EQUAL EXPERTS

Register to Vote Website

Lessons Learned Review for the Cabinet Office

Authors: Equal Experts UK Ltd

Introduction

You have asked Equal Experts to look into the issues that recently affected the Register to Vote service, (the online element of Individual Electoral Registration, or IER) as described in the Terms of Reference issued by GDS on the 6th July 2016. One of the original Government Exemplar services, this had performed well during the run up to the 2015 general election and 2016 May elections, but ran into significant issues immediately prior to the registration deadline for the recent EU Referendum.

The issues in question arose on the evening of 7 June 2016 – when an unexpected surge of traffic caused a period of downtime lasting some 105 minutes.

This summarised version of our report has been prepared by us for you so that you can publish it, if you wish, without revealing confidential and sensitive technical information that could affect the security of the digital service.

Background

The Register to Vote service is an online service available via the www.gov.uk website. It offers citizens the ability to add their details to the UK electoral register online, as an alternative to more traditional options such as postal applications.

The service has been very successful in terms of uptake by citizens – now, 79.17% of registrations[1] are received via the online service and it successfully supported the general election in 2015 and the May elections in 2016 without incident. The Cabinet Office manages a detailed and rigorous programme to support the live usage and promotion of the service in addition to policy and legal support, including comprehensive support arrangements during peak usage.

Despite the thorough preparation and previous very good performance of the system[2], at 10.15pm on 7th June 2016 – just under a couple of hours before the midnight deadline for EU referendum registrations – the Register to Vote service became overwhelmed with traffic, enduring an unprecedented spike.

Scope of the Review

The review that follows will examine the factors which contributed to the outage on the 7th June 2016, from both a technical and process perspective. It is necessary to look at both angles: as we have learned more about the events leading to loss of service, it has become evident that while the issue was technical in nature, gaps in technical ownership and risk management contributed to the problem, and prevented it from being mitigated in advance.

Summary of Key Findings & Recommendations

Our general view is that the Cabinet Office supports the Register to Vote service very well. There is clear evidence of a detailed and thorough approach in supporting high demand for the service during key election periods despite the outage on the referendum deadline day, such as the 2015 general election, the 2016 May election and the EU referendum.

[1] From 11th September 2015 - 11th September 2016, as provided by the Cabinet Office.

[2] From initial go live - August 2014, average uptime was 96.33%.

Our key findings are as follows:

- We support the views provided by Kainos and FCO Services noted in their incident reports – that there was a bottleneck between the service's front-end and back-end in the application layer, causing the system to fail under sudden, unprecedented high levels of traffic;
- We support the view of FCO Services and Kainos that the hosting infrastructure was stable and functional throughout the incident period;
- The performance testing was limited, and the conclusions drawn from the results were not sufficiently detailed or tested;
- Risk management documented technical risk at a high level – but there was not enough actionable technical detail to fully identify and mitigate the issues we have identified in this report;
- There also appears to have been a gap in technical ownership, with roles and responsibilities unclear – consequently making it harder for technical issues to be identified and solved (or mitigated).

You have also asked us to provide our recommendations on how to improve the service in future. A full list will follow, but for ease of comprehension a high level list follows:

- **Improve system monitoring capability** – so that it is possible to easily gain a detailed view of all metrics relating to the system's architecture and performance, at any given moment in time.
- **Application changes and move to a cloud based infrastructure** – to allow for automated testing in a production-like environment. This will require application changes as the application at the time of the incident would not have benefited from a cloud based infrastructure. We understand this is part of the re-platforming project currently in progress.
- **Run performance tests frequently** – Make sure the service is routinely 'tested to destruction', is tested against real world user scenarios and is also tested continuously, as part of the continuous integration and deploy build. This will assist you in spotting and fixing issues before they ever become a problem in production.
- **Appoint a technical owner for the service** – Clear, well-understood technical ownership (in the hands of a suitably experienced, senior person or team), will ensure that the planning, coordination between technical suppliers, running, testing and ongoing development of the service is closely managed – especially in advance of and during key events.
- **Refine technical risk management activities** – be more specific regarding technical risk, for example, correlate expected spikes in usage to specific impacts on the system e.g. predicting likely bottlenecks within each of the different components of the architecture. Test any assumptions made by referring to historical monitoring data and performance data.

We hope the following review proves useful to you.

What went wrong?

Background

For context, the details of the incident are as follows, as described by Kainos in the MIR report:

> *"A televised referendum debate at 21:00-22:00 coincided with a very large spike in user traffic. Throughout the debate, [very high] traffic levels . . . were seen with no significant errors in the service. Immediately when the debate concluded, traffic roughly doubled . . . At 22:15 the largest spike was seen . . . It was at this point that the service became entirely unusable for all citizens and error messages were delivered to users trying to engage with the service. Successful applications reduced to zero as a result.*
>
> *The incident lasted 105 mins beginning at 22:15 and lasted until 00:00.*

The following further supports the details of the incident as recorded by FCO Services:

> *"The Register to Vote website performance degraded during the period of 22:45 to 23:58 on the 7th June 2016; primarily caused by high volumes of applicants requesting registration for the upcoming EU Referendum debate on 23rd June. The high levels of traffic were created due to a televised political debate and marketing campaigns. These generated application volumes exceeding the known safe limits of the website . . .*
>
> *. . . No other . . . [customer] was affected by performance issues during this period."*

Technical issues

Overloaded gateway and database

From our discussions, and from analysing the data and graphs at the time of the incident we can see that there was a problem within the service, which was under heavy load. Until further performance test results and associated monitoring is available to help identify the ultimate bottleneck, we cannot see anything to contradict the findings in the Kainos report, i.e. that "a sudden and very significant increase in user demand to complete registration… [caused] resource exhaustion . . ."

It is worth noting that a report from FCO Services highlighted that no DDOS attack was ongoing at the time of the incident.

Difficulty in obtaining a complete historical picture of the system's health

In compiling this review we experienced some difficulty in assembling an accurate, comprehensive 'snapshot' of the system and associated metrics at the time of the incident. A well-considered historical dashboard should be able to provide a complete picture of activity at any point in time, in moments, but this was not the case in this instance; monitoring was not in place for all components of the architecture, and web analytics were shared across GDS and the Register to Vote service.

Process issues

The high levels of system uptime and the very good wider support of the service during previous election periods suggests good process, and generally we support this view. In trying to understand how the outage on the 7th of June was not completely mitigated, we reviewed factors concerning process which we now believe were contributors – some taking place well before the incident on the 7th June.

Technical ownership

While the Cabinet Office is the owner for the service, its technical responsibilities were delegated to multiple suppliers. Although Kainos, FCO Services and NCC all held project responsibilities, there should still be an overall technical owner residing within the Cabinet Office to review and approve any technical risk and mitigations. This role demands the requisite technical experience and expertise to understand the system in its entirety.

Mistaken assumptions around traffic

The very good performance of the service in advance of and during the general election in 2015 and the May 2016 elections could well have created a sense of security that the system was able to perform beyond any expected peaks in traffic. We have found that the Cabinet Office did review in detail the historical usage and performance statistics for the service and did anticipate spikes in usage as election deadlines approached. The referendum is an example of one such deadline, but the level of the large spike seen during the referendum period was not anticipated.

It was assumed that peak traffic close to the deadline day for the referendum vote would be similar to that experienced in the run up to the previous year's general election; which is a reasonable assumption to make in the absence of other relevant data. It is however unclear how this assumption was tested or whether capacity planning and risk mitigations considered the scenario that demand could be higher than previously experienced.

In advance of any new deadline-driven events that may drive an increase of traffic to an existing service, it is always prudent to expect and to test for a sudden spike in traffic. This is different to testing for sustained load over a period of time or short bursts of traffic. The NCC load testing, commissioned in April 2016 prior to the referendum did provide a level of test assurance but could have provided more coverage and better explained the results (but we recognise that the scope of testing was limited with agreement from the Cabinet Office).

Limited testing

It is evident from the DDOS, capacity and load testing activities in preparation for initial launch, the general election and referendum that the Cabinet office did invest in a level of testing. This should continue and be further enhanced to include integrating a suite of performance tests into the continuous integration and deploy process. We understand that the existing environment is not ideal for such a setup in its current incarnation (as there is no

production-like environment in which to test and it is not cloud based) but the new re-platforming project will address this[3].
Load tests were performed by NCC in the weeks before the referendum and the scope of the testing was jointly agreed with the Cabinet Office in advance. The load testing did indeed report a problem with system performance when put under increased load but it was assumed that such load would not occur.

We have reviewed the test report provided and note the following:

- Not all user scenarios were tested.
- The performance tests did not continue to the point of destruction – which would have flagged up the system's breaking point in advance.
- The test results were not comprehensive; for example, they did not provide any idea of CPU load, disk load or memory load on the server. We understand that is because it was outside of the agreed scope of work for NCC.
- Key metrics such as memory allocation and usage logs were not included. Without this, the tests cannot provide any real information on where potential bottlenecks exist. We understand that is because it was outside of the agreed scope of work for NCC.
- A lack of time and a production-like test environment limited the scope of the testing performed and possible mitigations.

In summary, we believe that the scope of testing was too limited which would have made it difficult to conclude exactly how the system would perform under very high load, which in turn would make it difficult for the Cabinet Office to fully appreciate the associated risks. This also points to the need for greater technical expertise.

Virtualisation vs Cloud Infrastructure

While we recognise that FCO Services is able to increase infrastructure capacity quickly making the necessary application changes and moving to the cloud will allow for automated scaling. Having access to a pool of shared technology (as with a cloud solution) will allow additional resource to be directed quickly to where it's most needed.

We would like to highlight that when the Register to Vote website was designed, a cloud infrastructure was not an option as there were no cloud providers offering the required level of security. This constraint has now been removed. It's also important to restate that moving to a cloud based infrastructure in isolation would not have prevented the incident since the problem occurred within the application layer.

[3] As confirmed by Kainos via a conference call on 17th August 2016.

Our recommendations

Based on our findings above, we would make the following recommendations to avoid future issues of the kind experienced prior to the referendum.

Technical recommendations

Reduce risk through automated testing

We always recommend that any service is tested to absolute breaking point, in order to provide a detailed understanding of exactly how the system degrades over time, and to identify the exact bottlenecks encountered through the entire process, from front end through to the backend.

We recommend that once the proposed application changes and the move to the cloud has been completed as part of the re-platforming project, a performance environment is put in place. This would be a perfect clone of the production environment, where engineers can automatically test the performance of the system with a full range of measures: load tests, soak tests, peak tests, tests for destruction. This way the real constraints of the system will be known in advance of any potential performance improvements. This is especially important for any occasion that may attract a higher than usual level of traffic (like a referendum).

Improvements to performance testing

We recommend an increased focus on performance testing in the future (and especially for the re-platforming project). This type of performance testing is different to the automated performance testing, as noted above. Automated performance tests will detect changes in general performance with each release of new code, but for specific events or any significant changes to the code or architecture, in-depth performance testing should be performed.

It is also crucial to have an appropriate technical person or team to review the test reports for coverage and the test results. Someone with deep understanding of the system, of good modern architecture, and who can interpret detailed performance metrics. Without this, it will be difficult for non-technical people to understand the results and therefore ascertain risk.

Rewrite the service to use an asynchronous design

We understand it has already been suggested to rewrite the system to an asynchronous design – in particular, leveraging Amazon Web Services (AWS) to save voter registrations into the Amazon SQS (Simple Queue Service). We would highly recommend using an asynchronous pattern for the Register to Vote service (and any other service that might experience high, seasonal spikes in traffic).

Currently, the service design passes data back and forth through the gateway in real-time, creating an unnecessary bottleneck on that component whenever the service experiences a high volume of concurrent users. Given the processing of the data downstream to the local authorities is asynchronous, there is little need for this synchronous design in the first place.

An asynchronous architecture would scale far better than the current system – in theory allowing a very large number of concurrent users (certainly more than are entitled to vote in the UK).

Rewrite the Citizen API to use a different database implementation

The data saved during the Register to Vote application is largely non-relational. Arguably, a relational database is therefore unnecessary. We would recommend using any masterless noSQL database (open source options are available, such as Cassandra or CouchDB). This would perform well at writes and scale horizontally.

Incorporate new mitigation measures

Even where robust end-to-end performance testing has been implemented, we recommend that further mitigations be set in place. This is especially important for services that have a high profile (and potential political impact, if unavailable).

One such measure that should be put in place is some kind of visitor prioritisation system (or 'waiting room'). This will prevent users from connecting to the system if and when it is unable to cope with additional traffic – allowing them in only once there is capacity available. While it will provide a sub-optimal experience for any user encountering it, as a last resort it is still preferable to the entire service becoming unavailable. With proper testing and improved architecture in place, it should be a last resort that no user sees.

Implement a dashboard which retains historical data in detail

Without a clear window on a system's vital signs – one which can provide a snapshot of the system at a given point in time – it is harder than it should be to identify issues and potential causes. Once sufficiently robust testing is up and running, we recommend an analytics dashboard is put in place to allow the technical owner and other key stakeholders a single point of view on key metrics. The data should be retained at a detailed level for reporting purposes.

Process recommendations

Deepen technical expertise within the organisation

We recommend highly experienced, technically-minded people always review the current state of the technology in detail, well in advance of any key event. They should also take ownership of any necessary changes, and enjoy a level of seniority sufficient to push those changes through to completion. They may be within the Cabinet Office, or GDS; just as long as the review happens. We recognise that this role did exist during the original implementation supplied by GDS; although GDS played a very beneficial but perhaps informal role in supporting IER during the referendum, the role was lost while transitioning the live service to the Cabinet Office.

Information management and risk management

As already noted, there was a great deal of time and effort dedicated to setting up and managing risk management processes and governance, we believe the following may assist with shaping this process, for example:

- Include assumptions in the risk register and use data to validate them.
- Breakdown risks into single actions or points to allow for more targeted mitigation plans. Use a framework for assessing technical risk e.g. SWOT Analysis.
- Include performance testing in the risk register.
- Performance testing should have a technical owner.
- Be clear on the different performance metrics used.
- Do not rely on email communication for complicated issues.
- Allow time to understand any new events, investing in user research if relevant to try to predict user behaviour.
- Allow time to build in mitigations.
- Include technical and (systems) delivery representation within the programme boards. Experienced individuals with a history of delivering highly performant systems.

In conclusion

It is fair to recognise that the Register to Vote service is a successful service for many reasons as previously stated, and that the events of 7th June should be looked at as a valuable learning experience. We believe it would be very dangerous, however, to assume that this level of demand (or higher) for the service will never happen again. Now is the time, then, to prepare for such a future occasion – and ensure the technology is able to cope with large spikes in traffic. We now understand that this is part of the new re-platforming project.

It is also important to note that although the service is in live operation, the same due diligence and cross functional skills employed during the original implementation are required now, to ensure the service continues to improve and meets new and emerging user needs[4].

[4] This needs of a live service are also detailed in the GDS Service Manual - www.gov.uk/service-manual