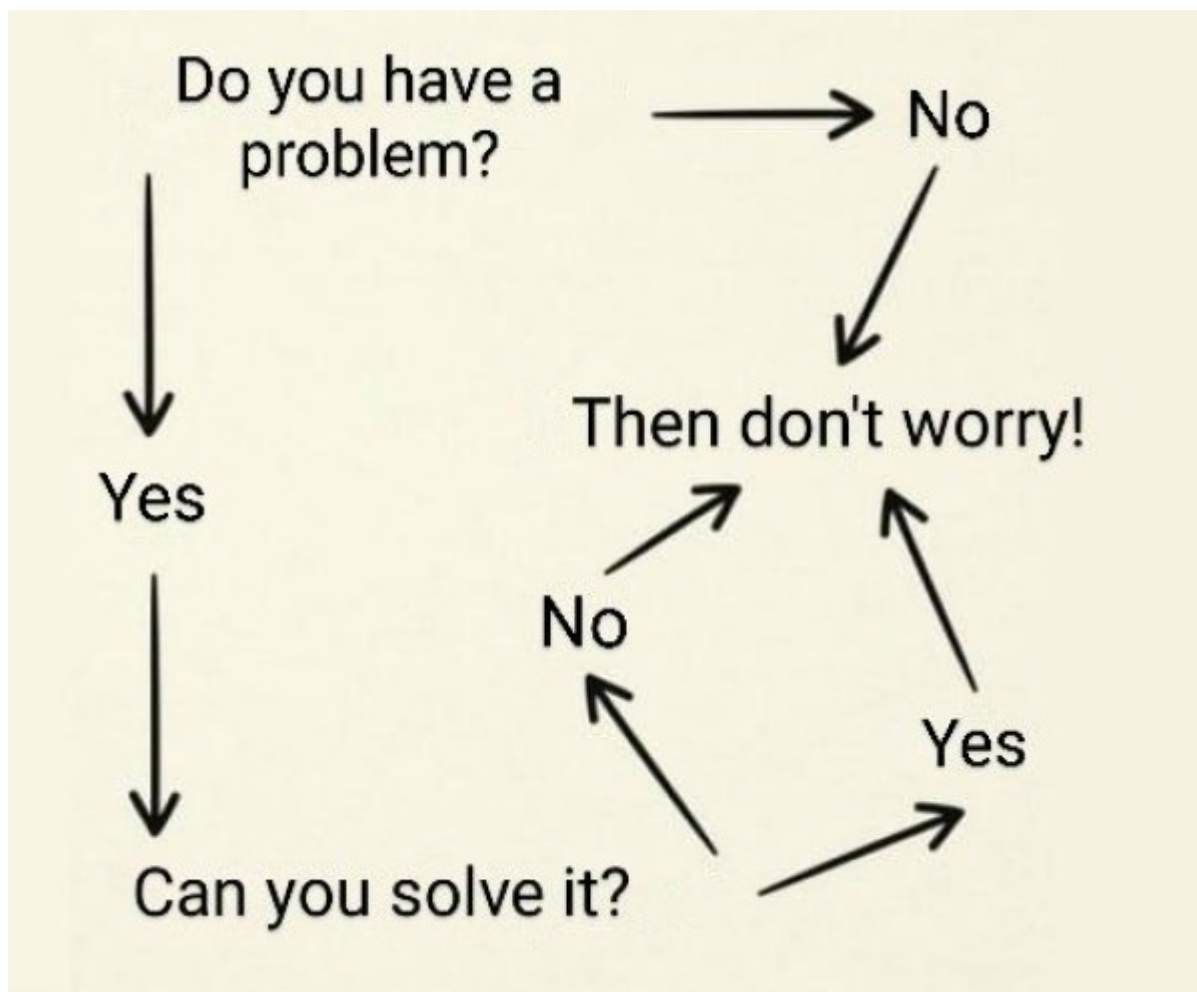


DevOps: Как разгребать проблемы

Вводная: Что-то пошло ну очень не так и ситуацию можно описать как «шеф, все пропало» или «мы в крутом пике». Т.е. это не рядовой факап, а прямо катастрофа высокого уровня.

Задача: Вылезти максимально быстро и желательно с минимальными потерями из данной ситуации

Пути выхода из ситуации и решение проблемы



1. Первое что нужно сделать - остановиться и понять в какой ситуации мы оказались. Казалось бы - горит и даже сильно уже припекает, надо руки в ноги и тушить. Увы, скоропалительными действиями можно нанести еще больший вред. Поэтому останавливаемся, убираем руки от клавиатуры и оцениваем ситуацию. Здесь главное ответить на вопрос о масштабе события: насколько оно глобальное, имеет ли тенденцию к ухудшению, насколько быстро все доломается окончательно и т.д. Самое сложное - понять насколько тяжела ситуация, т.к. лавина начинается из-за небольшого камешка, а уже через несколько минут ее невозможно остановить. Так же важно зафиксировать все начальные условия: что и когда приключилось, начать вести хронологию событий. Если ваша команда практикует написание Post Mortem'ов - самое время создавать новый.

2. Подключение ресурсов (если они у нас есть). Как правило, тяжелая ситуация подразумевает невозможность сделать все целиком самому: отсутствие компетенций во всем подсистемам, недостаток времени и сил, возможно требуются дополнительные ресурсы (люди, железо и прочее). На этом этапе необходимо подключать руководство, коллег, дополнительные мощности. Задача в распараллеливании задач и подготовке к дальнейшим действиям. Если вам прямо сейчас не нужна помощь - это хорошо, но чтобы эта самая помощь была в ближайшем будущем надо ей озаботиться уже сейчас (например, иногда сложно разбудить человека или он недоступен в текущий момент или просто далеко от компьютера). Так же на данном этапе необходимо провести назначение единого ответственного, к которому стекается вся информация и решения. Как правило на этом этапе уже возникает некоторое понимание масштабов и можно провести первичную оценку времени по восстановлению работы, как минимум фраза «потребуется примерно 2 часа на устранение» звучит намного понятнее чем «да хрен его знает».

3. Разгребание проблемы. Выполняем пошагово, синхронизируя свои действия с командой и коллегами. Конкретную работу должен делать кто-то один, не должно быть картины «мы вдвоем подключились к серверу и параллельно пытались починить» - как правильно это приводит к проблемам, т.к. каждый из участников получает искаженную картину. Если на данном этапе открываются новые обстоятельства (а в 95% это так и случается) - производим переоценку ситуации, и, соответственно, время на ее устранение. Если стало понятно что требуются дополнительные затраты (времени, людей, ресурсов) - возвращаемся к пункту 2. Очень важно понимать что гораздо лучше честно сказать «ситуация изменилась, произошло то-то и то-то, время на восстановление увеличивается по нашей оценке до XX часов» чем пытаться впопыхах уложиться в первично обозначенное время, мол «пацан сказал - пацан сделал». На этом этапе главное не накосячить еще больше из-за спешки и не привести ситуацию к варианту «сами угробили все до чего дотянулись руки».

4. Стабилизация обстановки. Вроде все понятно по текущей ситуации, нужные люди устраняют проблемы. Освободившиеся ресурсы можно переключить на вторичные задачи: помощи наиболее загруженным специалистам, исключения повторения инцидента, общение с руководством/клиентами, ведение Post Mortem'a, или вообще отправить отдыхать если они больше не нужны.

5. Проведение анализа событий. Здесь основная задача - оценить что нужно сделать для исключения повторения инцидента, запланировать и провести работы. Разумеется, все это должно быть отражено в базе знаний и т.д.

[решение инцидентов](#), [все в огне](#), [горим](#), [паника](#)

From:

<https://wiki.rtzra.ru/> - RTzRa's hive

Permanent link:

<https://wiki.rtzra.ru/devops/devops-how-to-solve-problem>

Last update: **2023/10/06 17:36**

