Federation monitoring

Jaime Perez Crespo
Middleware Engineer

Virginia Martin-Rubio Pascual
System Administrator

RedIRIS / Red.es
Edificio Bronce, Plaza de Manuel Gomez Moreno, s/n
28020 Madrid, Spain
{jaime.perez,virginia.martinrubio}@rediris.es

Keywords

Monitoring, federations, nagios, jmeter

Abstract

One essential aspect of every network service is the ability to monitor its status along time. There are many mature solutions currently available in the market, but there's still an important lack of what refers to complex web application monitoring. Identity federations, and more precisely, their web profiles, have become critical for institutions, so there's a clear need to develop a monitoring system capable of determining the exact status of the infrastructure, as accurate and close to the user's experience as possible.

1. Goals and requisites

The motivation for the development presented here is clear and concise: to be able to monitor the status of the infrastructure running an identity federation, like the one held by RedIRIS in Spain. Since such infrastructures are a complex amalgam of web applications that interact between each other, sharing data without noticing the user, and automated by heavily relying on features provided by both HTTP protocol and application level languages run by web browsers, it is really difficult to emulate the behavior of users accessing an online resource and identifying themselves on their own institution. So our main concern for such a monitoring system is to be able to emulate completely all the steps made by both the users and their web browsers to access an application through a federation. Unfortunately, this has proven to be a difficult goal to achieve.

We imposed two requisites to be met by this project:

1. First of all, the product of this research must be compatible with the monitoring infrastructure already running internally in RedIRIS, based on Nagios. That would allow us to focus on the problem itself and forget completely about the underlying technology, notifications, reports, graphics and all the stuff usually related to such systems, but completely irrelevant for the goal we wanted to achieve.

2. Secondly, we should find a way to monitor any Identity Provider in the federation, no matter which technology they are using internally or what procedures they have to authenticate users. That means the solution should be flexible enough to manage any federation protocols, web redirections or authentication methods, assuming users would be able to authenticate, at least, by using an username and a password.

Another key aspect for such a monitoring system is to provide automated tools to the final users, that is, administrators of Identity Providers, so they can manage their own preferences and gather statistics about their service. This is crucial not only for the functionality itself, but also because of security considerations. Users must identify themselves (through the very same federation we are monitoring) and access strictly the information they own, including statistics. No data should be disclosed to other users rather than the owners of each Identity Provider.

2. Implementation

The first issue we met when we began to develop this federation monitor was the need to emulate all redirections made by a web browser when someone wants to access to a federated service. Firstly, the user has to select his Identity Provider in the WAYF (*Where Are You From*), and then he will be redirected to the login page of his institution, where he must provide his credentials (user name and password). Finally, if the information the user entered is correct, he will be redirected back to the service. Unfortunately, not all scenarios are that simple, and we've found that many of our Identity Providers make internal redirections to cope with their own *Single Sign On* systems, to set additional cookies, to manage statistics, or even to show a form to the user. That makes it really hard to automate the login flow as most of those redirections are implemented diversely with Javascript language. And what it makes it worse, all the different software we tested lacks support for Javascript, so such kind of redirections must be triggered somehow manually.

Apache JMeter was the solution we found that best fitted our needs. JMeter is an open source software designed to load test functional behavior and measure performance. This software allowed us to emulate any GET and POST queries needed to access a federated service and test if everything is working fine. Going further, it's also useful to determine the possible reason of a failure when login did not succeeded, as failures might be due to the Identity Provider or the federation infrastructure itself, and gather statistics about access times at every single step.

One of they key aspects that made us go for Apache JMeter is its ability to trigger events based on results of regular expression evaluation. This is a complex functionality that helps compensate the lack of Javascript language support, and allowed us to design a process for each request that searches for a non-HTTP redirection and automatically follows it, in a style very much like a programming language: loops, conditional statements, variables and so on. It is powerful enough to allow us support even hidden forms that are automatically submitted when a page loads, which is somehow frequent as many Identity Providers in our federation are SAML-based or have *Single Sign On* systems that require such kind of features. In the end, although not perfect, JMeter provided us with the capabilities to closely emulate the behavior of a real web browser.

The last challenge was how to combine JMeter with the existing infrastructure based on Nagios, to integrate this new development with our current monitoring system, so we are now able to notify institutions when a problem is found within their Identity Providers. We solved this by developing a simple shell script that behaves as a Nagios plugin, acting as a common interface for the underlying JMeter test plans. Furthermore, by using Nagios with a database backend, we can easily offer technicians an alternate and simpler web interface to the monitoring, rather than the more complex and less customizable default one. This way they can check the status of their own Identity Providers, manage their preferences and even retrieve periodical reports.

We've been testing this new infrastructure in a pilot service for almost six months, with the help of up to ten institutions and a total of twelve Identity Providers. The pilot has recently gone into production and federation administrators have now a control panel available where to check the status of their providers. This control panel is also the Service Provider we are using for the automated monitoring, so we are able to run additional verifications about, for example, the attributes received from each Identity Provider tested, and trigger warnings according to our own attribute exchange recommendations (such like missing or malformed attributes). Additionally, we are in the process of adapting this development to build a parallel monitoring infrastructure for the *eduroam* network, as part of our effort to offer our customers quality services and support.

3 Author Biography

Jaime Perez works as a Middleware Engineer at RedIRIS. He received his Bachelor's degree in Computer Systems Engineering at the Rey Juan Carlos University of Madrid in 2004 and worked there until joining RedIRIS in November 2006. Jaime has presented at the TERENA Networking Conference in previous years and is involved in research groups and activities within the Spanish and TERENA communities focused on middleware architectures, including the Task Force on European Middleware Coordination and Collaboration (TF-EMC2) and the European Committee for Academic Middleware (ECAM). He leads a work item dedicated to reputation systems.

Virginia Martin-Rubio works as a System Administrator at RedIRIS since 2008, after she received her Bachelor's degree in Telecommunications Engineering at Carlos III University of Madrid. She has worked for two years inside the EGEE project (Enabling Grids for E-sciencE) and nowadays she is involved in several activities of the EGI (European Grid Iniciative).