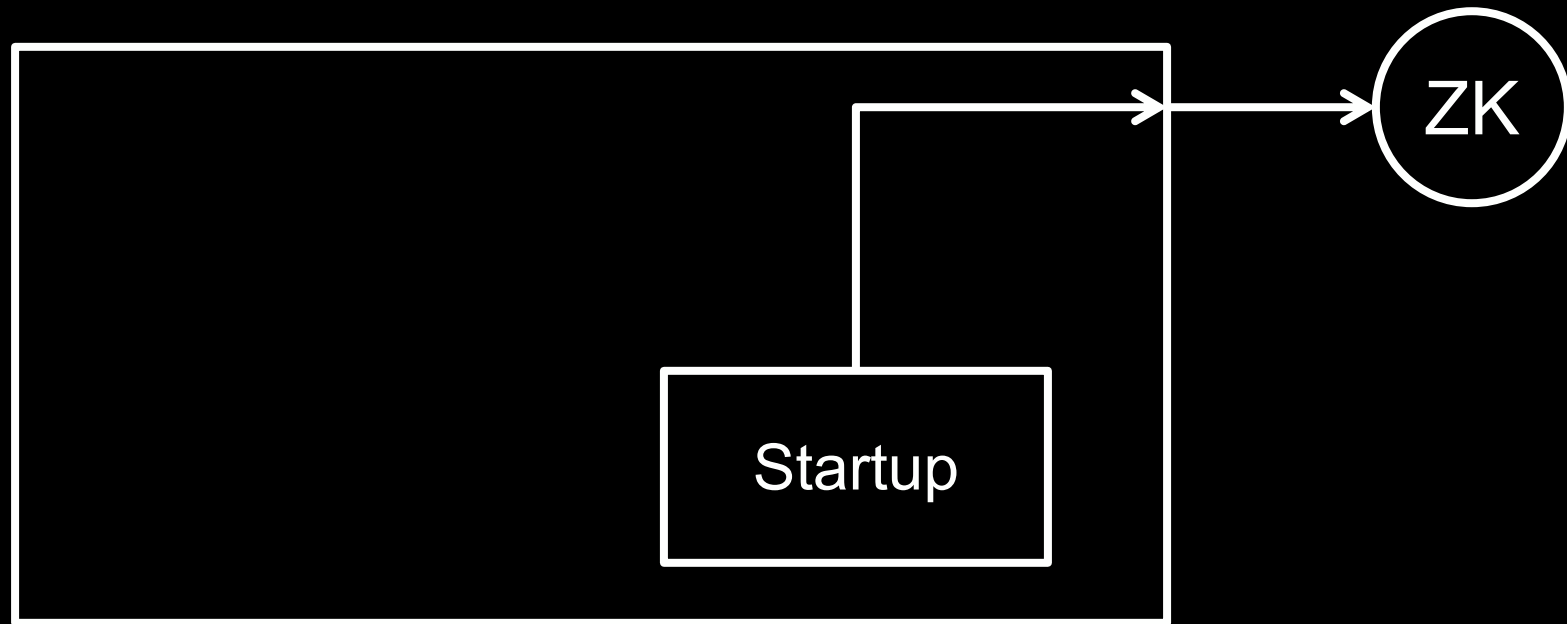
Автоматизированный перезапуск реплики

Я заставлю сервис исчезнуть



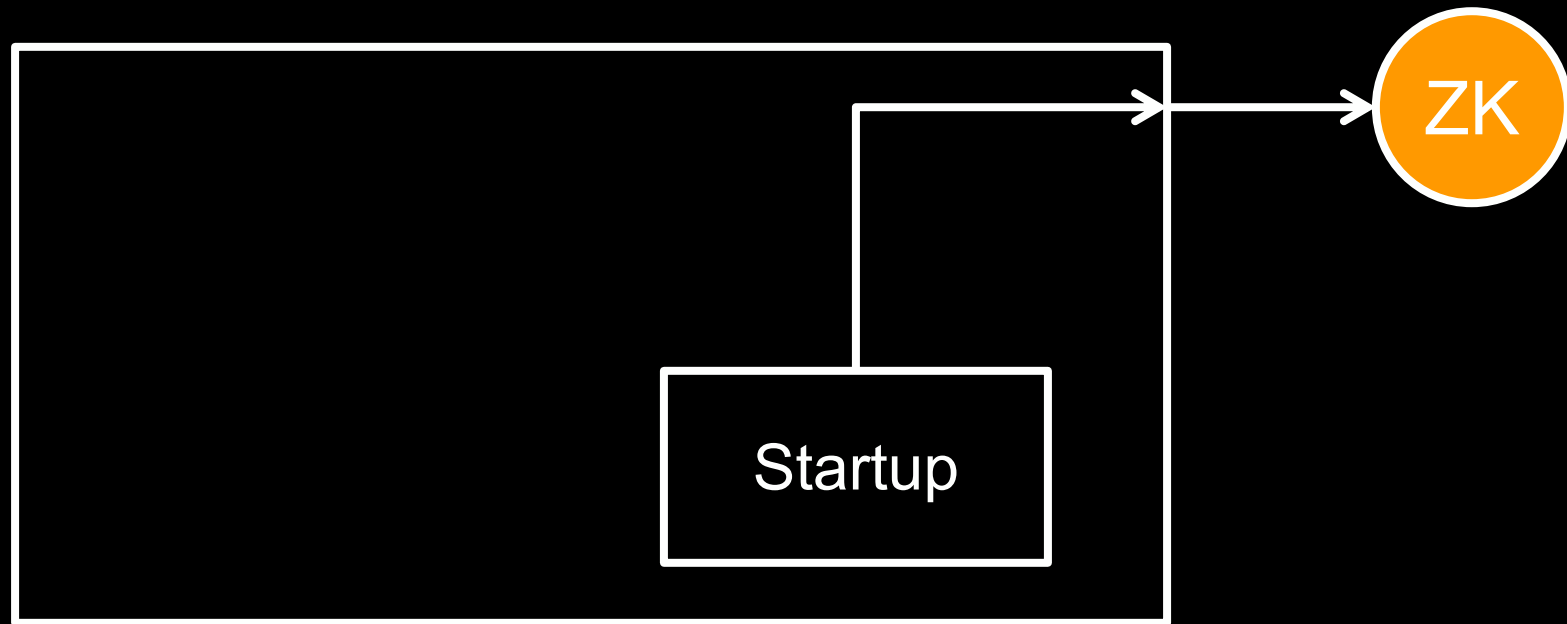
Автоматизированный перезапуск реплики

Я заставлю сервис исчезнуть



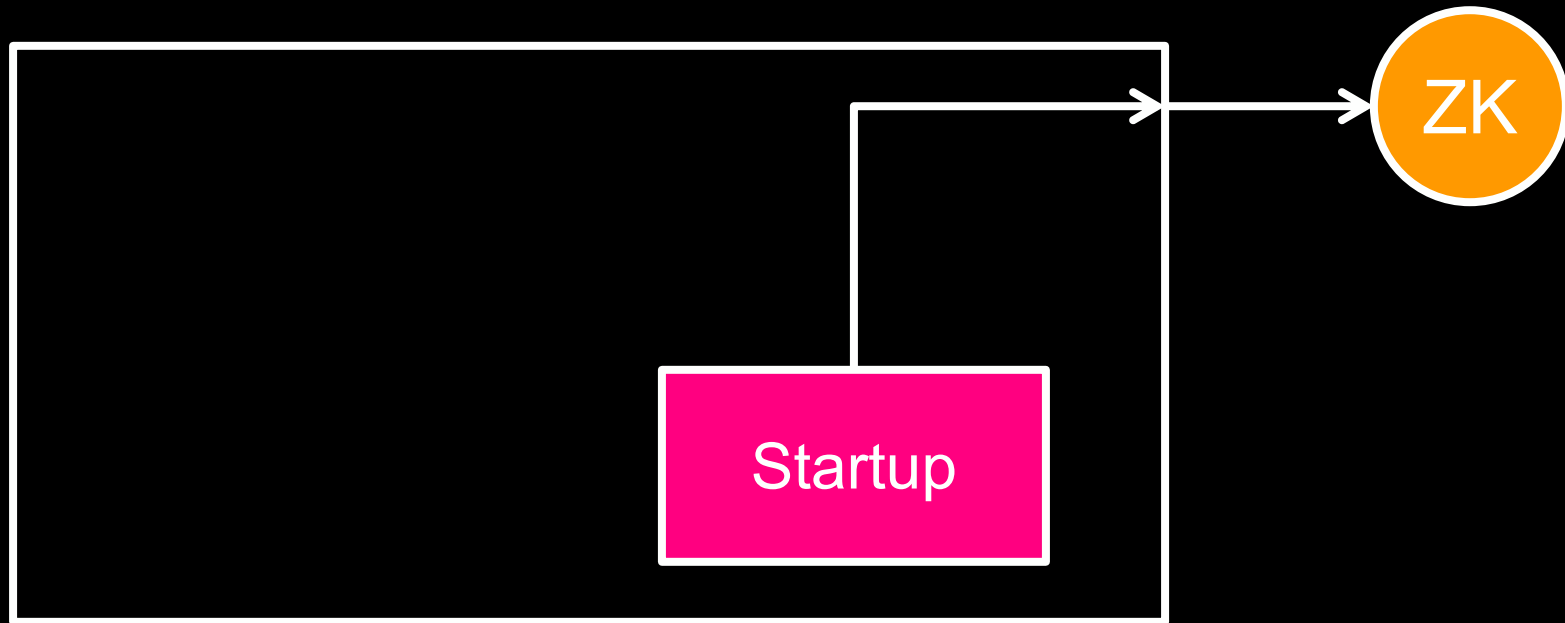
Автоматизированный перезапуск реплики

Я заставлю сервис исчезнуть



В ZooKeeper уже есть эфемерная нода этой реплики

Я заставлю сервис исчезнуть



Запуск прерывается... с Exit Code = 0

Я заставлю сервис исчезнуть

Kubernetes: Container restart policy

Я заставлю сервис исчезнуть

Kubernetes: Container restart policy

- Never
- Always
- OnFailure

Я заставлю сервис исчезнуть

Kubernetes: Container restart policy

- Never
- Always
- OnFailure

Я заставляю сервис исчезнуть

Где повезло?

Я заставлю сервис исчезнуть

Где повезло?

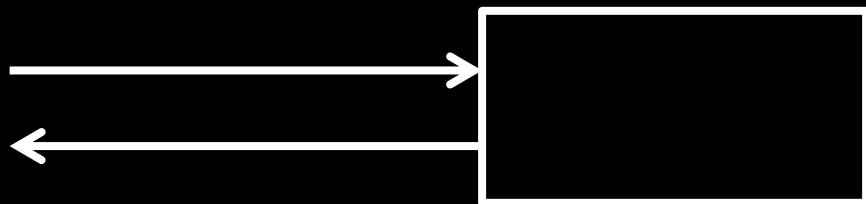
— Упала только одна топология сервиса



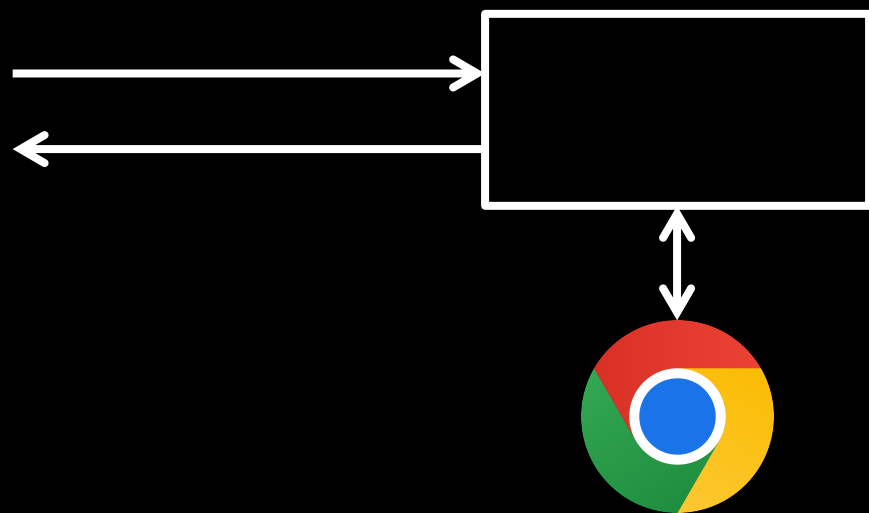
Изолированный Chrome



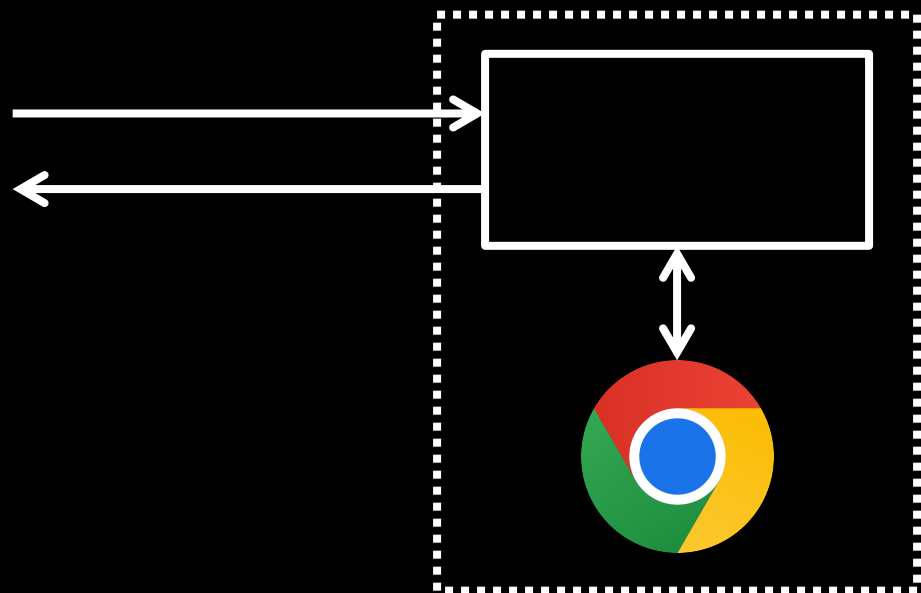
Изолированный Chrome



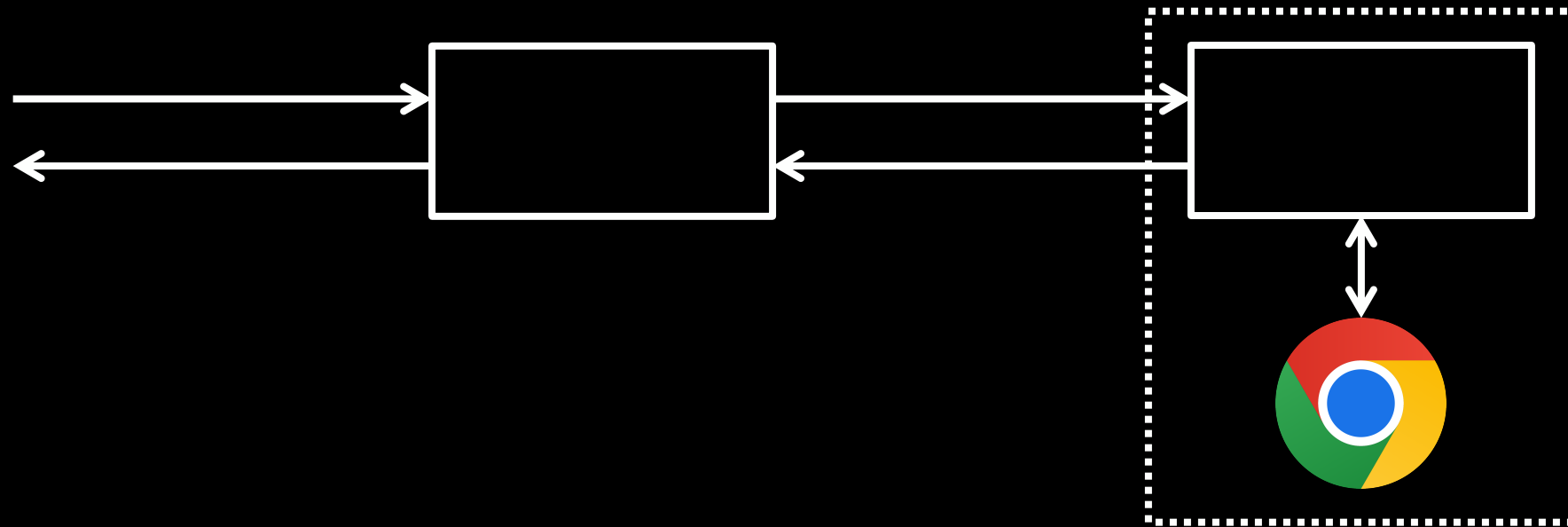
Изолированный Chrome



Изолированный Chrome



Изолированный Chrome



Я заставлю сервис исчезнуть

Выводы

Я заставлю сервис исчезнуть

Выводы

— Не тестировать по ночам

Я заставлю сервис исчезнуть

Выводы

— Не тестировать по ночам... на продакшене

Я заставлю сервис исчезнуть

Выводы

- Не тестировать по ночам... на продакшене
- Обеспечивать изоляцию пользователей

Я заставлю сервис исчезнуть

Выводы

- Не тестировать по ночам... на продакшене
- Обеспечивать изоляцию пользователей
- Лучше работать

Я заставлю сервис исчезнуть

Выводы

- Не тестировать по ночам... на продакшене
- Обеспечивать изоляцию пользователей
- Лучше работать

CrowdStrike

Я заставлю сервис исчезнуть

Выводы

- Не тестировать по ночам... на продакшене
 - Обеспечивать изоляцию пользователей
 - Лучше работать
- * сценарии обработки ошибок native-библиотек в memory-safe языках

Я заставлю сервис исчезнуть

Выводы

- Не тестировать по ночам... на продакшене
- Обеспечивать изоляцию пользователей
- Лучше работать
 - * сценарии обработки ошибок native-библиотек в memory-safe языках
 - * дополнительные fail-сценарии тестирования

73 часа

ROBBLOX

73 часа

ROBLOX

73 часа Roblox

73 часа Roblox

28 – 31 октября 2021

73 часа Roblox

28 – 31 октября 2021

Включение новой фичи в Consul привело к деградации его производительности, а высокая нагрузка стриггерила проблему производительности в BoltDB (WAL для Consul)

73 часа Roblox

28 – 31 октября 2021

Включение новой фичи в Consul привело к деградации его производительности, а высокая нагрузка стриггерила проблему производительности в BoltDB (WAL для Consul)

73 часа Roblox

28 – 31 октября 2021

Включение новой фичи в Consul привело к деградации его производительности, а высокая нагрузка стриггерила проблему производительности в BoltDB (WAL для Consul)

73 часа Roblox



73 часа Roblox



Две проблемы производительности,
не связанные между собой, в **одном инструменте**

73 часа Roblox



Две проблемы производительности,
не связанные между собой, в **одном инструменте**

73 часа Roblox



Общий кластер Consul под разные нагрузки

73 часа Roblox



Зависимость систем мониторинга от Consul

73 часа Roblox



Одна **availability zone** из нескольких дата-центров

73 часа Roblox

Выводы Roblox

- Устранение циклической зависимости Consul и мониторинга
- Улучшение observability Consul
- Миграция в несколько availability zones
- Шардирование Consul

Три дня — 



Три дня — 



4 – 7 апреля 2013

Три дня —



4 – 7 апреля 2013

Для восстановления системы
вручную перезагружали 5 000 серверов
из-за сильной связности сервисов между собой

Три дня —



4 – 7 апреля 2013

Для восстановления системы

вручную перезагружали 5 000 серверов

из-за сильной связности сервисов между собой

Три дня —



4 – 7 апреля 2013

Для восстановления системы

вручную перезагружали 5 000 серверов

из-за сильной связности сервисов между собой

Три дня —



4 – 7 апреля 2013

Для восстановления системы
вручную перезагружали 5 000 серверов
из-за сильной связности сервисов между собой

Три дня —



4 – 7 апреля 2013

Для восстановления системы
вручную перезагружали 5 000 серверов
из-за сильной связности сервисов между собой

**The Slack is dead.
Long live the Slack!**



**The Slack is dead.
Long live the Slack!**



22 февраля 2022, 6:00 AM PST – 9:14 AM PST

The Slack is dead. Long live the Slack!



22 февраля 2022, 6:00 AM PST – 9:14 AM PST

— Изменение настроек привело к неожиданному
росту нагрузки на БД и отказу сервиса

The Slack is dead. Long live the Slack!



22 февраля 2022, 6:00 AM PST – 9:14 AM PST

- **Изменение настроек** привело к **неожиданному росту нагрузки на БД** и отказу сервиса
- Настроили жёсткие ограничения, чтобы часть пользователей могла продолжить работу

73 часа Roblox

Выводы++

73 часа Roblox

Выводы++

— Observability — наше всё

73 часа Roblox

Выводы++

— Observability — наше всё

— Инструменты коммуникации столько же критичны, как и инструменты мониторинга

73 часа Roblox

Выводы++

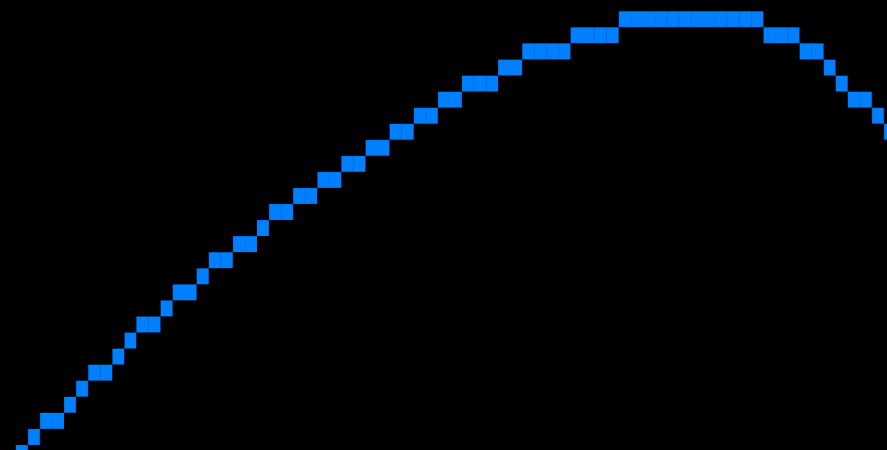
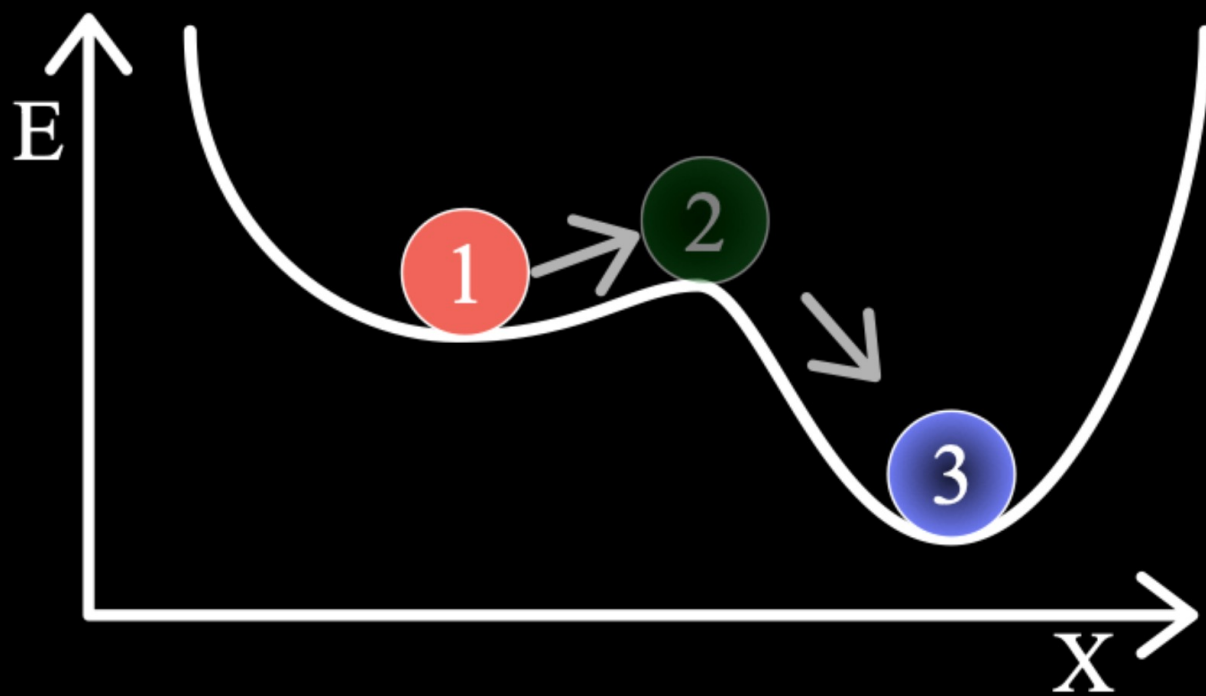
- Observability — наше всё
- Инструменты коммуникации столько же критичны, как и инструменты мониторинга
- Возможность перезапустить систему с нуля

73 часа Roblox

Выводы++

- Observability — наше всё
- Инструменты коммуникации столько же критичны, как и инструменты мониторинга
- Возможность перезапустить систему с нуля
- Превышение нагрузки может иметь необратимые последствия для работы системы

Метастабильные отказы



Метастабильные отказы

Метастабильные отказы

Метастабильное состояние отказа — состояние частичного или полного отказа, в которое попадает система под воздействием триггера и остаётся в нём после удаления триггера

Метастабильные отказы

Метастабильное состояние отказа — состояние частичного или полного отказа, в которое попадает система под воздействием триггера и остаётся в нём после удаления триггера

Метастабильные отказы

Метастабильное состояние отказа —
состояние частичного или полного **отказа**,
в которое попадает система под воздействием
триггера и остаётся в нём после удаления
триггера

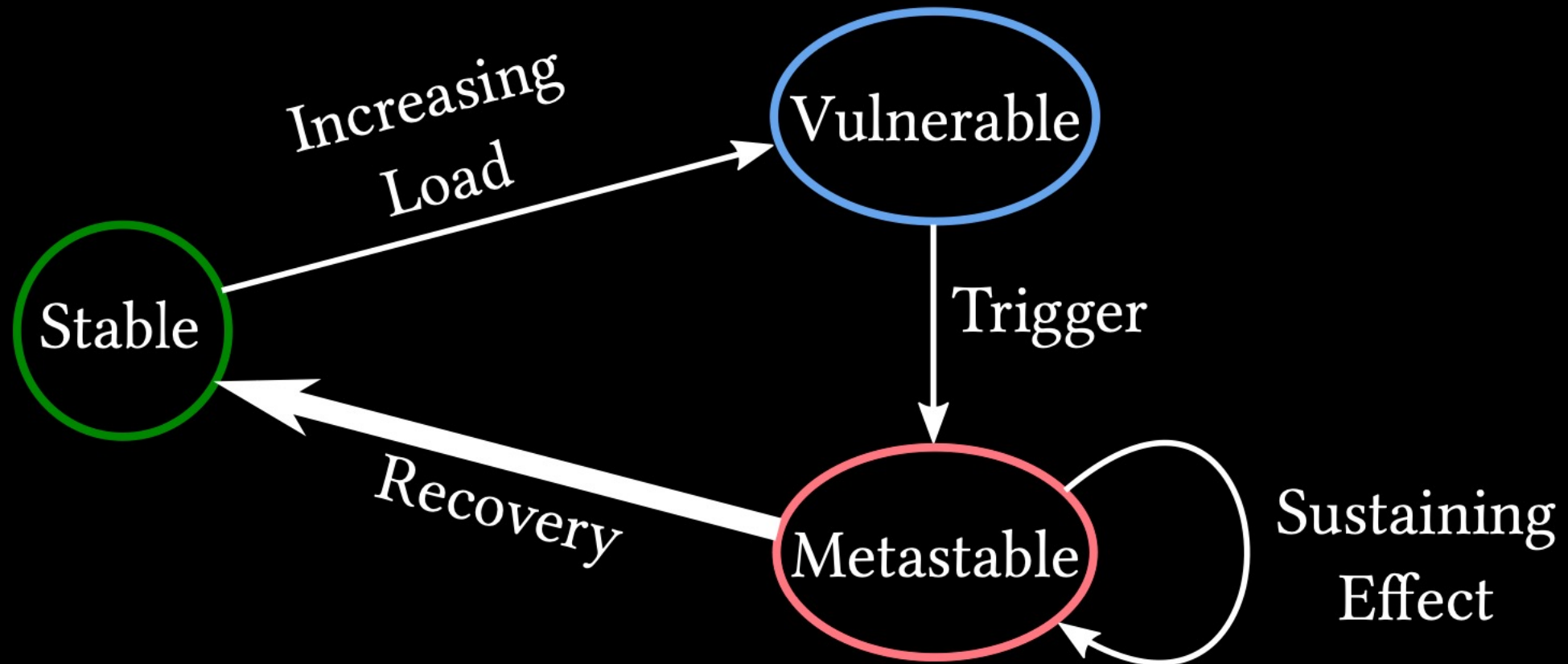
Метастабильные отказы

Метастабильное состояние отказа — состояние частичного или полного отказа, в которое попадает система **под воздействием триггера** и остаётся в нём после удаления триггера

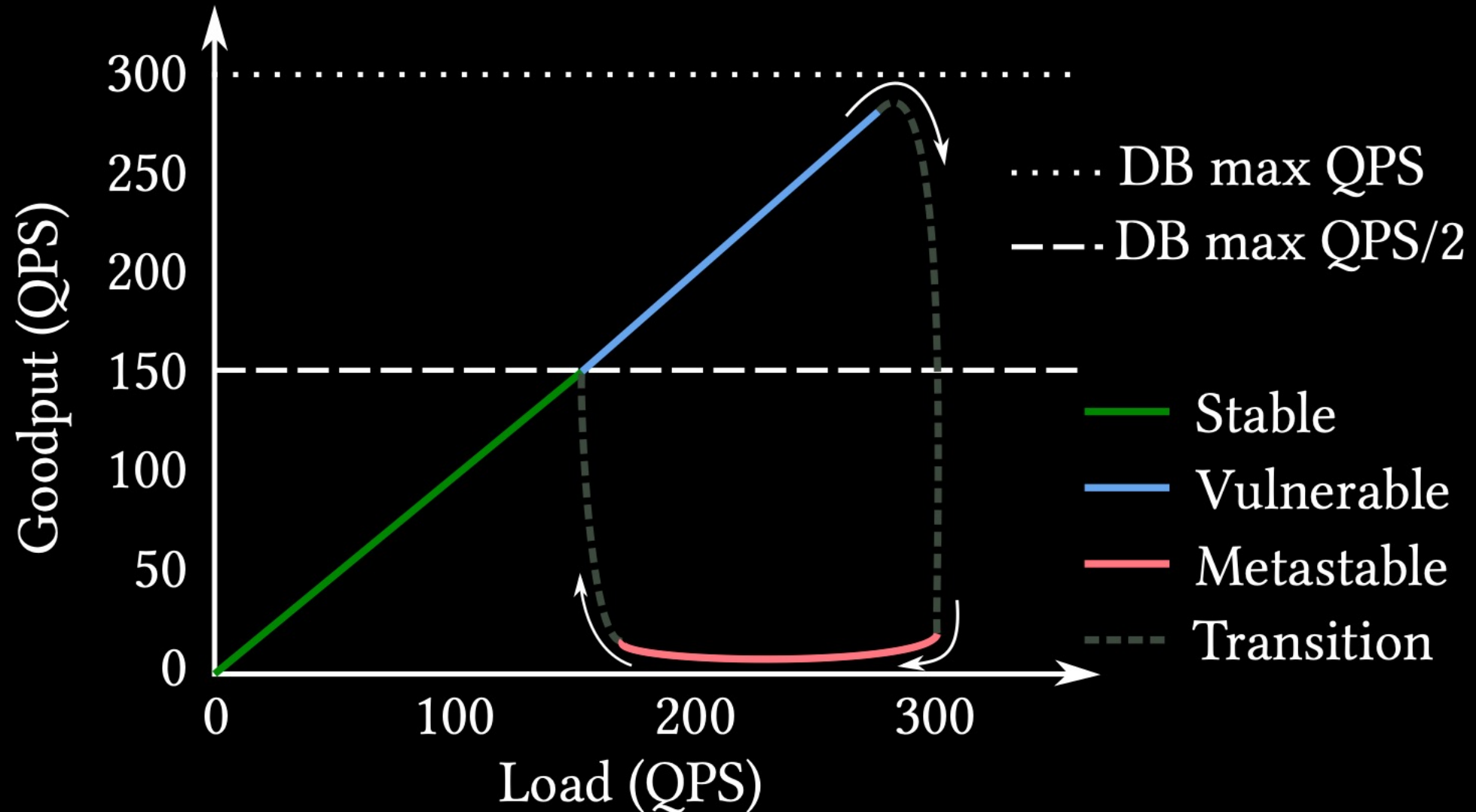
Метастабильные отказы

Метастабильное состояние отказа — состояние частичного или полного отказа, в которое попадает система под воздействием триггера и **остаётся** в нём **после удаления триггера**

Метастабильные отказы



Метастабильные отказы



Кafka в один конец

Кafka в один конец

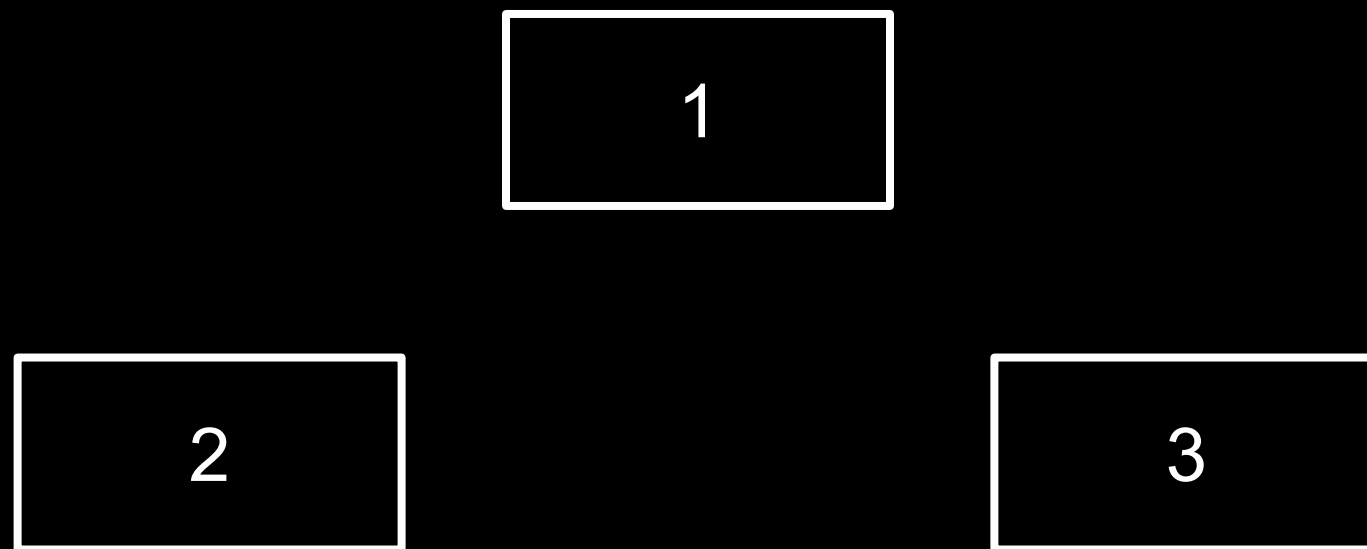
Apache Kafka — потрясающе
надёжный инструмент*

Kafka в один конец

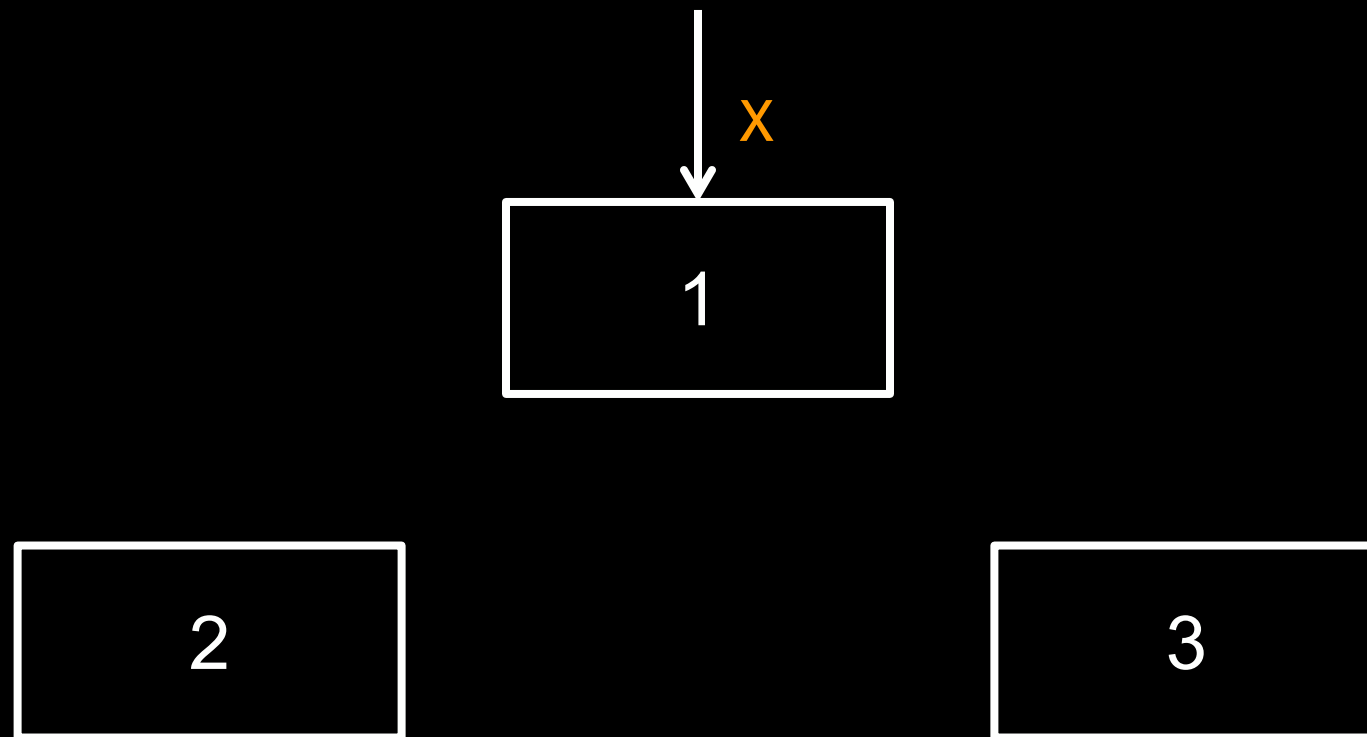
Apache Kafka — потрясающе
надёжный инструмент*

*) если не трогать

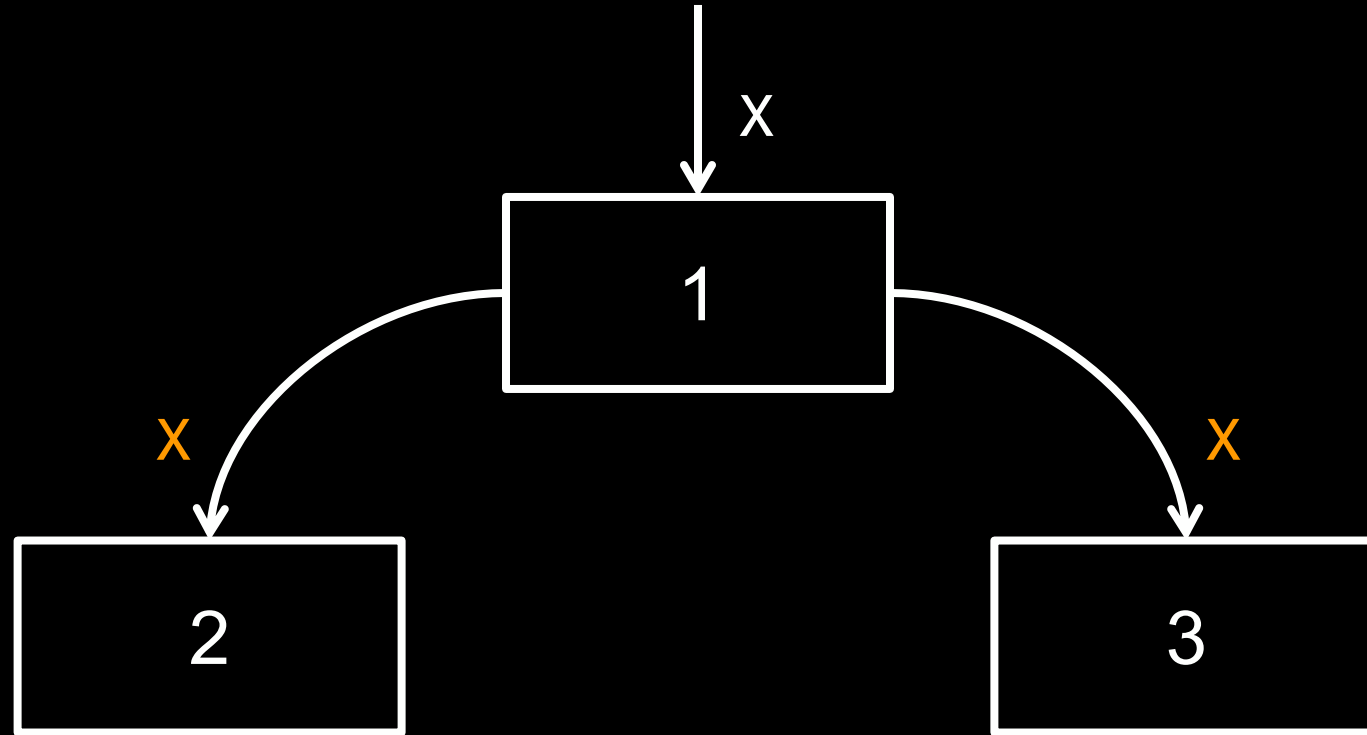
Кafka в один конец



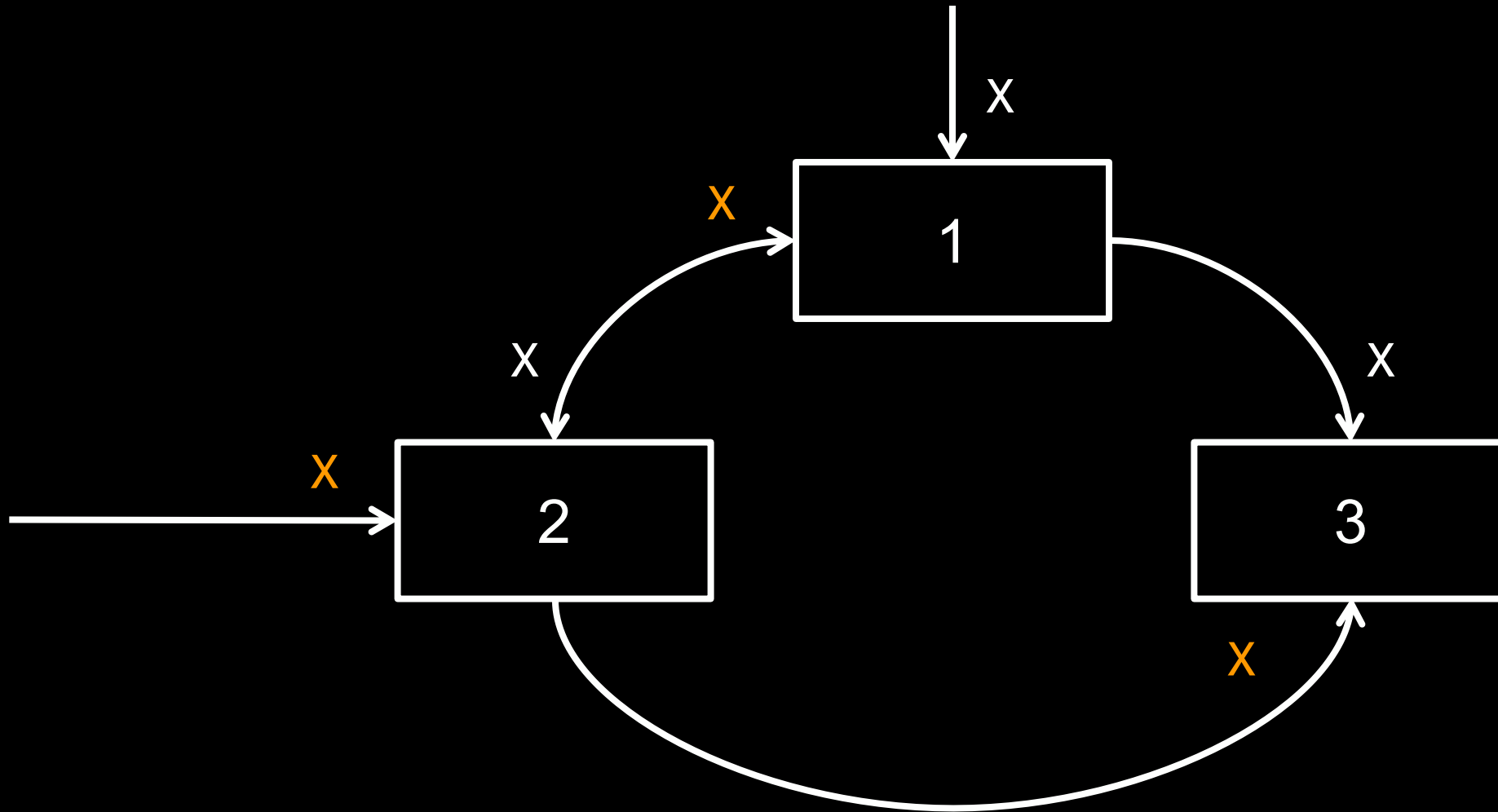
Кafka в один конец



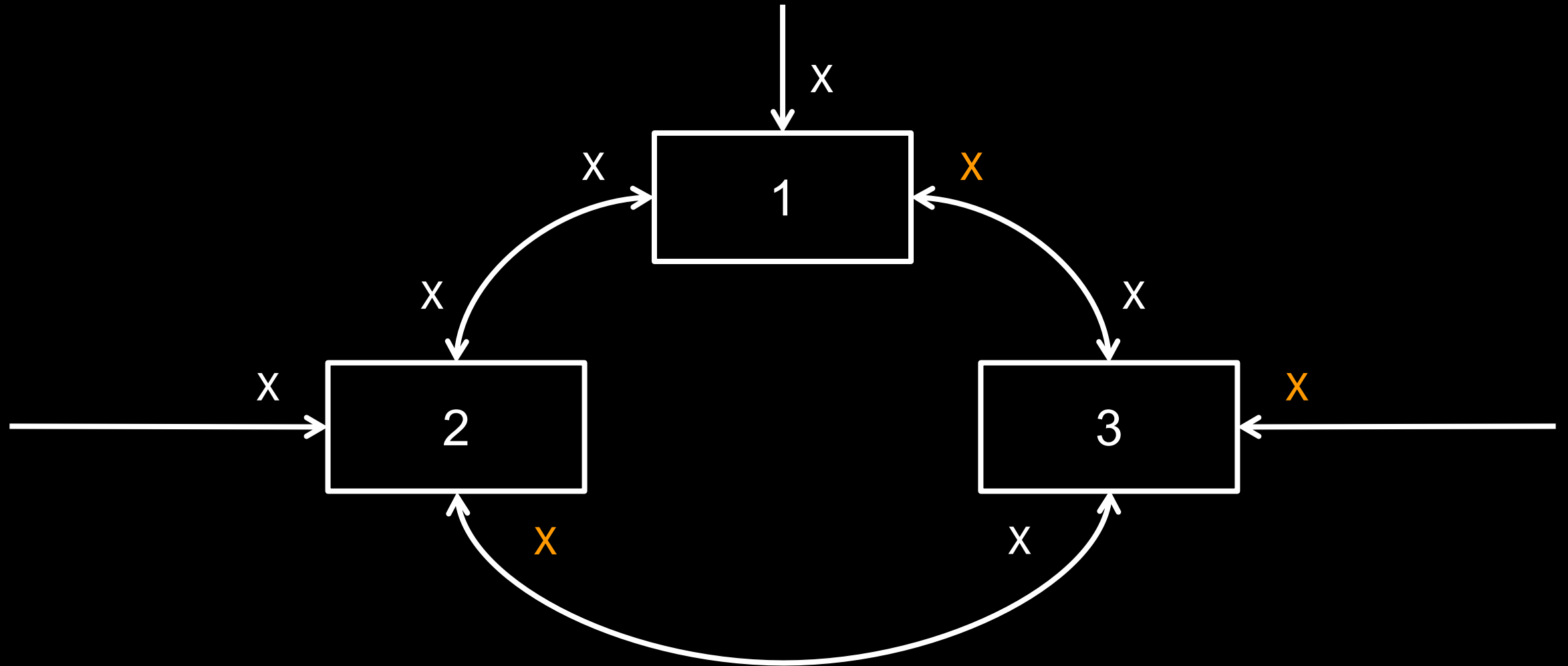
Кafka в один конец



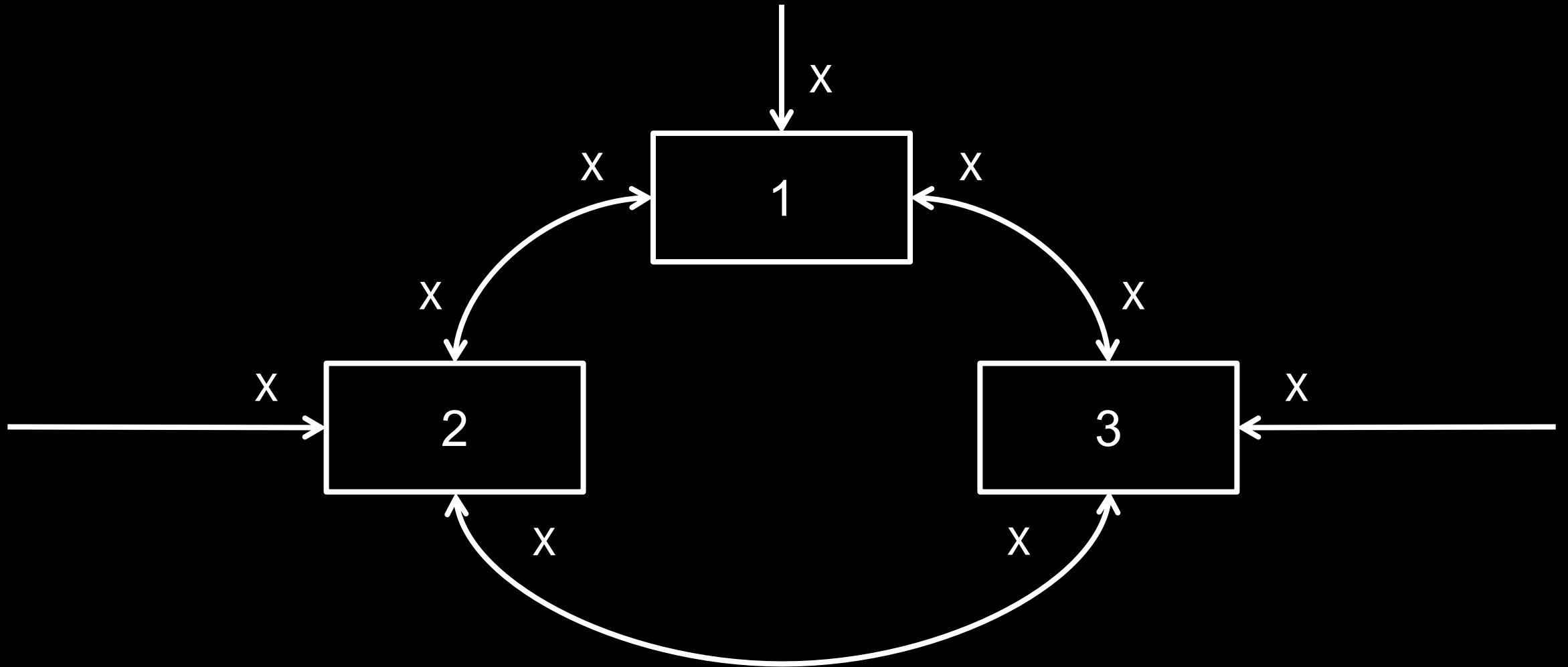
Кafka в один конец



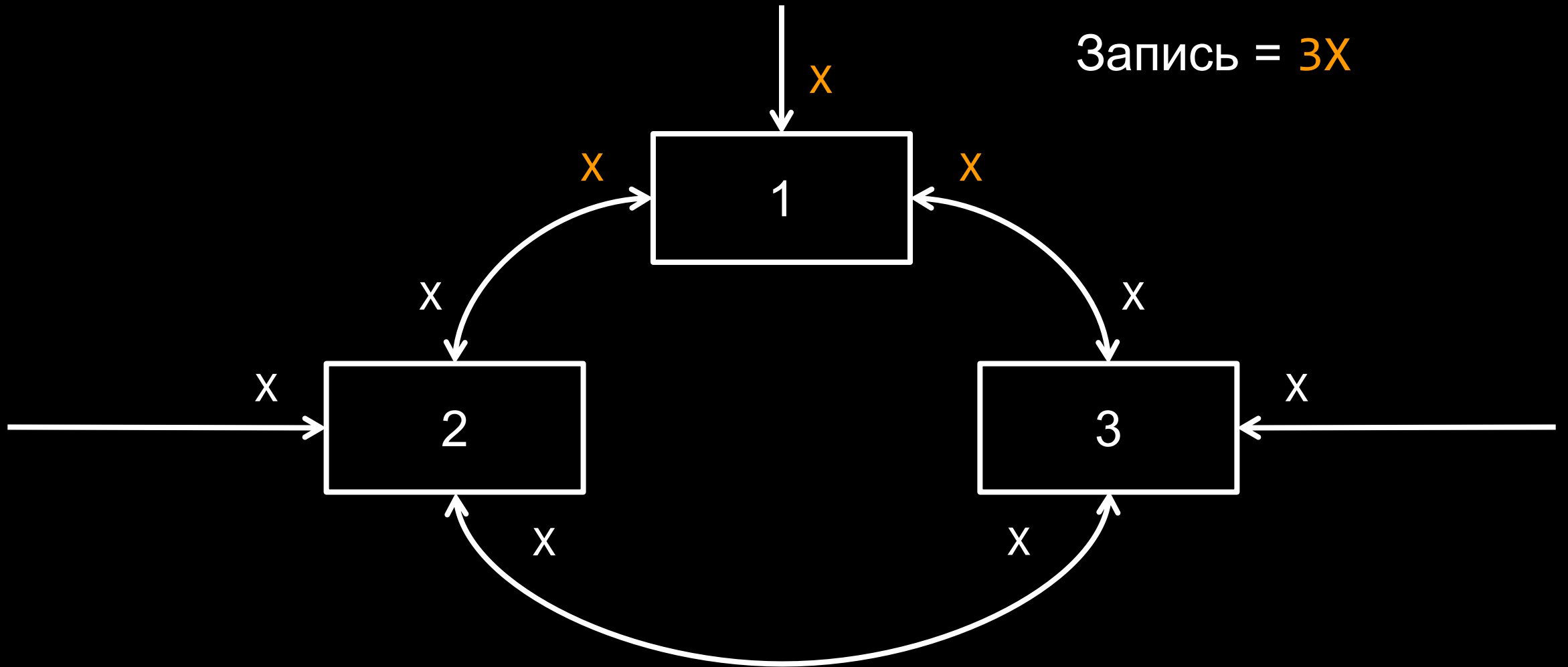
Кafka в один конец



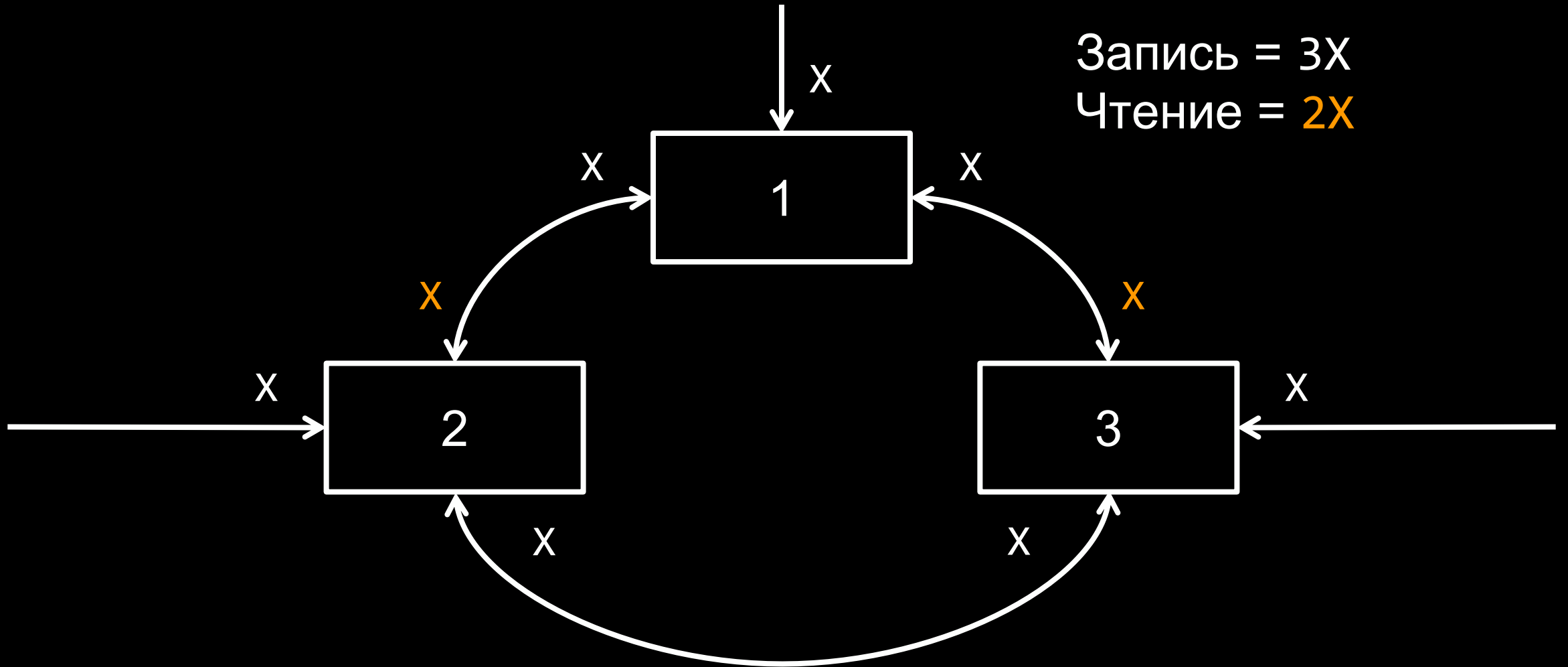
Кafka в один конец



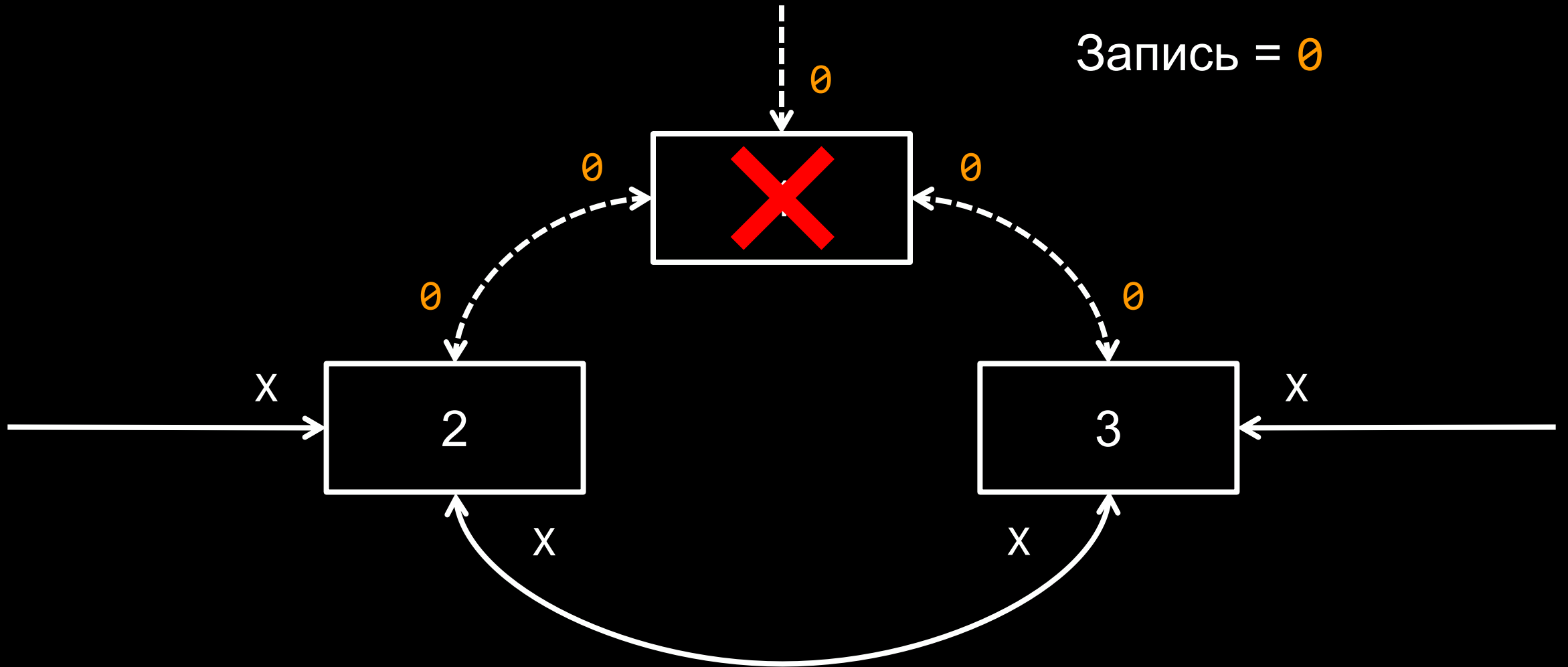
Кafka в один конец



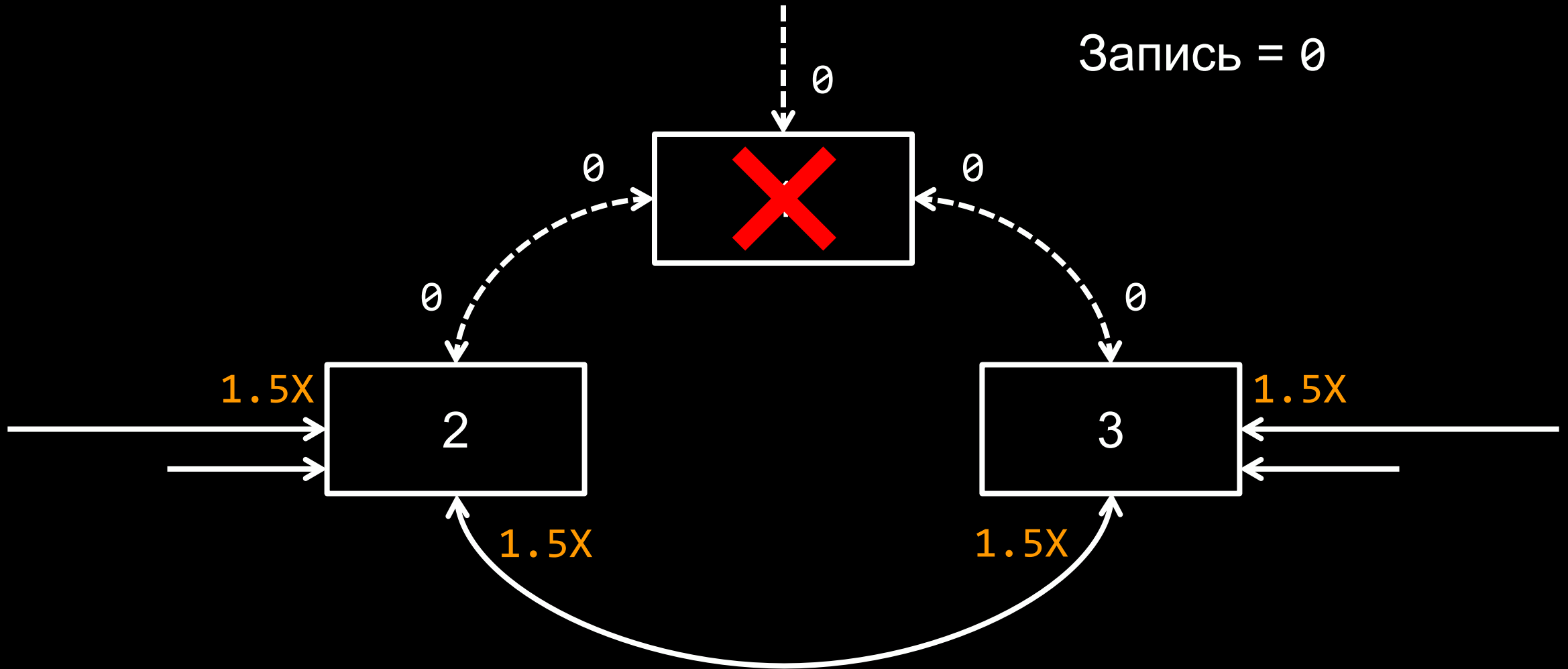
Кafka в один конец



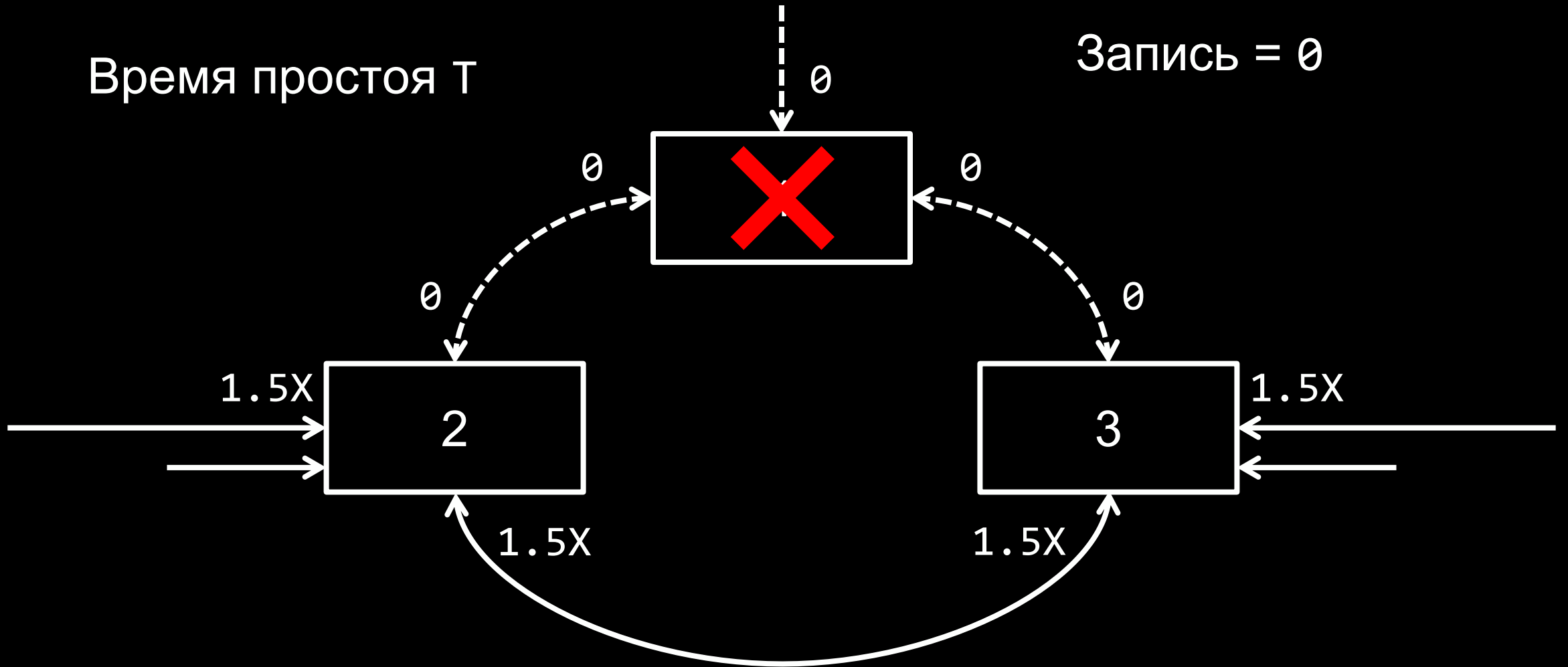
Кafka в один конец



Кafka в один конец



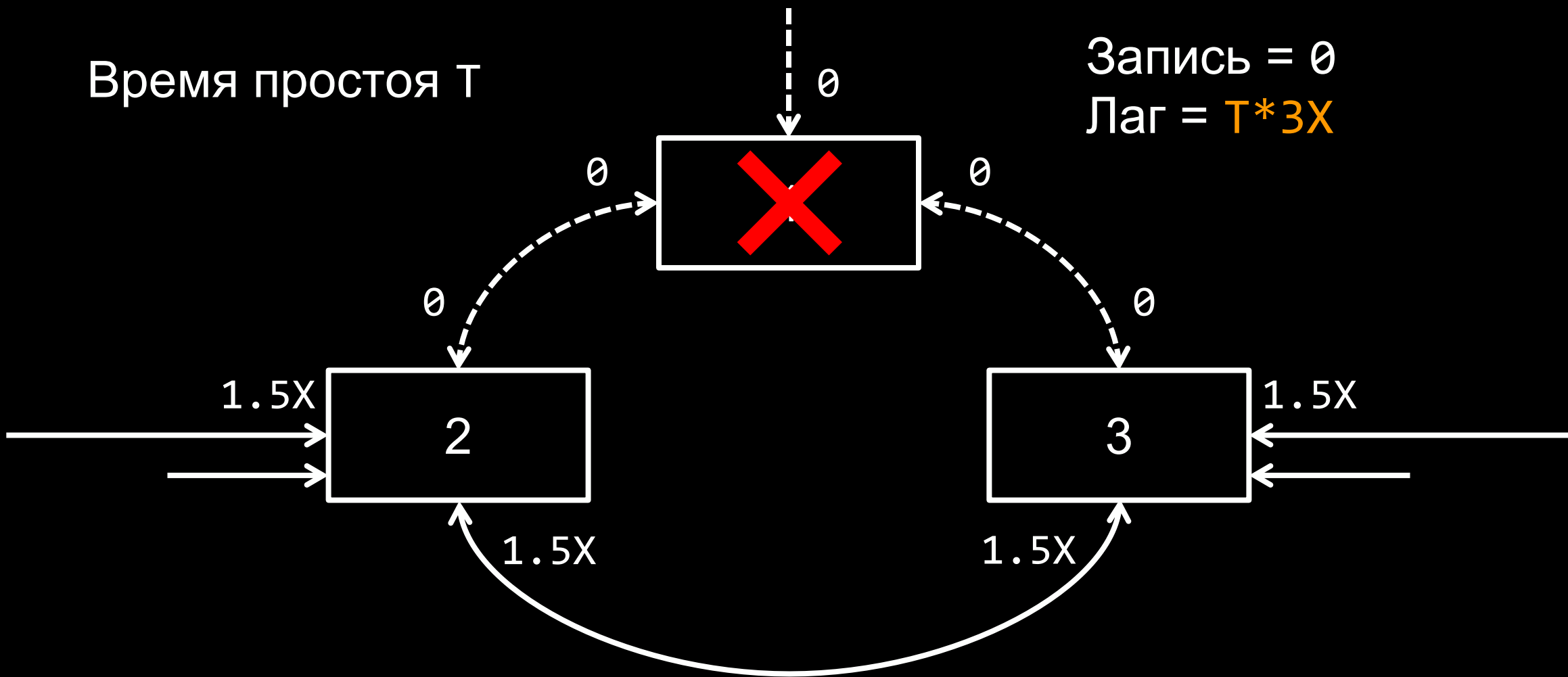
Кafka в один конец



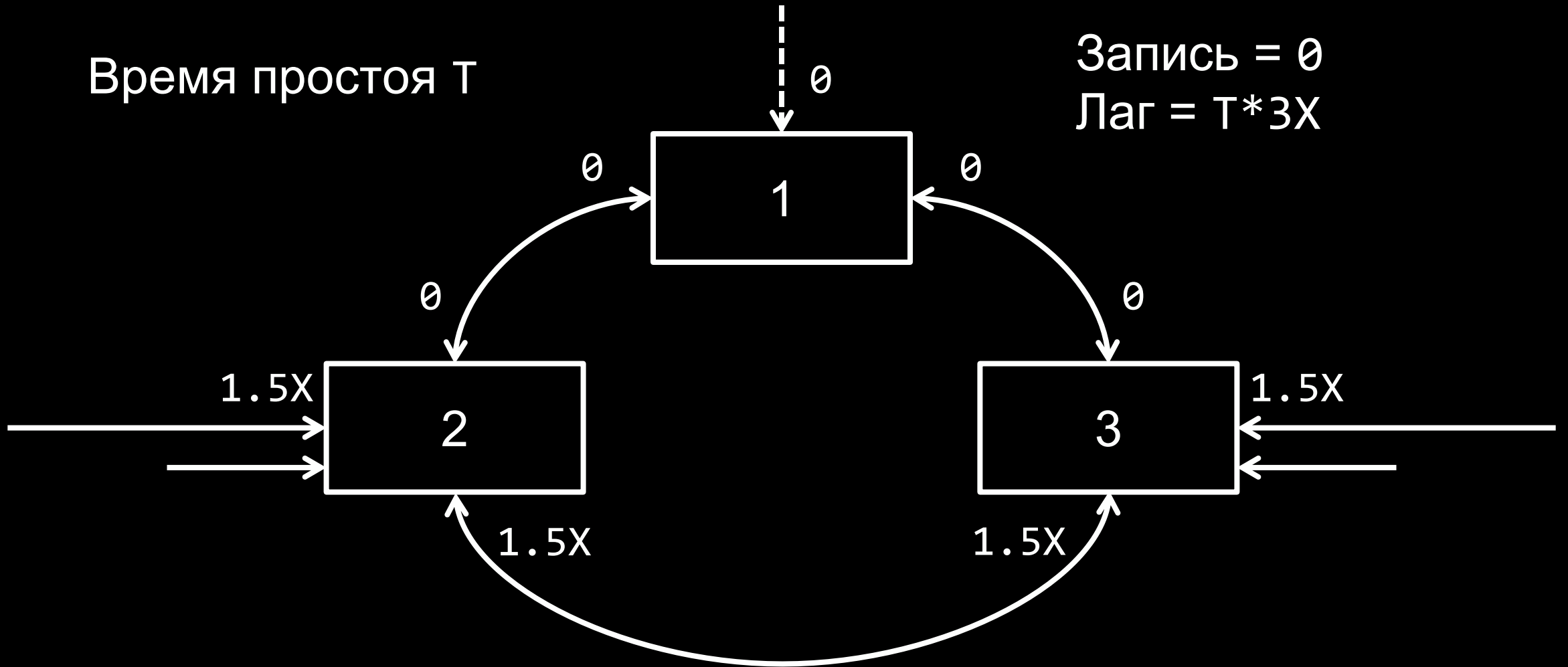
Кafka в один конец

Время простоя T

Запись = \emptyset
Лаг = $T * 3X$



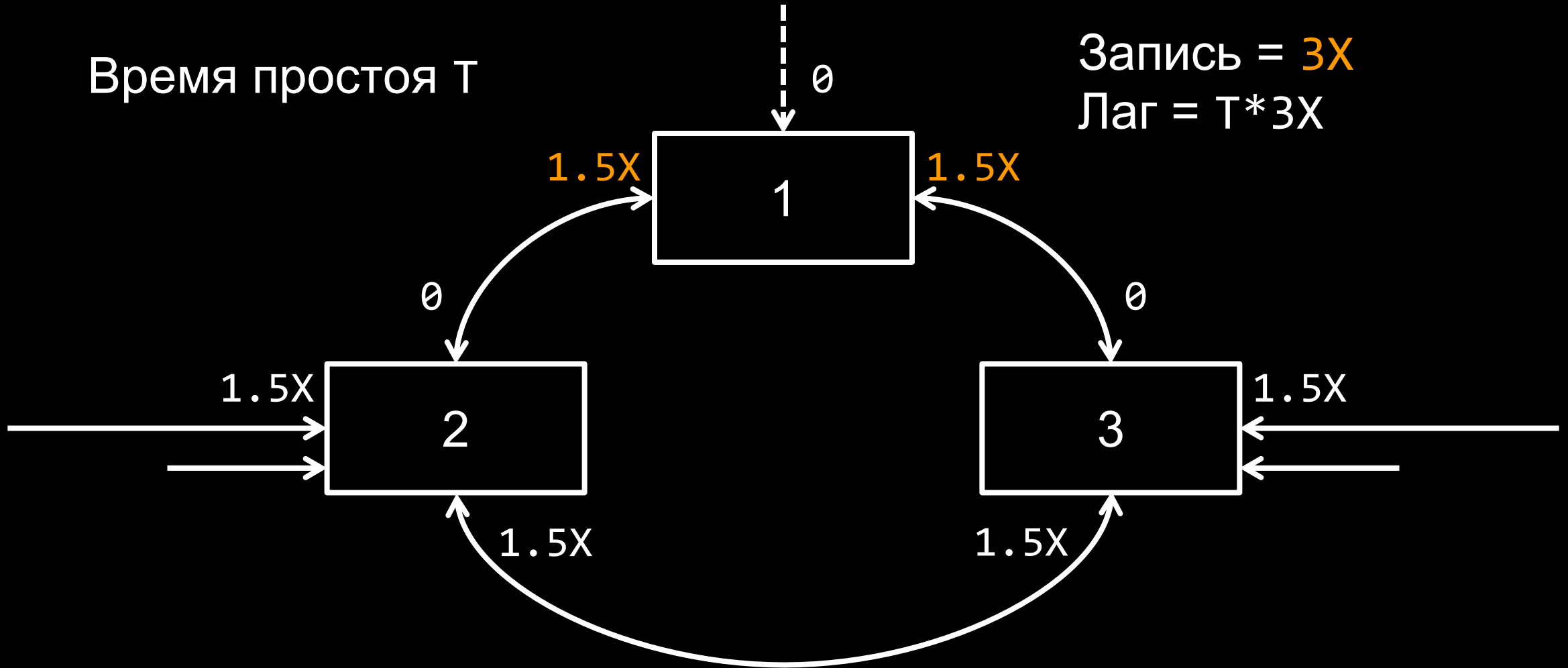
Кafka в один конец



Кafka в один конец

Время простоя T

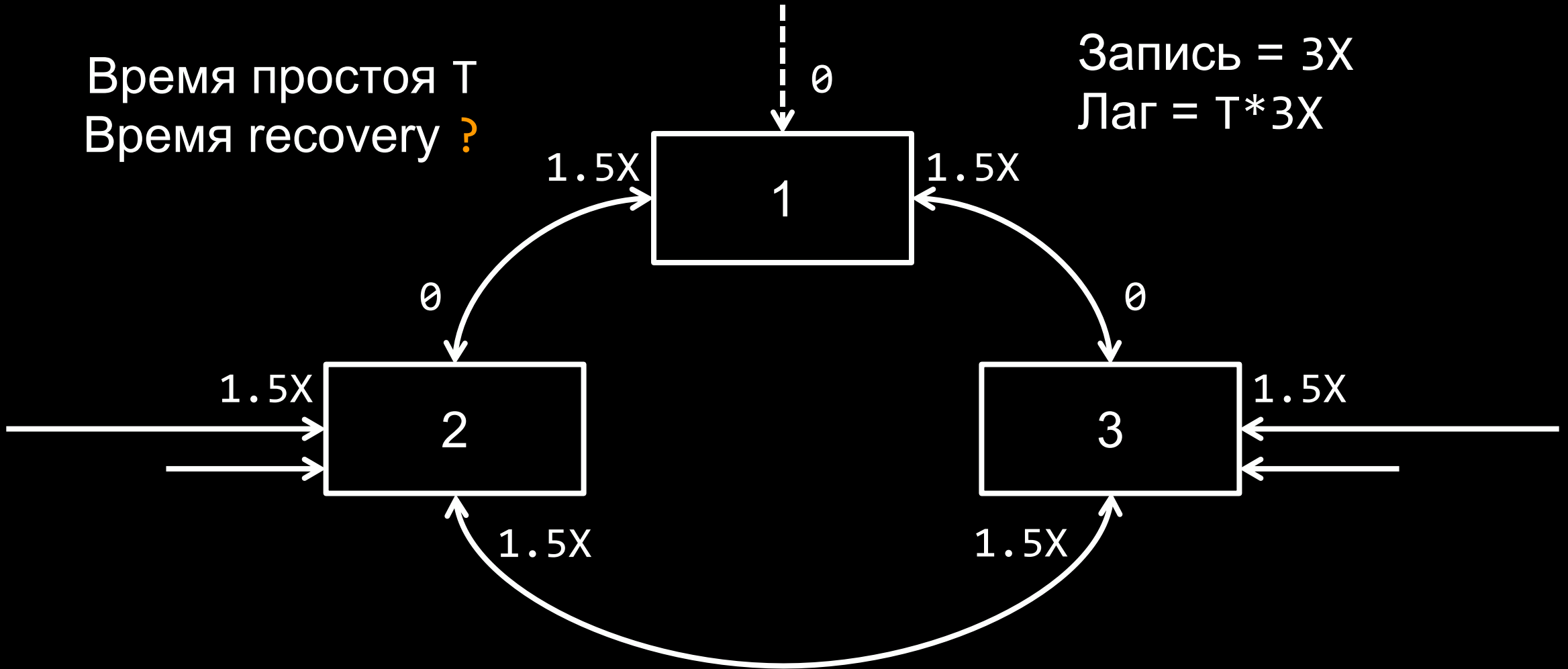
Запись = $3X$
Лаг = $T * 3X$



Кafka в один конец

Время простоя T
Время recovery ?

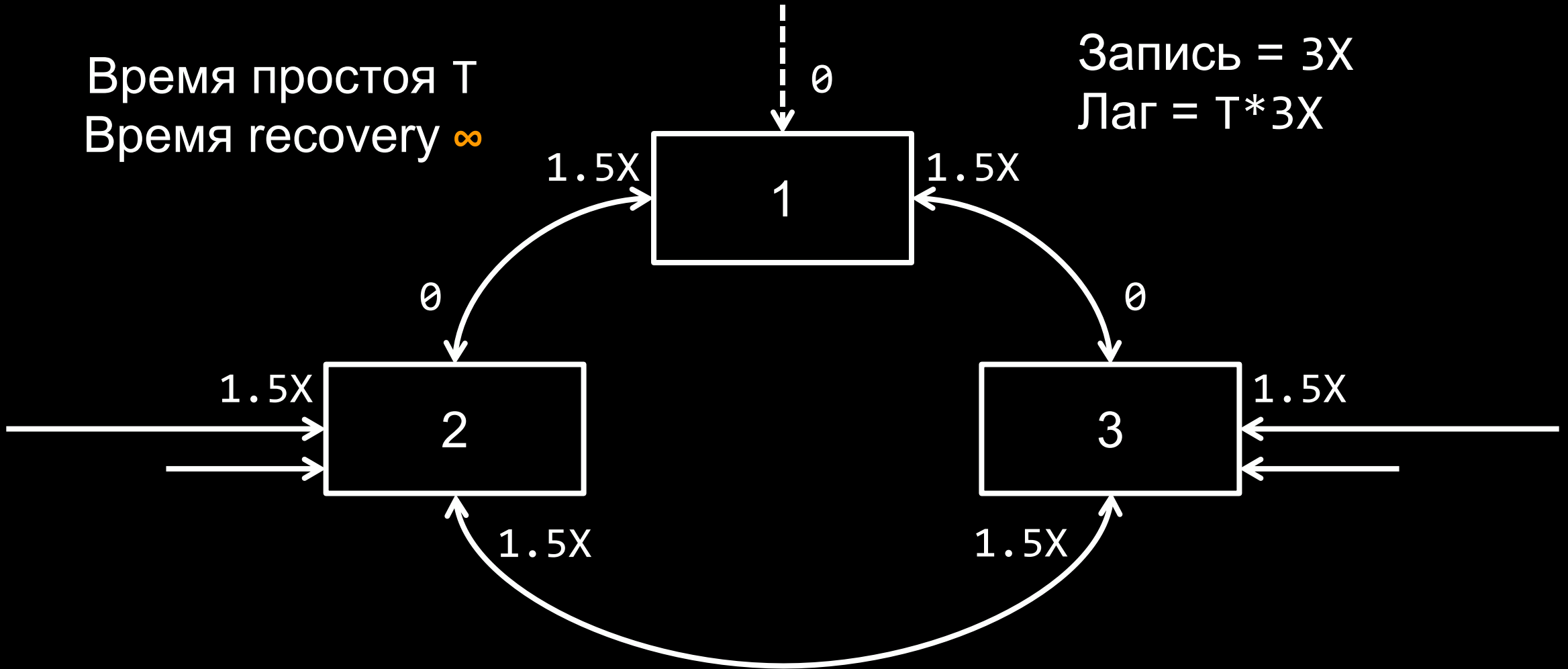
Запись = $3X$
Лаг = $T * 3X$



Кafka в один конец

Время простоя T
Время recovery ∞

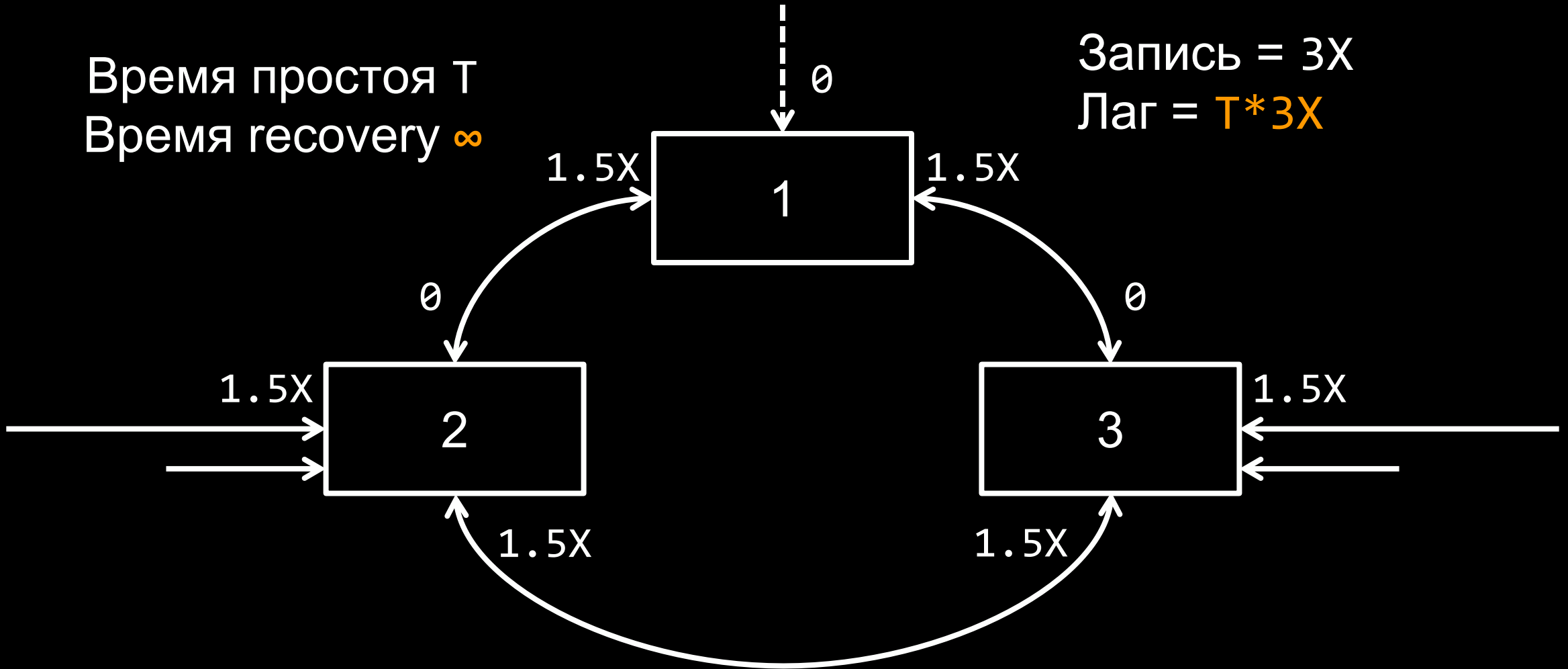
Запись = $3X$
Лаг = $T * 3X$



Кafka в один конец

Время простоя T
Время recovery ∞

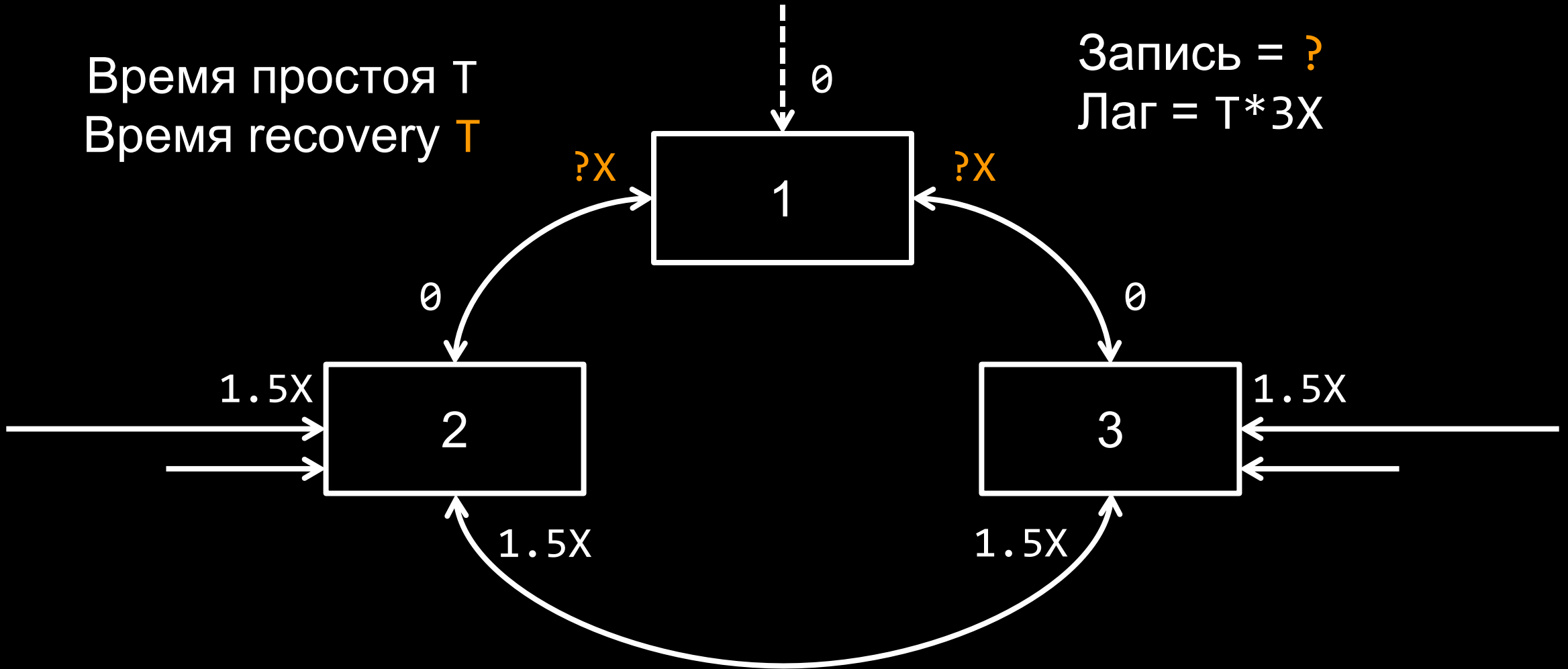
Запись = $3X$
Лаг = $T * 3X$



Кafka в один конец

Время простоя T
Время recovery T

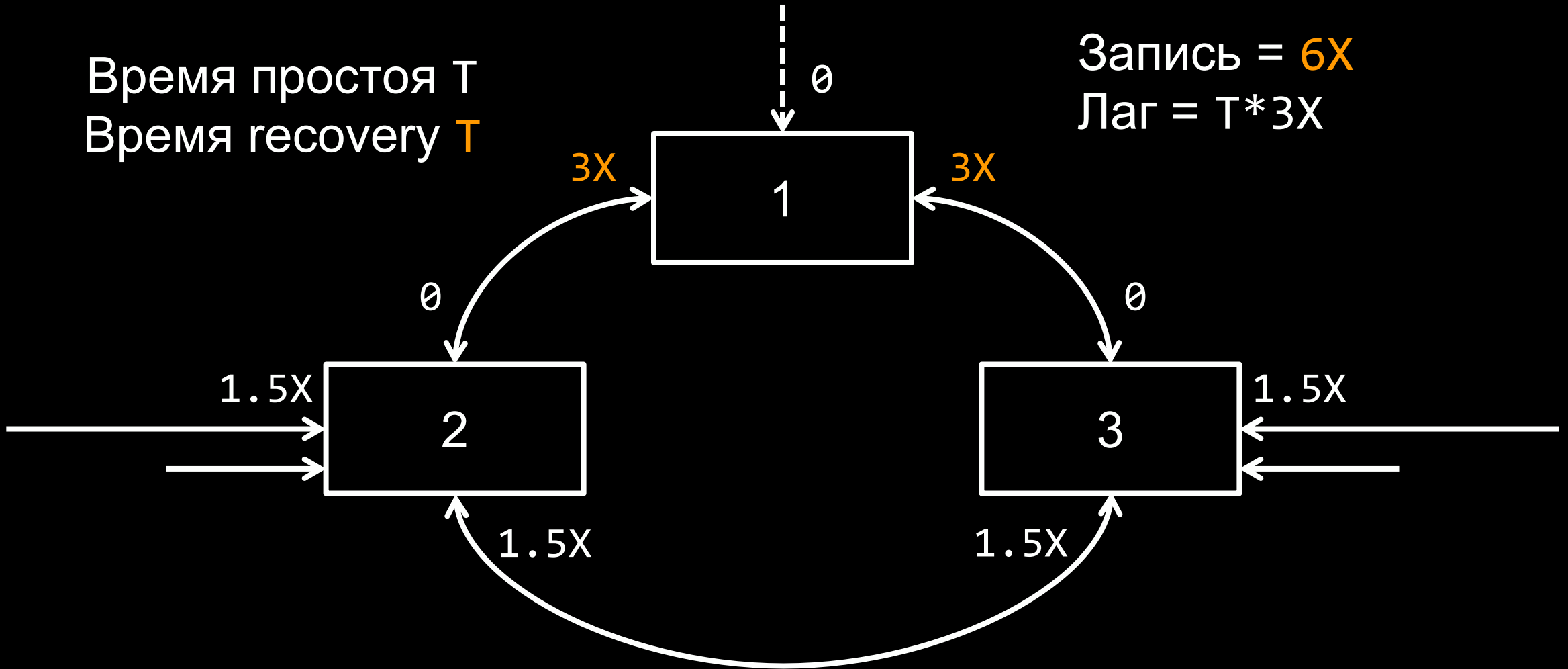
Запись = ?
Лаг = $T * 3X$



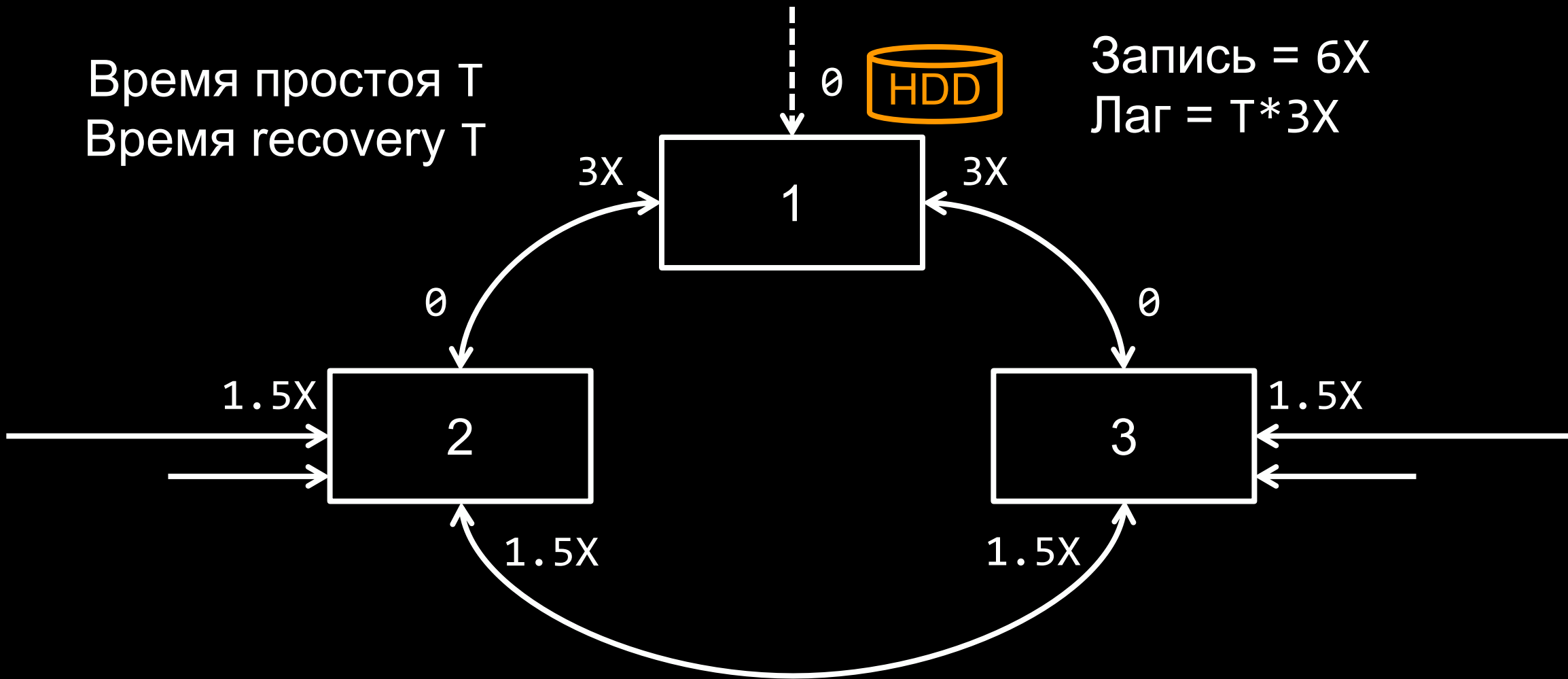
Кafka в один конец

Время простоя T
Время recovery T

Запись = $6X$
Лаг = $T * 3X$



Кafka в один конец



Метастабильные отказы

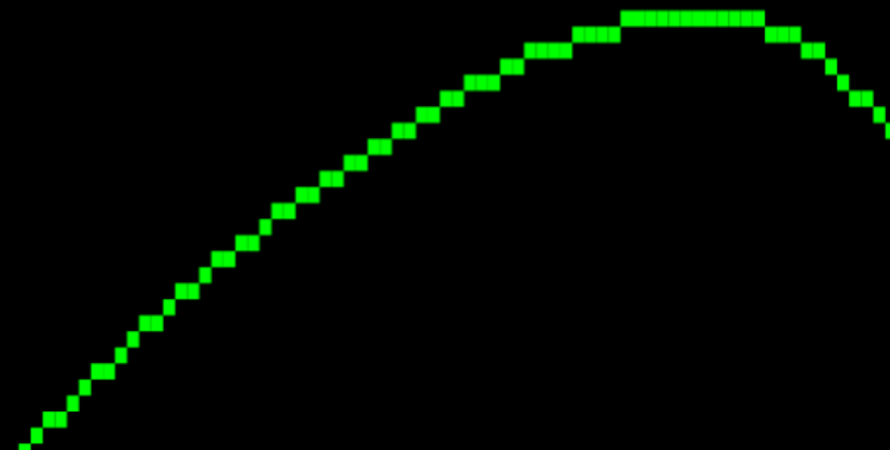
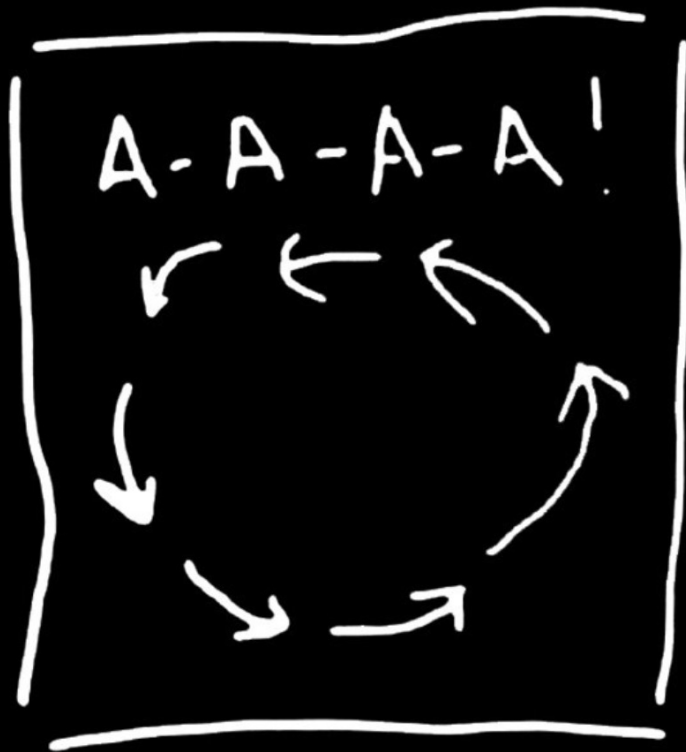
Выводы

— Удерживающий эффект важнее триггера

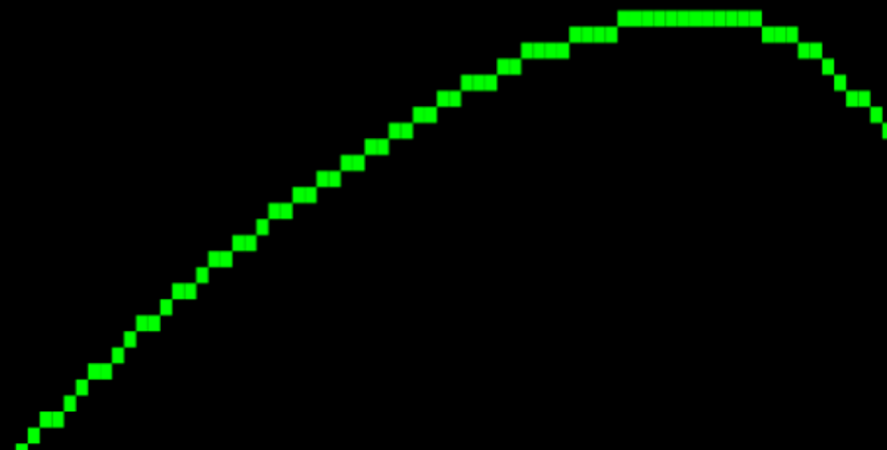
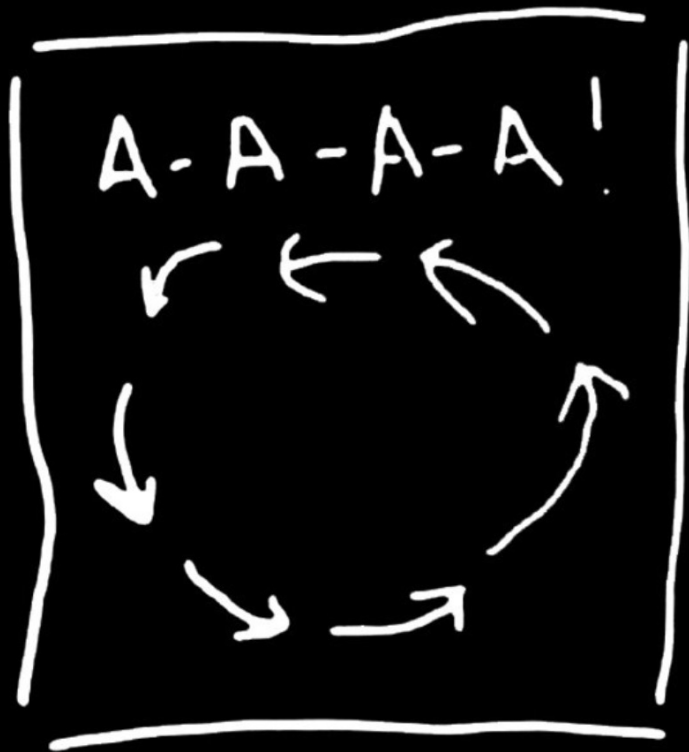
* Стресс-тестирование

* Возможность значительного снижения нагрузки

Выводы



Выводы Заблуждения



Заблуждения

Заблуждения

8 заблуждений о распределённых вычислениях L. Peter Deutsch и James Gosling, Sun Microsystems, 1994-1997

1. Сеть надёжна
2. Сетевая задержка равна нулю
3. Пропускная способность не ограничена
4. Сеть защищена
5. Топология сети не меняется
6. Есть один администратор
7. Транспортные расходы равны нулю
8. Сеть однородна

Заблуждения

8 заблуждений о распределённых вычислениях
L. Peter Deutsch и James Gosling, Sun Microsystems, 1994-1997

+ 3 заблуждения

Mark Richards и Neal Ford,
«Fundamentals of Software Architecture», 2020

- Версионирование — это просто
- Компенсационные обновления всегда работают
- Observability опциональна

Заблуждения

8 заблуждений о распределённых вычислениях
L. Peter Deutsch и James Gosling, Sun Microsystems, 1994-1997

+ 3 заблуждения

Mark Richards и Neal Ford,
«Fundamentals of Software Architecture», 2020

- Версионирование — это просто
- Компенсационные обновления всегда работают
- **Observability** опциональна

Заблуждения

5 заблуждений о факапах

Григорий Кошелев, DevOps, 2024

Заблуждения

5 заблуждений о факапах

Григорий Кошелев, DevOps, 2024

Можно избежать всех факапов

Заблуждения

5 заблуждений о факапах

Григорий Кошелев, DevOps, 2024

~~Можно избежать всех факапов~~

Заблуждения

5 заблуждений о факапах

Григорий Кошелев, DevOps, 2024

1. Можно не искать корневую причину, если факап устранён
2. Человеческий фактор можно контролировать
3. Два одновременных факапа невозможны
4. Изменение конфигов — это безопасно
5. Observability — это просто

QA

Другие доклады и материалы:

https://tg.me/chn1_GregoryKoshelev



Байки для дискуссии



Шатали тачку —
сломали весь прод

Kafka в один конец —
подробности

Каскадная
нагрузка

Rate limiter на
Docker Hub

Enterprise Kafka