



Avaliação da Clearinghouse do GENI
RF - RELATÓRIO FINAL DA EXTENSÃO

Autor:

Joao F. SANTOS

Orientador:

Natalia C. FERNANDES, D.Sc.

Data:

Abril/2016

Sumário

Introdução	1
Ambiente de testes	2
Lições Aprendidas	4
Conclusão e avaliação	5
Apêndice: Scripts para instalação e configuração	6
Referências	8

Introdução

O Fibre foi criado para ser um ambiente de experimentação em larga escala de infraestrutura e aplicações em Internet do Futuro, a partir da federação de *testbeds* entre instituições do Brasil e da Europa. Ele permite pesquisa e experimentação em novas arquiteturas de rede e Internet do futuro, de forma que seus membros podem estudar novas abordagens, avaliar arquiteturas e protocolos existentes e criar possíveis soluções para superar as limitações atuais das redes de comunicação de dados. Para cumprir tais requisitos, o Fibre possui em suas *testbeds* diferentes *Control and Management Frameworks* (CMFs), capazes de prover aos membros da federação funcionalidades complementares para a criação e execução de experimentos. Tais funcionalidades compreendem o uso de recursos OpenFlow para controle completo do substrato de rede e criação de arquiteturas de rede, recursos de redes sem fio para desenvolvimentos de novas formas de comunicação sem fio, e uso de máquinas virtuais como servidores ou para testes de novos protocolos de rede [1].

Este cenário com diversos CMFs garantiu que os membros da federação fossem capazes de realizar os mais variados tipos de experimentos, contudo, ele também dificultava o processo de aprendizado e uso dos recursos da federação. Cada um dos CMFs usados no Fibre possui uma forma de uso própria, de forma que um membro da federação precisa primeiro aprender a descrever seu experimento usando o CMF escolhido antes de ser capaz de executar o experimento em questão. Além disso, a arquitetura heterogênea dos CMFs impossibilitava a criação de ferramentas e soluções globais para os problemas da federação. Na reunião do comitê técnico do Fibre em Fevereiro de 2016 foi decidido adotar um único CMF para dar seguimento a evolução do Fibre, e a escolha unânime foi para o uso do OMFv6 como substituto dos CMFs usados até então no Fibre, i.e. OMFv5.4, OCF e ProtoGENI.

Enquanto isso, há planos de tornar o Fibre compatível com o arquitetura *Slice-based Federation Architecture* (SFA) para permitir o uso de clientes, ferramentas e soluções de forma *out of the box*, desenvolvidos por terceiros e também compatíveis com SFA. Para permitir o uso destes clientes, ferramentas e soluções pelos membros da federação, o Fibre precisa de uma *Clearing House* (CH), a entidade central da federação *Global Environment for Network Innovations* (GENI). Ela é utilizada para a descoberta de recursos das ilhas da federação e possui as autoridades responsáveis pela autenticação e autorização do acesso dos membros aos recursos da federação. A CH GENI usa o *GENI Aggregate Manager (AM) Application Programming Interface* (API), protocolo de comunicação criado para interagir com os AMs das ilhas e com as ferramentas dos membros [2]. O *Broker* do OMFv6 é o órgão central do AM de cada ilha com OMFv6, ele é compatível com clientes usando SFA, mas não se sabe se esta compatibilidade é limitada, ou se todas as funcionalidades estão disponíveis. Neste trabalho propomos implementar a CH GENI e avaliar sua interação com um AM OMFv6.

Ambiente de testes

Para começar a construir o ambiente de testes nós decidimos implementar a CH em uma máquina virtual pela facilidade de retroceder a estados com configurações passadas e para posteriormente disponibilizar a imagem criada no projeto. Utilizamos o *hypervisor* XEN instalado numa máquina com o sistema operacional Ubuntu 15.04 e com as especificações vistas na Tabela 1.

Especificações:	
Processador:	Intel Core i7-2600 CPU @ 3.40GHz
Placa-mãe:	Commel LV67G, Mini-ITX LGA1155
Armazenamento:	Kingston SV300S37A60G, SSD 60GB
Memória:	Kingston 9905471014.A00LF, 4GiB DDR3 @ 133MHz
Rede:	2x Intel 82574L Gigabit Network Connection
Wireless:	AR93xx Wireless Network Adapter

Tabela 1. Especificações técnicas da máquina física.

A máquina virtual da CH possui um único núcleo de processamento, 1 GiB de memória RAM, 10 GB de armazenamento e usa o CentOS 7 como sistema operacional. A instalação do software da CH GENI seguiu o passo a passo disponibilizado no repositório oficial da CH [3], resumido a seguir:

1. Adicionar o repositório RPM que contém todo o software da CH GENI.
2. Instalar o pacote da CH e todas as suas dependências.
3. Editar o arquivo de configuração "*parameters.json*" no diretório "*/etc/geni-chapi/*" com o *Full Qualified Domain Name* (FQDN) do servidor da própria CH e da base de dados.
4. Instalar a sistema de gerenciamento de base de dados relacionais PostgreSQL.
5. Configurar as variáveis de ambiente com diretórios da instalação da software da CH.
6. Inicializar a base de dados.
7. Configurar a autoridade que valida os certificados das entidades da federação.
8. Inicializar os serviços da CH.
9. Configurar o sistema para usar os certificados da CH para SSL.
10. Configurar a raiz de confiança da CH.
11. Instalar o AMSoil e reiniciar os servidor web Apache.

Vale notar que um mesmo servidor pode hospedar a CH e a base de dados da federação ao mesmo tempo, portanto um mesmo FQDN pode ser usado para ambos no passo #3. Já o AMsoil é um arcabouço desenvolvido para facilitar o desenvolvimento de AMs, ele implementa a *GENI Aggregate Manager API* (GENI AM API) e cuida de todo o processo de comunicação entre o controlador do recurso e as ferramentas e/ou clientes usados para se interagir com o dado AM [4]. A CH utiliza a sua implementação do AMSoil da GENI AM API para se comunicar com os AMs da federação.

Neste momento a instalação da CH foi concluída e teoricamente seria possível testar os seus serviços. A interação com estes serviços é feita através da *Common Federation API* (CFA), um protocolo de comunicação desenvolvido para padronizar toda a interação com os serviços contidos na CH [5]. A instalação padrão da CH GENI trás um protótipo de cliente CFA, usado justamente para testar como e se os serviços da CH estão funcionando. Este cliente foi implementado usando Python 2.7, pode ser localizado em `"/usr/share/geni-ch/chapi/chapi/tools/client.py"` e implementa apenas uma parte dos métodos acessíveis dos serviços. A partir do uso de tal cliente pudemos constatar que a CH foi instalada e configurada com sucesso, respondendo como esperado a requisições tais como: versão do CFA usada por cada serviço e lista de serviços oferecidos pela CH.

O próximo passo foi adicionar usuários à base de dados de membros da federação e adicionar uma máquina com OMFv6 como um AM federado. Entretanto, não foi possível usar o protótipo de cliente para criação de membros pois a chamada para o método de criação de membros não havia sido implementado no protótipo de cliente, uma vez que o cliente não foi desenvolvido com essa intenção. A criação de membros deveria ficar a cargo do Portal GENI, uma outra entidade da federação, responsável pela validação da identidade dos membros e a criação de suas contas na federação. O Portal GENI implementa a CFA para comunicação com a CH e se comunica com ela para atualizar a base de dados de membros da federação com novas contas, assim como para obter os certificados das novas contas assinados pela CH.

Criamos uma outra máquina virtual com as mesmas especificações e sistema operacional da máquina virtual usada para a CH. Seguimos o passo a passo disponibilizado no repositório oficial do Portal GENI [6], mas não conseguimos concluir a instalação do Portal. O Portal exige que um *Identity Provider* (IdP) esteja devidamente instalado e configurado, para ser utilizado na verificação da identidade dos membros. Entretanto, não tivemos tempo para implementar nosso próprio IdP com Shibboleth ou tentar usar uma alternativa como a Comunidade Acadêmica Federada (CAFe) antes do término do prazo do projeto de extensão.

Em resumo, a CH aparentemente foi devidamente instalada, mas não foi possível criar usuários e testar sua interação com um AM usando OMFv6, uma vez que a princípio não é possível criar membros diretamente na CH. Um cliente CFA é necessário para a criação de membros e o protótipo que acompanha a instalação da CH não implementa as chamadas necessárias, deixando tal função para o Portal GENI, mas o Portal requer a existência de um IdP para verificar a identidade dos usuários.

Lições Aprendidas

A instalação da CH exige uma série de passos relativamente repetitivos e que em grande parte dos casos não exige que o administrador do sistema entre com dados diferentes do guia de instalação apresentado. Portanto, para deixar o processo de instalação e configuração mais rápido, nós criamos scripts que automatizam grande parte do processo, bastando que eles sejam executados na ordem correta e com a ação do administrador em momentos necessários, tais como para inserir o FQDN dos servidores da CH e da base de dados num arquivo de texto. O código dos scripts pode ser encontrado no Apêndice.

Existe uma certa confusão na literatura devido a muitas ferramentas ou entidades alegarem ser compatíveis com SFA, enquanto na verdade, é possível dizer que elas se distinguem em dois grupos: as que se comunicam com a CH e implementam a CFA, e as que se comunicam com os AMs e implementam a GENI AM API. Por exemplo, o Portal GENI implementa apenas a CFA, a *geni-lib* se comunica AMs conhecidos e usa a GENI AM API [7], enquanto clientes como o Omni utilizam ambos os protocolos [8]. Ambas as APIs são baseadas no *Extensible Markup Language Remote Procedure Call* (XML-RPC), um protocolo para chamada remota de procedimento que utiliza *Extensible Markup Language* (XML) para codificar suas mensagens e *Hypertext Transfer Protocol* (HTTP) como mecanismo de transporte.

O cliente criado pelos desenvolvedores da CH é capaz de se comunicar com os diferentes serviços da CH, listados a seguir:

Service Registry Serviço responsável pela descoberta de recursos e outros serviços, é contato inicial de qualquer cliente acessando a CH. Por padrão é acessível através de chamadas CFA no endereço: "*CH_HOST:8444/SR*"¹.

Member Authority Autoridade responsável por gerenciar os membros da federação, através dela os membros são adicionados ou removidos, assim como suas permissões e atributos alterados. Por padrão é acessível através de chamadas CFA no endereço: "*CH_HOST:443/MA*"¹.

Slice Authority Autoridade responsável por gerenciar os projetos e os slices do membros da federação. Por padrão é acessível através de chamadas CFA no endereço: "*CH_HOST:443/SA*"¹.

Entramos em contato com os desenvolvedores da CH e do Portal GENI para tirar dúvidas sobre o processo de instalação e configuração da CH e do Portal. Descobrimos inconsistências no passo a passo da instalação do Portal e contribuimos com a correção do guia de instalação disponibilizado no repositório oficial. Após percebermos que não haveria tempo hábil para tentar utilizar a CAFe como IdP ou criar nosso próprio IdP com Shibboleth, resolvemos tentar estender o protótipo de cliente CFA para possibilitar a criação de contas. Conseguimos realizar esta extensão e os membros foram devidamente criados, nossa contribuição também foi submetida ao repositório oficial da CH GENI [9].

¹Onde *CH_HOST* é o FQDN do servidor da CH.

Conclusão e avaliação

Não foi possível realizar os testes inicialmente pretendidos no começo do trabalho devido a falta de um IdP para uso no Portal GENI. Mas fomos capazes de compreender melhor como a CH funciona e a divisão de seus serviços, além de entender as APIs para acesso aos serviços da CH e comunicação com AMs.

É possível instalar a CH GENI e começar a operar sobre ela, os desenvolvedores foram dispostos a ajudar quando nos deparamos com empecilhos e problemas na configuração, assim como incentivaram nossa contribuição. Ainda é preciso estudar como é a interação entre a CH GENI e AMs com OMFv6, mas primeiramente é necessário avaliar a possibilidade de utilizar a CAFe como IdP para o Portal, ou a criação de um IdP com Shibboleth do zero, ou até mesmo continuar a incrementar o protótipo de cliente CFA para dispensar o uso do Portal e do IdP para os testes.

Apêndice: Scripts para instalação e configuração

Script 1: Atualizar o sistema, desabilitar o SELinux e reiniciar a máquina.

```

1 # update the OS
2 sudo yum update -y
3
4 # Install the EPEL repository
5 sudo yum install -y epel-release
6
7 # Disable SELinux
8 sudo sed -i -e "s/SELINUX=enforcing/SELINUX=disabled/g" \
9 /etc/selinux/config
10
11 # Reboot
12 sudo reboot

```

Script 2: Adicionar o repositório RPM que contém o pacote da CH e instalá-lo.

```

1 # GENI RPM repository to CentOS
2 sudo sh -c "echo $'[geni]\nname = GENI software \
3 repository\nbaseurl = http://www.gpolab.bbn.com/\
4 experiment-support/gposw/centos/\$releasever/os/\
5 $basearch/' >> /etc/yum.repos.d/geni.repo"
6
7 # Install the GENI ClearingHouse RPM package
8 sudo yum install -y --nogpgcheck geni-chapi
9
10 # Move the example parameters to the right place
11 sudo cp /etc/geni-chapi/example-parameters.json \
12 /etc/geni-chapi/parameters.json

```

Neste momento o administrador deve editar o arquivo de texto */etc/geni-chapi/parameters.json* para adicionar valores dos FQDNs dos servidores que hospedam a CH e a base de dados, vale notar que eles podem ser colocados.

Script 3: Configurar a CH com os parâmetros entrados no passo anterior e instalar o PostgreSQL.

```

1 # Configure the ClearingHouse with the set parameters
2 sudo /usr/sbin/geni-install-templates
3
4 # Make the PostgreSQL installation script executable
5 sudo chmod +x \
6 /usr/share/geni-chapi/templates/install_postgresql.sh
7
8 # Change directory to the PostgreSQL installation script
9 cd /usr/share/geni-chapi/templates/
10
11 # Install PostgreSQL
12 sudo ./install_postgresql.sh

```

O administrador deve editar o arquivo "*~/bashrc*" e acrescentar as seguintes linhas ao fim dele:

```
1 # Append to bashrc of current user
2 export CH_DIR=/usr/share/geni-ch
3 export CHAPI_DIR=/usr/share/geni-chapi
4 export CH_HOST='geni-install-templates --print_parameter ch_host '
5 export DB_HOST='geni-install-templates --print_parameter db_host '
6 export DB_USER='geni-install-templates --print_parameter db_user '
7 export DB_DATABASE=\
8 'geni-install-templates --print_parameter db_name '
9 export DB_PASSWORD=\
10 'geni-install-templates --print_parameter db_pass '
11
12 export PSQL="psql -U $DB_USER -h $DB_HOST $DB_DATABASE"
```

Script 4: Carregar as novas variáveis de ambiente e configura as permissões da base de dados da CH.

```
1 # Reload environment variables
2 source ~/.bashrc
3
4 # Create .pgpass file
5 echo "$DB_HOST:*:$DB_DATABASE:$DB_USER:$DB_PASSWORD" > ~/.pgpass
6
7 # Change .pgpass permissions
8 chmod 0600 ~/.pgpass
```

Os passos para inicialização dos serviços podem ser encontrado em [3].

Referências Bibliográficas

- [1] S. Sallent, A. Abelém, I. Machado, L. Bergesio, S. Fdida, J. Rezende, S. Azodolmolky, M. Salvador, L. Ciuffo, and L. Tassiulas, “FIBRE project: Brazil and Europe unite forces and testbeds for the Internet of the future,” in *Testbeds and Research Infrastructure. Development of Networks and Communities*, pp. 372–372, Springer, 2012.
- [2] T. Mitchell, “GENI Aggregate Manager API,” in *Geni Engineering Conference 2010 (8th GEC)*, pp. 1998–2001, 2010.
- [3] “geni-ch/INSTALL-centos.md at develop · GENI-NSF/geni-ch.” <https://github.com/GENI-NSF/geni-ch/blob/develop/INSTALL-centos.md>. (Acessado em 21/04/2016).
- [4] “GENI-NSF/geni-soil: Service library to support GENI servers.” <https://github.com/GENI-NSF/geni-soil>. (Acessado em 21/04/2016).
- [5] “Common Federation API v1 – GENI: geni.” <http://groups.geni.net/geni/wiki/CommonFederationAPIv1>. (Acessado em 21/04/2016).
- [6] “geni-portal/INSTALL-centos.md at develop · GENI-NSF/geni-portal.” <https://github.com/GENI-NSF/geni-portal/blob/develop/INSTALL-centos.md>. (Acessado em 21/04/2016).
- [7] “geni-lib — geni-lib 0.6.” <http://geni-lib.readthedocs.org/>. (Visited on 21/04/2016).
- [8] “GENI Tools Overview – GENI: geni.” <http://groups.geni.net/geni/raw-attachment/wiki/GEC20Agenda/DeveloperTopics/>. (Acessado em 21/04/2016).
- [9] “Added the ”create_member”method in the client.py · geni-nsf/geni-ch@7077be6.” <https://github.com/GENI-NSF/geni-ch/commit/7077be6151ee2836dee6415c44fd576e0587c970>. (Accessed on 04/21/2016).