Result Transmission System (RTS)

What is RTS?

For the first time in election history, Pakistan used a Results Transmissions System (RTS) — an online system to relay results from 85,000 polling stations around the country to the central headquarters in Election Commission of Pakistan.

Here's how it (should have) worked. The Presiding Officers (PO) were supposed to use an app to enter results and take a picture of Form 45. The data would then be transferred to the ECP server… and that's it. Really.

What seems rather basic from a technical point of view became a point of contention as RTS went down at 9 p.m. delaying results.

RTS system failure

There has been a huge furor in the media in the country over the failure of the Engineering Council RTS or the Result Transmission System on election day. The results of the elections have been marred by continuous allegations of rigging, manipulation, and fraud. Opposition parties seem to be united in rejecting the results of the elections. There have been several calls for an inquiry into who is responsible for the crash of the RTS during a critical time on Election Day.

The system was meant to allow speedy transmission of results from polling stations to returning officers and ultimately to ECP headquarters. An Android app was developed for all presiding officers who have digitally transmitted form 45's to ECP headquarters via an intermediate server. The system had been pilot-tested in January of this year and a deal was signed with the Election Commission of Pakistan in February to develop an electronic data transmission system for the upcoming general elections in 2018.

Most international observers to the general elections 2018 in Pakistan have said that the polling had been "transparent and fair". Nevertheless, it is imperative that the entire saga of the failure of the Result Transmission System be investigated and the culprits behind the crash, if any, need to be brought to trial.

On the day of elections, the entire nation was glued to their TV screens as the results came flooding in. Then suddenly, about 15 minutes before midnight on the 25th of July, the system crashed. The Secretary of the Election Commission of Pakistan held a press conference detailing how there had been a technical failure due to which there would be inordinate delays before the declaration of the official results of the general elections. It took days before we came to know the complete results.

As soon as the technical failure was announced, leaders of political parties who had been losing the elections up till that point told the media they reject the results of the elections because they saw this technical failure as further evidence of rigging in the elections. Shahbaz Sharif, leader of the PML-N, was among the first to reject the election results, even before they had been officially announced by the ECP.

RTS Was Far from a Perfect System

There were question marks about the system even before the elections. According to local media reports, Returning Officers (RO) from Faisalabad, Okara and other cities wrote to the ECP and expressed concerns about RTS.

According to the Express Tribune, complaint letters were received from multiple ROs.

An RO said:

> ***I have the apprehension that in such circumstances your RTS system will not perform on the poll day.***

An unnamed NADRA source cited by the publication claimed:

> ***Due to possible failure of the RTS system, serious issues related to transparency may arise on the night of poll day.***

RTS Was Designed by NADRA

RTS was developed and its usage was made mandatory while PML-N was in power, through a law that was passed by the previous government.

It was developed by NADRA, who have expertise with large-scale apps. This is not the first time the institution has developed apps on such a scale so it's unlikely that technical abilities were a factor in the failure of the system.

Not to mention, ECP and NADRA knew the system scope from the very start: 85,000 endpoints would be submitting data and that makes a developer's job easy.

If they know the load, they can test and optimize for it. This means that all possible outcomes were known to the developers even before they started working on RTS. It's usually a simple task to design a system where all variables and constraints are known, compared to a system that has to deal with unknowns.

Why Hasn't the System Come Back Online?

The worst happened and the system went down. Fine. But does it really take hours to restore it? Even if the system was designed poorly and the load became unpredictable, absolutely not.

In the time since the system went down, completely new server-side solutions could have been made from scratch and launched – considering the nature, emergency and scale of the problem.

If an entirely new solution could be designed, developed and launched from scratch in the time RTS has been offline, you can imagine how much time would be required to restart services and put the system back online: merely a few minutes.

The lack of a contingency plan on part of ECP and NADRA is simply baffling, if not downright criminal.

So what happened to RTS? There are a few possibilities here:

- People using the system were unfamiliar with it to the point that they didn't know what to do in case of failure
- The system was targeted by a malicious entity with a DDoS attack
- **It wasn't brought back online on purpose**

We should mention here that mitigating for DDoS attacks is quite easy these days. ProPakistani suffers DDoS attacks from within Pakistan, India and abroad and with a limited budget, we suffer near zero downtime.

Even if all 85,000 end-points were connected and submitting data at the same time, it should not have caused the system to collapse. We are aware of websites, apps, and servers within the country which serve hundreds of thousands of concurrent connections so technical failure simply does not make sense.

And even if it did collapse, it does not take so much time for it to be restored.

Possible Scenarios in Light of Failure of RTS

The solution is quite simple: conduct an audit.

An audit for a mobile app and a system like RTS isn't a complex task. If Pro-Pakistani were given access to the servers, it could determine what went wrong within hours, if not minutes. It's not a testament to our aptitude; rather, it really is that simple.

We want to emphasize that with the government's resources, it's child's play to determine why RTS failed.

After the audit, there could be three possibilities.

1 – RTS Failure was Intentional

The failure was intentional and the system was sabotaged by the outgoing government in case of PTI taking the majority.

Remember, this is the first time RTS has been used in Pakistan. It was designed, developed and made mandatory when PML-N was in power.

As soon as a clear lead was established, the system was shut down to create controversy and unite the losing parties against Imran Khan.

2 – RTS Went Offline Due to High Load

RTS went down unintentionally and if that's the case, then it is gross incompetence so severe that it can be classified as a crime against Pakistan.

The responsible people and departments should be investigated, tried and put in jail for messing up a system so critical to the country's future.

3 – Outside Involvement in Putting RTS Offline

Considering Indian involvement and stakes in Pakistan election, we would not want to rule out the possibility of outside interference in elections via attacking RTS through a DDoS attack.

But then again, it's not that hard to plan against DDoS attacks. There are websites (such as banks, social media websites and so on) which are routinely attacked and never fail.

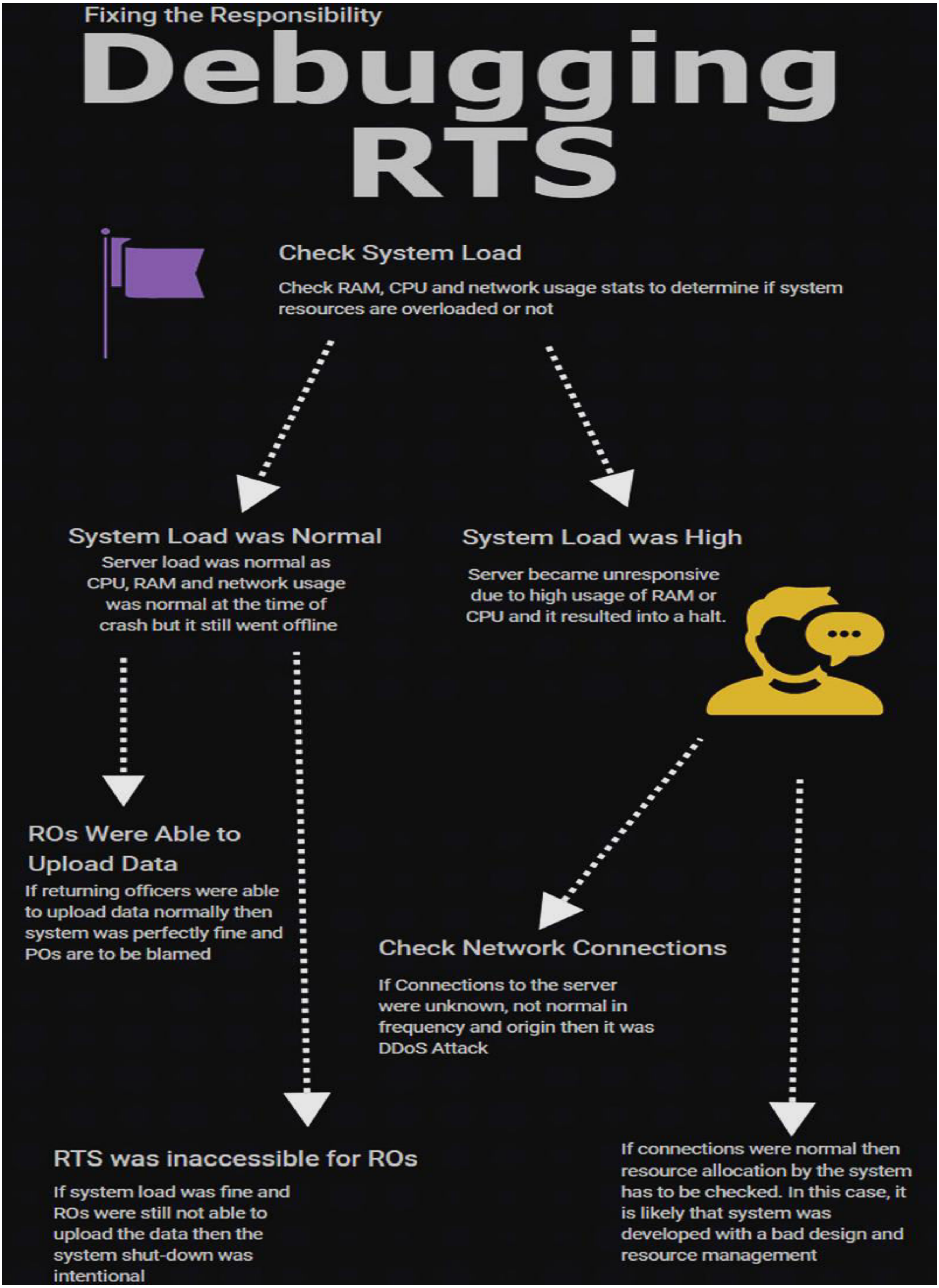
If it's an outside job, the people at the core of app design and deployment should be fired immediately with heavy fines at the least.

Finding Out Who is Responsible

If you think that a JIT and several months are required to determine responsibility for the catastrophic RTS failure, we have good news for you.

Unlike other fields, debugging computer systems and determining the cause of failure is straightforward. A single expert (or more if you want to be completely certain), can determine the cause within minutes, if not hours.

Here’s how it can be done:



Fixing the Responsibility

Debugging RTS

Check System Load

Check RAM, CPU and network usage stats to determine if system resources are overloaded or not

System Load was Normal

Server load was normal as CPU, RAM and network usage was normal at the time of crash but it still went offline

ROs Were Able to Upload Data

If returning officers were able to upload data normally then system was perfectly fine and POs are to be blamed

RTS was inaccessible for ROs

If system load was fine and ROs were still not able to upload the data then the system shut-down was intentional

System Load was High

Server became unresponsive due to high usage of RAM or CPU and it resulted into a halt.

Check Network Connections

If Connections to the server were unknown, not normal in frequency and origin then it was DDoS Attack

If connections were normal then resource allocation by the system has to be checked. In this case, it is likely that system was developed with a bad design and resource management