Enhancing Experiment Central Service Reliability: from delivery to security and virtualization

F Donno[1], **S Baranov**[2], **S Buzykaev**[3] **and M D Saiz Santos**[1]
[1] CERN/IT, Switzerland
[2] Joint Institutes for Nuclear Research (JINR), Russia
[3] Budker Institute of Nuclear Physics (BINP), Russia

E-mail: Flavia.Donno@cern.ch

Abstract. The four LHC experiments rely on experiment specific services running on machines mainly located at CERN. Some of these services have been rated by the experiments as very critical: any loss or degradation of performance has a major impact on the experiment's production and analysis activities. It is therefore important to provide a reliable and robust operational environment. In this work we describe the strategy based on service deployment, security and virtualization adopted to enhance the reliability of ATLAS and CMS central services.

CERN-OPEN-2011-001
09/02/2011

1. Introduction

The four LHC experiments rely on experiment specific services running on machines largely located at CERN. Some of these services have been rated by the experiments as very critical: any loss or degradation of performance has a major impact on the corresponding production and analysis activities. It is therefore important to provide a reliable and robust operational environment. In this work we describe the strategy based on service deployment, security and virtualization adopted to enhance the reliability of ATLAS and CMS central services.

In order to allow for fast recovery in case of a disaster, we propose a set of policies for service delivery, deployment and configuration. Fabric management framework and tools built around these policies help streamlining the management of experiment services. The deployment of such services to other sites is therefore made simpler and possible.

Most of the experiment central services rely on web frameworks and a number of loosely integrated technologies which could be source of potential security threats. While it is necessary to harden secure web programming, education on security matters and good development practices might be difficult or time consuming for the average user. We propose the adoption of web application firewalls, reverse proxies, authorization mechanisms and web analytics for the containment of the most common vulnerabilities while providing auditing and virtual patching.

In this work we further describe how ATLAS and CMS have profited from the CERN virtualization project and how specific reliability requirements have been addressed by the existing infrastructure.

This paper is organized as follows. In section 2 we describe the policies and common practices that are now an integral part of the daily operations for some of the experiments. In particular, we report on the ATLAS case. In section 3 we introduce the web redirector service that is being actively used by three of the four LHC experiments. Section 4 is dedicated to AutoVOS, the autonomous system developed

on top of the LHC Era MONitoring (LEMON) [1] framework at CERN. Finally, in section 5 we describe how virtualization is being used to optimize hardware resources and better manage the experiment services. We conclude in section 6 with details about our future plans and related work in the area.

2. Policies and common practices

In order to streamline and automate operations and make the recovery of a service more effective, a set of policies have been established and made operational for the ATLAS experiment [2]. In particular the roles for the VO contacts, the application manager, the service manager and the security contact have been defined. The VO Contact is responsible for the management and operations of machines where experiment specific services run. The Application Manager is responsible for the development of a specific service while the Service Manager is responsible for running the service. The Security contact is responsible to establish security policies and manage a security incident.
Among the policies introduced we would like to mention the following that we consider most relevant:

1. We established the existence of a catalogue where all experiment services are described together with relevant information for management and operation.
2. Experiment services should be delivered in the form of an rpm and include pre and post configuration scripts and a list of sensitive configuration files.
3. A common repository is available to store source and binary rpms for all experiment services and dependency packages. Furthermore, the source code is also stored in a central SVN repository. For sensitive files, the SINDES [3] framework is used and a set of tools have been created to automate the storage of sensitive configuration files from the service managers and to extend some of the current functionality to experiment service managers.
4. The same set of tools used to manage CERN services and machine in the computing center are also used to manage experiment services [4].
5. A set of rules and frameworks have been established for application managers to develop experiment specific services in a secure manner. Procedures to manage security incident have been details and the service managers are contacted and involved in the process.
6. Virtual machines are used to host experiment services whenever possible.
7. Good practices and guidelines to be used to assign DNS aliases to services, for load balancing, for the management of development machines, for passwords, database management and web services have been described in experiment specific twiki pages.
8. Machines hosting experiment specific services are monitored through LEMON. Through AutoVOS, described later, either service managers or the system manger on duty from the CERN computing center are called for actions in case AutoVOS is unable to restore the functionality of an unavailable or degraded service.

3. The web redirector

Most of the experiment central services rely on web frameworks and a number of loosely integrated technologies which could be a source of potential security threats. While it is necessary to harden security practices for web programming, education on security matters and good development practices might be difficult or time consuming for the average user.
A Web Application Firewall (WAF) is an appliance or software that provides customized protection for web applications against attacks. A Reverse Proxy is a server that routes all connections coming from the Internet addressed to one Web server. Reverse proxies can deal with the request itself or pass the request wholly or partially to the main web servers.
The Web Redirector is a reverse proxy developed by CERN IT Experiment Support Group for the WLCG experiments. It is used in front of other Web services to act as a web application firewall. The Web Redirector attempts the mitigation of potential threats coming from the underlying network, managed and unmanaged clients and hosts, potential untrustworthy users.

As shown in figure 1, all connections coming from the Internet addressed to one of the experiment Web servers are routed through the experiment specific Web Redirector service. The Web Redirector filters the requests before redirecting them to the real web server serving the request, as shown in figure 2. The filters applied can be of many kinds: authorization filters, cross-scripting attack checking, etc.
Besides acting as a WAF, the Web Redirector offers as well the possibility of providing customizable load balancing algorithms for the web applications running behind it. It can also act as a web cache.

3.1. The Web Redirector Technologies

The web redirector offers the following features through well established technologies:

- WAF through the Apache ModSecurity [5] module.
- SSL based authentication as shown in figure 2.
- Single Sign On (SSO) through the CERN Shibboleth [6].
- Load distribution: requests can be served by several experiment specific web servers, each serving the same or its own application. Load distribution is achieved through the Apache mod_proxy_balancer [7].
- Caching support. The reverse proxy can offload the web servers behind it by caching static content through the Apache mod_cache and the frontier-squid server [8].
- Support for special configurations: AJP protocol for Tomcat-based applications, customized redirection through rewrite rules; session-aware forwarding; kerberos-aware sessions.
- Hardware sparing by supporting virtualization.
- Web analytics through AWStats [9] and webalizer [10].

The required operating system under which the Web redirector runs is SLC5.

Figure 1. The web redirector architecture.

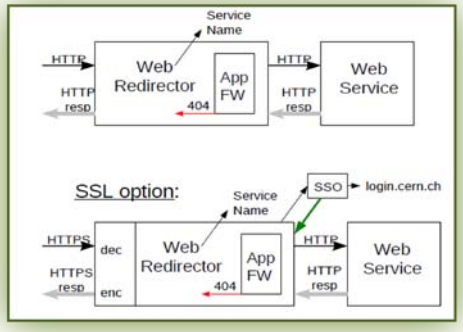

Figure 2. The web Redirector HTTP and SSL workflow.

4. Autonomous systems

Effective monitor of services and prompt reaction in case of service degradation of interruption are important factors to guarantee service reliability. Using the LHC Era MONitoring (LEMON) [1] framework developed at CERN we provided the AutoVOS autonomous system for experiment services. This system is service agnostic. It performs periodic checks of the service status. Self-healing actions can be defined depending on the status of the service. Furthermore, e-mail or tickets in a tracking system can be generated to inform a predefined set of expert about the status of the service. Specific actions can also be applied depending on the status of the underlying host. AutoVOS has

helped improve our ability to guarantee the required SLAs while minimizing the workload required running the services. Figure 3 shows the result of AutoVOS in action.

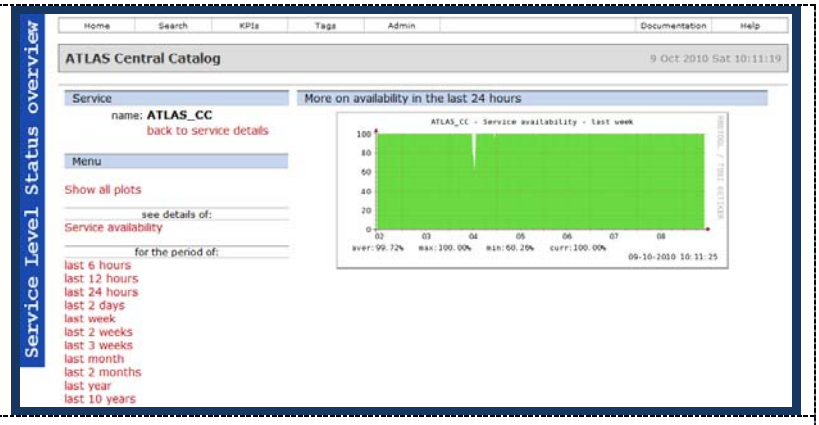

Figure 3. AutoVOS in action. The functionality of the degraded service has been recovered without human intervention.

5. Virtualization

Specific reliability requirements have been provided by the LHC experiments to the CERN virtualization project. Today the virtualization service running at CERN fully satisfies these needs. Currently ATLAS and CMS serve half of their critical services through Virtual Machines running both at CERN and at a service host in Geneva for reliability purposes. Virtualization allows service managers to provide an insulated environment for their services even if running on the same physical hardware, avoiding conflicts in dependent software. Furthermore, virtual running images can be migrated to different physical hardware in case of problems or for maintenance reasons. In case of load issues it is very easy to replicate an existing service on a virtual machine running on different hardware and at different locations for reliability and stability issues.

Virtualization is also heavily used for ATLAS and CMS services behind the Web Redirector. Virtualization has allowed for a very flexible management of experiment services.

6. Conclusions

Through our effort of extending common practices used for general CERN services to the experiment specific services we have experienced a better quality of service, quicker recovery times, better monitoring and reporting. Through automation the man power needed to run the experiment services can be reduced considerably and sometimes even reduced by half. To achieve such results a modest effort had to be invested in order to extend the functionality of existing management and operation tools to the experiment services, to document procedures and tools in a coherent and experiment-oriented way and to disseminate common practices through experiment managers. This work is still not finished. With the help of the ELFms [4] developers, we intend to make AutoVOS more effective, allowing service managers to describe service degradation and actions in a more flexible and descriptive way. We are working also to introduce better authorization schemas and namespace management in SINDES so to allow experiment service manager to directly manage sensitive information for their service without involving VO contacts, as it is done today. Furthermore, we intend to introduce more functionality in the web redirector in order to automatically analyze the reports produced by the Apache ModSecurity and allow for active filtering of requests, instead of operating such a module in log only mode.

Our experience is taken as an example by other collaborating institutions part of the WLCG project. Indeed, it is advantageous to be able to run copy of experiment highly critical services at remote sites

in order to grant full availability. From an infrastructure point of view, as of today, ATLAS and CMS services running at CERN can be easily exported and run at remote institutions part of the collaboration.

Acknowledgments
The authors wish to acknowledge the CERN IT Computing Facilities, Platform and Engineering Services and Experiment Support Groups for their help and support without which this work would have not been possible.

References

[1] Siket M 2008 *Lemon/LAS for System Administrators* (CERN)
[2] Donno F 2010 *ATLAS Procedures* https://twiki.cern.ch/twiki/bin/viewauth/Atlas/CentralServicesManagementPoliciesAndProcedures
[3] Lopienski S 2008 *SINDES: Secure INformation DElivery System* https://twiki.cern.ch/twiki/bin/view/FIOgroup/SinDes
[4] McCance G 2008 *ELFms and other related tools* https://twiki.cern.ch/twiki/bin/view/FIOgroup/ServiceManagersStartHere
[5] Apache *ModSecurity* 2009 http://www.modsecurity.org/documentation/
[6] CERN *The Shibboleth Service* 2010, https://espace.cern.ch/authentication/CERN%20Authentication%20Help/Shibboleth.aspx
[7] Apache 2011 *Apache Server Version 2.0* http://httpd.apache.org/docs/2.2/mod/mod_proxy_balancer.html
[8] Donno F 2010 *Frontier-squid* https://twiki.cern.ch/twiki/bin/view/PDBService/SquidRPMsTier1andTier2
[9] AWStats 2011 *AWStats official web site* http://awstats.sourceforge.net/
[10] Barrett B 2009 *The Webalizer* http://www.webalizer.org/