# https://upload.wikimedia.org/wikipedia/commons/a/ac/Obama_and_Biden_await_updates_on_bin_Laden.jpg

# **IT War Rooms**

# Abstract

In every industry failures happen, but only in IT we have so much incidents which affect so much people. We could list also hundreds of reasons why this is happening so often. This time although let’s focus on the actions which should be taken if something occurred and now, we are facing serious production issue.

# Failure

A lot of applications are deployed manually, during this process people make mistakes. Rest of the solutions are deployed automatically and, in those scripts, have bugs. For most of those situations we know how to behave. Repeat steps, make code correction and push again to production. The most difficult cases are when we are facing completely new issue which we don’t know what to do with. Serious failure is a situation in which something went wrong, we don’t have the simple answer how to repair it. Usually it also affects a lot of people. For example

* application not starting at all without any information why
* application starts, but after some time it crashes
* deployment went smoothly, application is working, but pushed code were not properly tested, and every second it removes part of data from database

# War Room

When failure detected, it become escalated until person which think is able to manage the process of resolving issue is found. It could be release manager, engineering manager or even IT director. The important here is that this person needs to be proper placed in the organization, this means that employees needs respect him, and he needs to have authority to make decisions. Let’s call him a **Leader**.

Leader needs as soon as possible gather all people in one place which he thinks will help in resolving the issue. List of people will be different for each application structure.



If we have very simple application like presented above, we need to have a lot of people to help

* Leader
* Developers – architect, technical leaders, senior developers
* Network engineer
* Operations engineer
* Database administrator
* Security engineer
* Tester
* Recorder engineer

As you can see every application could require group of employees to resolve the issue. (Think about more complex system, when we have multiple elements like Mainframe with core banking system, integration layer and external systems which could influence behavior of application)

# Action plan

After gathering all people in one room **Leader** again ask to describe found issue. It is important to align all people and make sure that everybody understood what we are dealing with. Next, **Leader** listen to propositions how to resolve the issue. All of them should be written in some visible place (wall, whiteboard).

After this discussion about each of them starts, during the conversation required people for each action should be marked on the whiteboard. So, if one proposition requires to reproduce issue on other environment and it needs QA, Operations, and Database engineers all three should be pointed on the wall.

First part of the discussion should finish with list of the most probably actions which could resolve the issue. Next part should assign the tasks to concrete people. This agreement also should be written on the table. It is very important to know who exactly will be working on which task. It is also important from the responsibility perspective. If someone is clearly pointed that this task is depend on him, he will be more motivated to work on it hard.

If possible, work should be planned to be done simultaneously. Very often we could work on the most probably issue on the production and try another approach on the test environment. Not running tasks at the same time, it is a waste of time.

After assignments leader, should establish next **status**. It is not a meeting as all people should all the time sit in one place and collaborate. It is a moment when everybody should stop working and rephrase and summarize everything what they have done. Those moments are important as if people start working on the tasks it could happen that they will turn in wrong direction and in the stress situation no one will ask questions which could show that performed actions could be wrong.

During the status again we are going through the list which is visible for everyone. After one hour of work this could be already heavily crossed, or with new ideas. During status everyone should synchronize and be sure that they understood what is on the wall.

We are repeating the steps until issue resolved.

## Recorder engineer

Most of the roles are pretty much straight forward. We need database administrators to help us restore database, restart instance and other things, network guys to turn on and off servers, redirect traffic to chosen instances and others.

One role could not be so clear. Record engineer it is a person which should wrote down every action which was performed. It is done for two reasons

* When difficult issue is found, a lot of ideas are tried. To not waste time for implementing the same solutions which hadn’t have worked before, we need to have written exactly what we have done previously
* After the incident analysis of the situation should be done and all we should check if any of performed action shouldn’t been reverted. Without clear list it could be difficult to remember about every change done.

Report engineer should have basic knowledge about development infrastructure and security as he needs to know what he is writing. He needs understood everything what people in the room are talking and if something is not clear he must ask questions to make it obvious.

Below you can find example of the report which I was creating during outage.

# 

# Pizza time

Very rarely production exception happens during working hours and it is resolved in normal working time. Usually failure means that all people involved will be staying long to resolve failure. During couple of the war rooms which I organized, after couple hours of working, people’s concentration decreased. And this moment pizza always helped to keep employees involved and focused on the issue. Pizza time also helps to look on the issue from other perspective and integrate all people involved.



# The day after and memories

If war room was successful, the day after is always very busy one. Leader needs to create detailed report and perform meetings to spread the knowledge about the issue. One for the business, one or more for the technical people to analyze the situation and learn. Leader also should identify what should be done to protect company against this kind of issue next time and assign tasks to proper group.

The celebration meeting when all people which took part will be rewarded should be also held. Reward could be oral one, but it is important to appreciate employee’s involvement.

It is important to behave correctly the next day to keep memory of the war room positive, and it will make people be more engaged and integrated. From my experience, people which were working together on the issue late night are rather remember this as nice adventure and they are coming back to this moment with good feelings rather than bad ones.

# Additional questions

## How long War room should take place?

As always it depends. One time I was leading the war room which last till late morning hours next day. Another time, we stopped war room after 8 pm and come back to it next morning.

## What kind of room and tools I need to lead it?

When you read on the Internet about the war rooms it is often described that you should make sure that room is proper size, chairs are comfortable and all required equipment is in place. In reality you need to take first room available and put everyone there. People will manage, if everyone are involved and understood that issue is serious probably, they will be working the same efficient in nice meeting room and in the elevator.