Journal of Physics: Conference Series **331** (2011) 062005 doi:10.1088/1742-6596/331/6/062005

Improving Security in the ATLAS PanDA System

J. Caballero[1], T. Maeno[1], P. Nilsson[2], G. Stewart[3], M. Potekhin[1], T. Wenaus[1]

[1] Brookhaven National Laboratory, PO BOX 5000 Upton, NY 11973, USA
[2] CERN, CH-1211 Geneva 23, Switzerland
[3] Department of Physics and Astronomy, University of Glasgow, Glasgow G12 8QQ, UK

E-mail: jcaballero@bnl.gov

Abstract. The security challenges faced by users of the grid are considerably different to those faced in previous environments. The adoption of pilot jobs systems by LHC experiments has mitigated many of the problems associated with the inhomogeneities found on the grid and has greatly improved job reliability; however, pilot jobs systems themselves must then address many security issues, including the execution of multiple users' code under a common 'grid' identity. In this paper we describe the improvements and evolution of the security model in the ATLAS PanDA (Production and Distributed Analysis) system. We describe the security in the PanDA server which is in place to ensure that only authorized members of the VO are allowed to submit work into the system and that jobs are properly audited and monitored. We discuss the security in place between the pilot code itself and the PanDA server, ensuring that only properly authenticated workload is delivered to the pilot for execution. When the code to be executed is from a 'normal' ATLAS user, as opposed to the production system or other privileged actor, then the pilot may use an EGEE developed identity switching tool called gLExec. This changes the grid proxy available to the job and also switches the UNIX user identity to protect the privileges of the pilot code proxy. We describe the problems in using this system and how they are overcome. Finally, we discuss security drills which have been run using PanDA and show how these improved our operational security procedures.

1. Introduction

During the last year the LHC (Large Hadron Collider)[1], the particle accelerator/collider built at CERN (European Organization for Nuclear Research) started to operate. Two beams of protons have been circulating in opposite directions inside the accelerator at a center of mass energy of 7 TeV. The experiments have started recording data and the first physics results have been published. ATLAS (A Toroidal LHC ApparatuS)[2] is one of these experiments.

The amount of data produced by the ATLAS detector makes storage and processing on site at CERN impractical. Consequently, ATLAS has adopted a multi-tiered data storage and analysis model based on distributed, Grid Computing[3]. The data produced is divided up and transferred via the Internet to several Tier-1 sites around the world, where it is stored permanently. After that, Tier-2 and Tier-3 sites process and analyze this data on large computing clusters (typically rack-mounted commodity Linux systems).

All these computing sites are affiliated with grids. EGEE (Enabling Grids for E-sciencE)[4] and NDGF (Nordic DataGrid Facility)[5] in Europe, and OSG (Open Science Grid)[6] in the U.S., provide the software and administrative infrastructure for making this highly distributed

computing architecture possible. Activities are coordinated by the WLCG (Worldwide LHC Computing Grid) organization.

The ATLAS processing and analysis tasks are defined as distinct jobs which, via the grid interfaces, are submitted to dozen of sites where the data to be analyzed already exists, to prevent a high data transfer time consumption. To meet ATLAS requirements for this data-driven workload management system capable of operating at the actual LHC data production rate, the PanDA (Production and Distributed Analysis) system has been developed built upon the pilot-based framework, as described in [7]. It includes an important subsystem (the pilot scheduler) that manages the delivery of pilot jobs to worker nodes via a number of scheduling systems. Once launched on a worker node (WN), the pilot process contacts the job dispatcher and receives an available job matched to the site resources and characteristics, combined with brokerage policies.

The advantages of a pilot job architecture are clear: it makes the working environment more homogeneous and isolates the job execution from the potential heterogeneities. However, there is an inherent security risk in the fact that the jobs inherit the identity of the user who originally submitted the pilot jobs. Accordingly, all end-user jobs will then possess the same identity and the same privileges granted by the proxy [8] carried by the pilots.

To address this security risk, the PanDA system has been given the capability to change back the user identity of the payload job (initially run as pilot) to the original job submitter's. The change takes place based on the end-user grid credentials, previously stored on a MyProxy [11] caching service, from which they are retrieved via delegation steps by the pilot process running on the WN. The mechanics of the identity switch are performed by gLExec. gLExec[9] is a super-user privileged executable with the capability of modifying the UID and GID to provide for a mapping between the grid user and the local Unix user accounts. This mapping is performed based on the results from gLite LCAS and LCMAPS[10] security components. It is an open source application for managing grid proxy credentials. The identity change, as an optional feature of PanDA, is employed only at sites which mandate its use. Such a condition is reflected in the site metadata recorded in the PanDA server's database.

2. Identity switch

When the user submits a job to the PanDA system, the client software (pathena)[12] will check if the user already has a proxy stored on a dedicated MyProxy server, owned and managed by the ATLAS organization, and that this instance has a sufficiently long remaining period of validity (which is configurable). If that is not the case, the client will store a new user proxy onto the MyProxy server. The user's credentials are never cached in PanDA.

During this delegation to the MyProxy server, the identity of the entities authorized to retrieve is specified. These identities are declared as a list of Distinguished Names (DN) corresponding to the pilot job submitters. In this way it is assured no other users than the authorized pilots will be allowed to retrieve the users' proxies from the server. This list of authorized retrievers is stored in the PanDA database, and read every time a new proxy is delegated.

In this scenario, the pilot job (and the pilot job only), once it has connected to the PanDA server and is ready to execute the end user job, has the credentials to extract the concrete user's proxy from the MyProxy server. Finally, to be able to perform the UID and GID switch, the pilot identity must to be included in a super-user owned white list of users allowed to invoke gLExec.

To prevent possible security concerns related to misuse of a compromised pilot proxy, an instance of the proxy being stored on the MyProxy server is assigned a unique key that will be required upon future retrieval. This key is a generated random string, stored on the PanDA server, and delivered to the pilot when the latter obtains a new job from the server.

In addition, single (or few) use tokens will be used by the pilot job in order to get a payload job from the server. The pilot will have to present the token to the PanDA server in the job request process, and server has to match it to the value obtained from the database, where it was stored during the pilot submission process. After validating the pilot, the server immediately deletes the token (or decrements a maximum usage count).

2.1. Execution

A detailed explanation of the execution process can be found in [15]. The VOMS system [13] is used to define and secure the privileges and access rights of the pilot jobs, which are started by a privileged user (who has the VOMS pilot role annotation). This user is mapped to a special account allowed to utilize gLExec.

The gLExec utility is activated and uses the user's proxy obtained from the MyProxy server. The identity switch via gLExec leads to each user's processes running under specific UIDs traceable to their respective identities. At the PanDA level, the DN of the job submitter is recorded permanently for each PanDA job, such that PanDA can trace and account usage.

When the retrieved credentials do not carry VOMS attributes, or they have expired, they need to be (re)added on the worker node by invoking the VOMS client. It is preferable that the delegated credentials already include the desired VOMS attributes (even if expired) to avoid an improper escalation of VOMS privileges in case a credential is compromised.

Finally, once the payload has been executed and the running process returns back to the pilot, the retrieved credentials are deleted from the local disk.

2.2. Issues related to the identity switch

As it was explained in detail in [15] several issues surrounding the identity switch are addressed by the PanDA pilot just before gLExec invocation, or during the gLExec execution just before invoking the final payload:

(i) The running process is moved back to the original working directory after the identity switch moves it to the new identity home directory.

(ii) Permissions to the working directories are modified to grant the new identity writing privileges.

(iii) Reconstruction of the complete set of environment variables after the identity switch deletes it.

As an alternative to the directory permission modification a different approach is under consideration. A new working area is created underneath the temporary directory by gLExec, belonging to the new identity, which grants writing and execution privileges. Then all original files can be copied to this new working space and the payload execution resumes. New generated files are copied after gLExec invocation to the original working space and the temporary directories are removed. This mechanism prevents the need of modifying UNIX directories and files attributes.

3. Security Service Challenge 4

Security service challenges are organised in Europe by the European Grids for E-Science Security Team (EGEE is now superseded by the European Grid Infrastructure project, EGI). These challenges are designed to test sites' security responses to a simulated, but realistic, security alert incident. As ATLAS is a major user of grid infrastructure we were approached by the EGEE security team and asked if we would help organise Security Service Challenge 4 (SSC4) via the PanDA system.

This was an opportunity to also test ATLAS's response to a grid security incident: to review workflows, to improve documentation and to provide training for ATLAS grid experts in security handling.

3.1. Security Contact and Procedures

As one of the first actions, before SSC4 jobs ran, we reviewed the security contact information provided by ATLAS to sites (although the playbook for security incidents stipulates that sites should contact their regional and grid security teams, direct contact between sites and ATLAS was anticipated). We refreshed the membership of the group, so that all key computing experts were involved, as well as our CERN sysadmin team and CERN security experts.

We also improved our security handling procedures. ATLAS Distributed Computing (ADC) has an ADC Manager On Duty (AMOD) who manages ATLAS offline computing operations in one week shift blocks. AMOD was given primary responsibility to triage security incidents and to coordinate any ATLAS response. AMOD documentation was improved, particularly in the area of tracing the payload submitter of any PanDA job. One important point was to use the security list for coordination between experts after the initial incident report, thus aiding communication between ATLAS and grid security experts.

Previously in PanDA banning users had been achieved by database manipulation. This was not a good or scalable solution, so a more robust system was put in place. This allowed the AMOD to run one script on the PanDA server which would ban a DN, preventing further job submission, and another script to cancel all jobs associated with a DN in PanDA. Naturally, an unbanning script was also provided.

With these changes in place, ATLAS was well prepared to help run SSC4.

3.2. Test Setup

To practically run SSC4 through PanDA we had to ensure that the exercise would not impact upon ongoing experiment activities. To achieve this we:

(i) Obtained a different grid credential which would be used only to run SSC4 challenge jobs.
(ii) Setup a special pilot factory, using this credential, and submitted pilots to the sites being challenged.
(iii) These pilots were modified to only allow the SSC4 challenge jobs to be run.

In parallel, the SSC4 team obtained a different grid credential for a 'rogue' user, who would submit the SSC4 payload into the PanDA system.

Sites would then be able to take banning or blocking actions against both of these credentials, which would not affect ATLAS production and analysis at the target site.

3.3. Running the Challenge

As the security team submitted the challenge payload into PanDA, and challenge jobs started to run on the sites, the sites were contacted with a report of suspicious behaviour characterised by network connections to certain hosts outside the site.

Sites were able to trace the job to a running process started by the pilot which had been submitted with the special SSC4 credential. At this point many sites asked for help from ATLAS to trace the user involved and understand the payload which had been submitted. With the documentation provided the AMOD was able to immediately provide the sites with information about:

(i) The DN of the payload submitter.
(ii) The hostname from which the payload was submitted.

(iii) The timestamp of the submission.
(iv) A link to the payload code itself.

In practice we discovered that many of the sites were able to use the PanDA monitor to obtain some (or all) of the above information.

After the site contacted us, the AMOD then enacted a simulated investigation of the incident. The site was informed that we were attempting to contact the user to investigate (at this point the site was expected, in the challenge, to ban both the pilot submitter's and the suspicious user's DN). After a few hours we contacted the site again and told them that the user's activities were not compatible with the ATLAS VO's aims, that they should be considered banned from ATLAS and that their was no evidence that their proxy had been compromised. The site was then allowed to unban the pilot DN, but should have left the user DN banned.

3.4. Lessons

SSC4 was a very useful exercise for ATLAS because it allowed us to:

(i) Review and improve our security contact information.
(ii) Provide a clearer workflow for handling security incidents.
(iii) Improve the technical tools needed to handle the banning and unbanning of users in PanDA.
(iv) Exercise the security procedures in practice, providing training to ATLAS computing experts.
(v) Strengthen our relationship with grid security teams.

4. Conclusions

The PanDA system has demonstrated the reliability and efficiency of grid operations based on a pilot model. Since ATLAS began analyzing data from the collisions at the LHC accelerator, the PanDA framework has robustly sustained the needed rate of analysis jobs. The security issues related with the pilot job model have been studied and solutions have been implemented. In addition, the ATLAS collaboration is participating in a series of security service challenges, organized by EGI, aimed at testing the response to a security incident. These tests have helped ATLAS to improve documentation, review workflows and train experts in security handling.

Acknowledgments

The authors would like to thank the developer of MyProxy Jim Basney (NCSA/OSG), and the developers of gLExec Oscar Koeroo, Gerben Venekamp and David Groep (NIKHEF) for their constant support. We are also grateful to the site administrators and experts who have helped with technical questions and made possible the development of this work: John Hover, Xin Zhao, and Maarten Litmaath. This work was supported by the US Department of Energy and National Science Foundation, and managed by the Open Science Grid Consortium.

References

[1] LHC Computing Grid Project *http://www.cern.ch/lcg/*
[2] ATLAS Collaboration 1994 ATLAS Technical Proposal *CERN/LHCC/94-43*
[3] Foster I, Kesselman C and Tuecke S 2001 The Anatomy of the Grid: Enabling Scalable Virtual Organizations *International J. Supercomputer Applications* **15**(3)
[4] Enabling Grids for E-sciencE *http://www.eu-egee.org/*
[5] Nordic DataGrid Facility *http://www.ndgf.org/ndgfweb/home.html*
[6] Open Science Grid *http://www.opensciencegrid.org/*
[7] Nilsson P, Caballero J, De K, Maeno T, Potekhin M and Wenaus T 2008 The PanDA system in the ATLAS experiment *Accepted for publication in PoS, Proceedings of ACAT 2008 Conference.*

[8] Tuecke S, Welch V, Engert D, Pearlman L and Thompson M Internet X.509 Public Key Infrastructure (PKI) Proxy Certificate Profile *RFC* 3820
[9] Groep D, Koeroo O and Venekamp G 2008 gLExec: gluing grid computing to the Unix world *J. Phys.: Conf. Series* **119** 062032
[10] Groep D, Koeroo O and Venekamp G Grid Site Access Control and Credential Mapping to the Unix domain *Nikhef PDP Technical Report http://www.nikhef.nl/grid/lcaslcmaps/*
[11] Basney J, Humphrey M and Welch V 2005 The MyProxy Online Credential Repository *Software: Practice and Experience* **35, Issue 9** 801-16.
[12] pAthena *https://twiki.cern.ch/twiki/bin/view/Atlas/PandaAthena*
[13] Virtual Organizations Membership Service (VOMS) *http://www.globus.org/grid_software/security/voms.php*
[14] Hover J, Packard J et al 2004 Grid User Management System (GUMS) *https://www.racf.bnl.gov/Facility/GUMS/1.3/index.html*
[15] Caballero J, Hover J, Litmaath M, Maeno T, Nilsson P, Potekhin M, Wenaus M and Zhao X gLExec and MyProxy integration in the ATLAS/OSG PanDA workload management system *J. Phys.: Conf. Series***219** 072028