University of Warsaw
Interdisciplinary Centre for Mathematical
and Computational Modelling

Jakub Kopeć
Student's book no. 417354

Exploration of cooperation-enabling solutions in HPC

Second cycle degree thesis
field of study COMPUTATIONAL ENGINEERING

The thesis written under the supervision of:
Marek Michalewicz, Ph.D.
Interdisciplinary Centre for Mathematical
and Computational Modelling

Warsaw, March 2020

Oświadczenie kierującego pracą

Oświadczam, że niniejsza praca została przygotowana pod moim kierunkiem i stwierdzam, że spełniła ona warunki do przedstawienia jej w postępowaniu o nadanie tytułu zawodowego.

Data Podpis kierującego pracą

Statement of the Supervisor on Submission of the Thesis

I hereby certify that the thesis submitted has been prepared under my supervision and I declare that it satisfies the requirements of submission in the proceedings for the award of a degree.

Date Supervisor's signature

Oświadczenie autora pracy

Świadom odpowiedzialności prawnej oświadczam, że niniejsza praca dyplomowa została napisana przeze mnie samodzielnie i nie zawiera treści uzyskanych w sposób niezgodny z obowiązującymi przepisami.
Oświadczam również, że przedstawiona praca nie była wcześniej przedmiotem procedur związanych z uzyskaniem tytułu zawodowego w wyższej uczelni.
Oświadczam ponadto, że niniejsza wersja pracy jest identyczna z załączoną wersją elektroniczną.

Data Podpis autora pracy

Statement of the Author's on Submission of the Thesis

Aware of legal liability I certify that the thesis submitted has been prepared by myself and does not include information gathered contrary to the law.
I also declare that the thesis submitted has not been the subject of proceedings resulting in the award of a university degree.
Furthermore I certify that the submitted version of the thesis is identical with its attached electronic version.

Date Author's signature

SAGE2

SAGE stands for Scalable Amplified Group Environment and it is Node.js-based (JavaScript) software that facilitates use of large video-walls that are intended to be used by multiple users at a time. It works as the server's software - all the resource-demanding operations are handled by the server, so the end-user do not need to posses powerful workstation in order to use a video-wall. The special script run on the machine that operate the displays creates a server that provide two services. The first one is Google Chrome-based interface that displays the working environment on the video-wall. The second one is web-accessible portal that allows authenticated users (each SAGE2 session could be password-protected) to control content displayed on the video-wall. When user connects to the server he is presented a simplified schema of the video-wall that shows how the display space is arranged and an intuitive interface that allows to control the contents of the display. [2]

SAGE2 provides user with the modules that may be displayed on the screen. The essential ones, like web browser, google maps, notepad etc. are already implemented by the creators of the SAGE2, but there is special appstore where developers publish their own modules designated to support another software. If one would like to create his own module the developer's guide for making such module is available at SAGE2 project homepage.

During the installation of the SAGE2 in Technology Center of Interdisciplinary Centre for Mathematical and Computational Modelling (ICM) author encountered few problems concerning the network configuration as SAGE2 server required public IP address and DMZ network what could create some complications if the network design had not been adapted to such use. The ICM technicians bypassed these issues with ports forwarding, but they also noticed a security weakness induced by such solution - open unsecured port pose threat of unauthorized access to the network. In order to eliminate such possibility the access to the port used by SAGE2 was secured with username/password authentication.

After the installation author prepared a presentation for ICM's staff about the functionality and instructions on how to use SAGE2 software. The presentation was followed by a discussion on possible appliances of SAGE2 as a tool for cooperation between research facilities. The most repeated remark was that this software allows only cooperation between two SAGE2 sites and that there is practically no support for users that are not present in front of the video-wall. Another issue that was pointed out by the audience was the fact that API for making own SAGE2 module is rather fixed on JavaScript and it would not be easy to create such module for an application that was not designed in advance to support such functionality. The last but not least was the matter of security of SAGE2. The audience noticed that there is no clear declaration about the decryption used by SAGE2 and that SAGE2 protocols could not be sufficiently secure to be used in projects that require confidentiality.

To sum up, the SAGE2 could act as platform for conducting collaborative research, but in present-day form it may be used locally as middleware for large resolution screens rather than for remote collaborative work. Even though the creators of SAGE2 successfully conducted remote collaborative work session [2], in the author's opinion preparing such sessions require significant effort to establish reliable connection between two SAGE2 sites that would be justified only in case of long cooperation between two institutions where multiple SAGE2 sessions would bring noticeable boost in cooperation. Moreover the issues mentioned in the previous paragraph should be addressed beforehand.s On the other side - SAGE2 is perfect tool to facilitate the collaborative effort in case when all users are physically present in front of the video-wall. Perfect example of such use is StickySchedule app that was intended to be launched on SAGE2 site and it's purpose is to ease and precipitate the conferences scheduling [3].

DTP

The second idea was to create from scratch a web portal that would mask IT's expertise-demanding part of big data moving aspects from the end user and simplify such task as much as possible. The draft name for the project was "Data Transfer Portal" (abr. DTP). The main motivation behind the project was fact that software that is used in HPC applications to move large amount of data is rather unfriendly and unintuitive for the user that is not IT-technician responsible for data transfer. Not only is the use of such software complicated, but it is also necessary to test the connection properties between source and destination in order to optimise the transfer. The DTP is intended to handle all this operations and provide the end-user with simple web interface that is easy to use and do not require IT expertise. On the beginning author committed some time to learn how to use django framework with python [1] as it seemed that project would require creating a web portal at some point. Nevertheless, when author started to think on how DTP should look like he encountered a problem trying to answer the question "How DTP server will know that the user that require data transfer is really who he claim that he is and if he is allowed to transfer that data (permissions control)?". At that point author completely focused on research on user authentication and authorization methods. The main issue was the fact that assumedly users would not be the members of one organization and each user should be a member of at least two different parties (one source and one destination).

Authentication methods:[4]

- basic authentication - user and password (may be encrypted)
- SAML - Security Assertion Markup Language[5]
- Oauth2.0[6] with OpenID[7]

SAML - Security Assertion Markup Language SAML is a XML-based standard created and maintained by OASIS (Organization for the Advancement of Structured Information Standards). It's main purpose is to describe how the security information could be exchanged on-line between two separate parties. It is based on the exchange of standardised messages , called SAML assertions, that are created according to the standard's syntax and rules. The framework's assumption is to provide components that could be used in many configuration to meet the user's requirements. Moreover, the SAML specification includes profiles that are predefined to satisfy the most common use-cases.[5]

OpenID Connect OpenID Connect is the authentication standard used on top of the OAuth2.0 authorization protocol. In the previous versions OpenID and OAuth were separate standards. OpenId's purpose was to verify the identity of the user based on the authentication that is performed by OpenID provider[8]. OAuth 2.0 protocol was responsible for verification of the user permissions to the requested assets[6] while OpenID just ensures the service provider that the user is in control of some identifier (e.g. the gmail account) and there was no way of determining if the user name or any other data are valid and real. It is possible to create OpenID provider on one's own "(http://wiki.openid.net/w/page/12995226/Run your own identity server)" and use it to issue conduct completely valid authentication to the service provider using OpenID. In the newest version of the standard - OpenID Connect - these two protocols were connected and now OpenID provides not only the user authentication but it also enables the user authorization.[7]

Difference between SAML and OpenID Before the implementation of the OpenID Connect there was a significant difference between SAML and OpenID. First of all the OpenID2.0 is the authentication protocol while SAML provides the authorization and the authentication as well. Secondly, SAML authorization was based on the thrust relationship and on the beforehand arrangements between partners. The service provider trusted that the identity provider is or was able to authenticate the user real identity. The example of such scenario are e-identity services available on government's and local authorities' sites. When a citizen is to fulfill administrative matters online he may use his bank account to log in to authorities' portal. In this case the user's bank is the identity provider and the government's site is a service provider that trusts that bank's employee verified the user's identity (for example - checked his ID card or passport) when he opened the bank account. SAML standardised the messages used in authentication (and, on the next stages, in authorization) process. As for OpenID such authentication was not possible as identity provider could not guarantee truthfulness of the user's data). After the introduction of the OpenID Connect where authorization was joined with authentication the difference in functions between those two standards blurred. Nevertheless the implementation of OpenID Connect is simpler than the implementation of SAML protocol so OpenID is used in simpler web applications while the SAML is used in large federations (e.g. university or enterprise federations) as it has already been widely adopted in existing federations and is more mature standard than OpenID. On the other hand SAML is restricted to browser use, so in the case of application or device usage OpenID is the obvious choice.[9]

Bibliography

[1] https://docs.djangoproject.com/en/3.1/intro/

[2] T. Marrinan, J. Aurisano, A. Nishimoto, K. Bharadwaj, V. Mateevitsi, L. Renambot, L. Long, A. Johnson, and J. Leigh, "SAGE2: A New Approach for Data Intensive Collaboration Using Scalable Resolution Shared Displays" (best paper award), 10th IEEE International Conference on Collaborative Computing: Networking, Applications and Worksharing. 2014.

[3] Vishal Doshi, Sneha Tuteja, Krishna Bharadwaj, Davide Tantillo, Thomas Marrinan, James Patton, G. Elisabeta Marai, "StickySchedule: an interactive multi-user application for conference scheduling on large-scale shared displays", Proceedings of the 6th ACM International Symposium on Pervasive Displays (PerDis '17), Lugano, Switzerland, June 7-9, 2017. http://dx.doi.org/10.1145/3078810.3078817

[4] https://dzone.com/articles/my-security-notes

[5] Security Assertion Markup Language V2.0 Technical Overview Committee Draft 02 25 March 2008 http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-tech-overview-2.0.html

[6] Security Assertion Markup Language V2.0 Technical Overview Committee Draft 02 25 March 2008 http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-tech-overview-2.0.html

[7] OpenID Connect specifications https://openid.net/specs/openid-connect-core-1 0.html

[8] OpenID2.0 specification https://openid.net/specs/openid-authentication-2 0.html

[9] Ubisecure's white paper "SAML vs OAuth 2.0 vs OpenID Connect" https://www.ubisecure.com/about/resources/saml-oauth-openid-connect/

[10] Great article about Federated Identity Management - may be useful later Chadwick, David W. (2009) Federated Identity Management. In: Aldini, Alessandro and Barthe,Gilles and Gorrieri, Roberto, eds. FOSAD 2008/2009. LNCS (5705). Springer-Verlag, Berlin,pp. 182-196. ISBN 978-3-642-03828-0. https://kar.kent.ac.uk/30609/1/FederatedIdManChapter.pdf