Become a hero firefighter at your software startup and boost team morale.

Tactical Guide

Problem

Working at a startup has lots of highs and lows, its common to have lots of fires along the journey, long weekend hours, last minutes code changes, production mishaps and all sorts of unidentified issues. In such intense work environments, few people are willing to work the on call and be the go to person for the difficult problem solving.

Firefighting skill set is valuable especially in organizations with complex products and high downtime cost. Performing diagnosis of an issue under pressure is stressful and demanding, therefore these positions are difficult to fill.

Would you consider taking on such a position and in process become a hero to your fellow team members?

Let me share with you how you can get started and what strategy to apply to become a super effective firefighter.

Solution

Let me share with you some tactics to :

- learn firefighting
- boost team morale
- become a hero

The key to success is to :

- say yes to on call, its sink or swim time, don't have any fear, if you are new then you will be paired up with an experienced engineer
- Identify parts of the system prone to highest rate of failure, then study the code and learn how to test all related components so that you have hand on knowledge of how they should behave when NOT working
- understand possible exceptions, that's the key to rapid diagnosis under pressure
- after a crisis is resolved step up and own the implementation of a permanent fix

Outcomes

Being on point for high pressure situations forces you into a rapid learning mode, your ability to comprehend various systems and tools will grow at an exponential rate, soon you will be able to stop guessing what is happening and instead actually know what's happening, and of course every problem solved becomes a story to talk about during your review as you negotiate your pay rise.

Your willingness to face issues head on will enable the rest of the team to focus on their work. The team morale will rapidly improve as a result and you will quickly become the hero you were always meant to be.

Your leadership will also be welcomed by the higher level management.

Become a firefighter!

First step to becoming a firefighter is to say YES to **on call** and **deployments**. Don't fear this, the rest of the organization will not let you fail, if you are new someone senior will be in charge and will provide guidance.

Focus on attention to detail, make sure to read any existing documentation regarding issues identification and resolution, this is especially true in larger technical organizations with lots of existing systems in place. Often, operational playbooks and issue resolution documentation is out of date and will need updates, that's great, own this and make the changes after handling an issue. In an early startup you may be forced to be the one writing this documentation which is also an excellent experience.

Please document everything, **how** the issue occur, **who** had to do **what**, **when** the issue occurred, **where** and **why** it happened. This will be the basis for developing skills around rapid issue diagnosis and resolution.

System Analysis

Becoming effective at firefighting requires proper understanding of the systems you are responsible for. You should develop comfortable understanding of the architecture, all the relevant system inputs, outputs, internal processing components, user capabilities and end to end testing procedures.

Focus on the parts of the system prone to highest rate of failure, if possible study the source code, unit and integration test and learn how to test all related system components so that you develop hands on knowledge of how they should behave **when NOT working** in addition to normal operating behaviour.

Understanding possible exceptions, how to identify and resolve them is the key to rapid issue diagnosis under pressure when system is down.

Validate your understanding by executing hands on testing in lower environments (dev/staging), also connect with the original engineers to review your newly developed body of knowledge. Ask questions to validate your knowledge the who, what, when, where, why and how in the context of the system.

Own the Fix!

As soons as the crisis has passed, step up and own the implementation of a permanent fix. If applicable, think long term solution that will permanently resolve the issue. Plan the work to implement source code fixes, deploy proactive monitoring, enable self healing, testing, update technical documentation, operational playbooks and train fellow team members.

Consider how the permanent fix impacts people, process and technology in the organization, each area has different risks, rank the risks and benefits as you see fit before deciding next steps. Analyze components of a proposed fix to identify those parts that can be delivered quickly to provide majority of the benefit, focus on the 20% of the work that delivers 80% of the value (Pareto principle).

Ownership mentality is one of the most sought after character traits and is desperately needed in all organizations. Own the fix, if possible apply corrections yourself to enable the rest of the team to focus on their work.

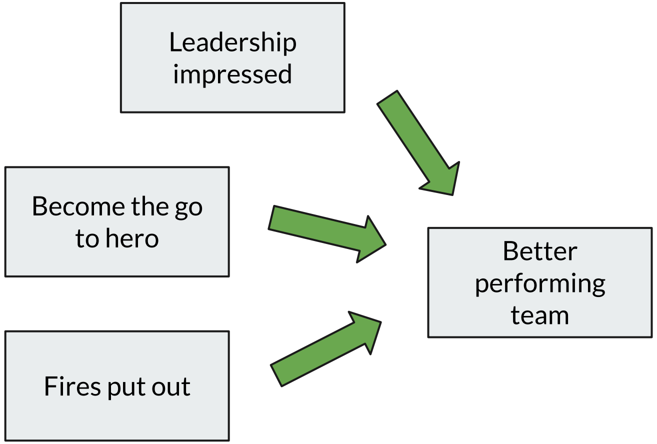
CONTROL THE OUTCOME

Now we celebrate!

How does this help me?

This approach brings about a WIN-WIN-WIN situation for the product team, organization and you.

- The team is enabled to continue to produce work with minimal interruptions to their schedule.
- Management can observe beneficial impact of your work across the team and the organization and will love your ownership mindset.
- Your willingness to be the firefighter and owning the fix is recognized and very much appreciated by your fellow team members, in effect you become indispensable.



Leadership impressed

Become the go to hero

Fires put out

Better performing team

Questions?

Filip Szalewicz
fszale@gmail.com