

Grids Computacionales Computing Grids

Vanessa Hamar

Centro Nacional de Cálculo Científico Universidad de Los Andes (CECALCULA),
Corporación Parque Tecnológico de Mérida, Mérida 5101, Venezuela
vanessa@ula.ve

Resumen

Hoy en día, donde la mayoría de las personas tienen suficiente poder de cómputo en su propia computadora, se les hace innecesario compartir recursos para lograr sus propósitos.

Pero por otro lado están los científicos, para los cuales es prácticamente imposible hacer ciencia sin el uso de computadoras, cada día se encuentran con problemas cada vez más complejos para los cuales requieren una mayor cantidad de recursos computacionales, y muchas veces la capacidad de cómputo que puede proveer un cluster no es suficiente para satisfacer estos requerimientos. De allí que los científicos sueñen con equipos con un poder de cómputo y espacio de almacenamiento ilimitado, que permita interactuar con sus colegas en sitios remotos, compartir la data, los procedimientos y los resultados de manera sencilla, eficiente y segura.

Para llenar esta necesidad han surgido los grids computacionales. Un grid es computación distribuida que utiliza un software, el middleware, que coordina y permite compartir los recursos, las aplicaciones, el almacenamiento de datos y los recursos de red, entre organizaciones dinámicas y distribuidas geográficamente.

Aun los grids computacionales no han sido desarrollados en su totalidad, en este artículo se presentan los conceptos básicos de esta tecnología y como manejan la seguridad actualmente..

Abstract

Today, where most people have enough computing power on their own machines, makes it unnecessary to share resource to solve their problems.

On the other hand there are the scientists, for whom it is practically impossible to make science without using computers, every day the problems are more complex and require a greater amount of computing resources, and often the power provided by a cluster is not sufficient to satisfy these requirements. Therefore

the scientists dream with a limitless power and storage space machine, that allows to interact with its colleagues in remote sites, to share the data, procedures and results in a simple, efficient and safe way.

In order to fill this gap computing grids have arisen. Grid is distributed computing that uses a software, the middleware, that coordinates and allows to share the resources, the applications, the data storage and the network resources, between dynamic and geographically distributed organizations.

Grids have not yet been totally developed, in this paper the basic concepts of this technology and how they manage the security today.

1. Introducción

Hoy en día las ciencias han avanzado de tal manera que han ido emergiendo una serie de necesidades tecnológicas. Entre ellas se encuentran las de compartir recursos, almacenar y analizar grandes cantidades de data, aunado al hecho de que los usuarios y las instituciones están distribuidos geográficamente. Para dar una solución a estas necesidades, han surgido los grids computacionales.

Los grids computacionales ofrecen a las organizaciones beneficios como el de aumentar su capacidad de cómputo y almacenamiento al compartir los recursos existentes y de esta manera ahorrar dinero evitando la adquisición de nuevos equipos.

El termino grid aparece como consecuencia de su analogía con una red eléctrica, cuya principal idea es que los usuarios puedan conectarse en cualquier sitio para utilizarlo sin tener que preocuparse desde donde esta obteniendo ese poder de cálculo.

Aun los grids computacionales no han sido desarrollados en su totalidad, y las necesidades de flexibilidad, seguridad y de coordinación de los recursos entre una colección dinámica de individuos e instituciones, los que se define como Organizaciones

Virtuales (*VO* por sus siglas en inglés), hacen que el software existente siga en continuo desarrollo.

Para elegir entre las diferentes plataformas de grid existentes es necesario conocer los requerimientos de los usuarios, los estándares y las diferentes plataformas que están disponibles hoy en día.

El objetivo de este documento es presentar algunos de los conceptos relacionados con los grids computacionales, los estándares y como manejan la seguridad.

2. Antecedentes

La primera definición de grid fue dada por Ian Foster y Carl Kesselman: “Un grid computacional es una infraestructura de hardware y software que provee acceso consistente a bajo costo a recursos computacionales de alto nivel” [2]

Otra definición dada por Len Kleinrock en 1969: “Probablemente veremos el esparcimiento de las utilidades de computación tal como se presentan la electricidad y el teléfono, como servicios individuales en casas y oficinas a través de todo el país”. [1]

La computación distribuida nace en los comienzos de los años 70, cuando las computadoras comenzaron a conformar redes y de allí surgió la idea de utilizar los ciclos de CPU ociosos.

En 1973, el Xerox Palo Alto Research Center (*PARC*) instaló la primera red Ethernet y el primer esfuerzo de computación distribuida estaba encaminado. Los científicos John F. Shoch y Jon A. Up crearon un *gusano*, así fue como lo llamaron, y su visión fue moverlo entre las máquinas utilizando los recursos que estuvieran libres con el fin de sacar provecho de los mismos.

Por otro lado, Richard Crandall, comenzó a trabajar en redes de computadores en *NeXTTM*. Crandall instaló un software que llamó *Zilla*, que permitía que las máquinas, cuando estas no estuvieran siendo utilizadas, realizaran cálculos y combinaran esfuerzos con otras máquinas en la red. En 1991, *Zilla* ganó el premio de ciencia Computerworld Smithsonian.

La Internet brinda una nueva escala. Con la madurez de Internet, en los años 90 surgen dos proyectos de computación distribuida a nivel global que llaman la atención.

El primero de estos proyectos innovadores utiliza cientos de computadoras independientes para crear códigos encriptados. Este proyecto, es el primero en su tipo, y fue llamado *distributed.net*, mejor conocido como *dnet*.¹

¹ <http://www.dnet.net>

El segundo y el más exitoso y popular en la historia de la computación distribuida es el *SETI@home_project*². Más de dos millones de personas, el mayor número de voluntarios para cualquier proyecto de computación distribuida en la Internet hasta la fecha, instalaron el agente de software desde que el proyecto comenzó en mayo de 1999. Este proyecto prueba de manera concluyente que la computación distribuida puede acelerar los resultados de proyectos computacionales manteniendo los costos en niveles manejables.

Los grids computacionales están relacionados con la coordinación de recursos compartidos y la solución de problemas en organizaciones virtuales dinámicas y multi institucionales. [3]

3. Grids computacionales

De los conceptos anteriores podemos definir un grid computacional como un sistema de software y hardware que permite compartir recursos, datos, espacio de almacenamiento y solucionar problemas en una o varias organizaciones a través de la red de una manera transparente al usuario.

Tenemos que el hardware de un grid presenta las siguientes particularidades:

- Es numeroso.
- Pertenece y es manejados por organizaciones y/o individuos diferentes con diferentes políticas de seguridad y distintas prácticas.
- Está expuesto a fallas.
- Es heterogéneo.
- Está conectado por redes diferentes y a distintos niveles.
- Está esparcido geográficamente.

El software es complejo y debe agrupar los recursos (CPUs, aplicaciones, etc.) y hacerlos accesible de manera segura, transparente, oportuna y en forma fácil a usuarios y aplicaciones.

El grid está asociado a un software llamado *middleware*, que es el que va a actuar como intermediario y de coordinador de los diversos recursos de hardware y software necesarios para manejar el sistema en su totalidad, incluyendo los sistemas manejadores de recursos, planificadores, etc. De esta manera se crea lo que el usuario percibe como un sistema único.

Según IBM existen tres tipos de grids³:

² <http://setiathome.berkeley.edu>

³ <http://www-1.ibm.com/grid/>

- **Intragrids:** Pertenecen a una sola institución
- **Extragrids:** Disponibles para varias organizaciones autorizadas.
- **Intergrid:** Aquellos que utilizan Internet para construir grids mas amplios.

4. Clusters vs. grids

No hay duda de que los clusters son conceptualmente similares a los grids. Lo más importante es que ambos dependen de un *middleware* para proveer la virtualización necesaria para lograr que un sistema computacional de múltiples computadoras aparezca como un único sistema para el usuario.

Aún cuando son similares entre si, existen importantes diferencias entre los clusters y los grids. Los clusters generalmente están localizados en un mismo espacio físico, mientras que los grids computacionales van a estar dispersos. Los clusters son homogéneos y estáticos mientras que los grids son heterogéneos y dinámicos.

La tecnología de grids, como todas, ofrece a las instituciones ventajas y desventajas. Entre las ventajas se encuentran el aumentar la velocidad de computo sin aumentar los costos al compartir la utilización de los recursos computacionales que no estén siendo utilizados y que existen dentro de las instituciones, además de permitir compartir estos recursos entre instituciones fácilmente.

Por otro lado, presenta la desventaja de ser una tecnología que esta emergiendo y la gran cantidad de requerimientos que debe abarcar como la gran diversidad de hardware y software existente, la necesidad de simplificar su manejo, ocuparse de la seguridad, recuperarse después de fallas, llegar a acuerdos de niveles de servicio y otras, hacen que existan deficiencias a la hora de la adquisición, instalación y soporte de estas herramientas.

5. Estándares

Para evaluar las herramientas de los grids es importante tomar en cuenta los estándares como medio para permitir la interoperabilidad de una infraestructura común. Ian Foster, propone una lista que captura la esencia de lo que debería ser un grid:

- Coordinar los recursos que no están sujetos a un control centralizado. Un grid computacional integra y coordina recursos y usuarios que están en diferentes dominios, unidades administrativas, etc.
- Utiliza estándares abiertos, protocolos de

propósitos generales e interfaces. Un grid es construido con múltiples propósitos e interfaces que están dirigidos fundamentalmente a la autenticación, autorización, descubrimiento de recursos y acceso a recursos. De otra manera se estaría trabajando con una aplicación específica para un sistema.

- Un grid computacional utilizara los recursos que lo constituyen, los cuales deberán ser usados de una manera coordinada para entregar servicios de calidad relacionados, por ejemplo: tiempo de respuesta, disponibilidad, seguridad y rendimiento al procesar, además de la asignación de los múltiples tipos de recursos para satisfacer las complejas exigencias de los usuarios, de modo que la utilidad del sistema combinado sea perceptiblemente mayor que la suma de sus piezas.[1]

Para tratar de unificar criterios en la estandarización de los grids, se crea la organización Global Grid Forum (*GGF*)⁴. Entre los estándares que proponen están:

- Open Grid Services Architecture (*OGSA*)
- Web Service Resource Framework (*WSRF*)
- Web Service Description Language (*WSDL*)

OGSA es un refinamiento de la arquitectura de Servicios Web. Esta específicamente diseñada para soportar los requerimientos propios del grid. Su principal función es asegurar que sistemas de recursos heterogéneos puedan comunicarse y compartir información.

WSRF provee un conjunto de operaciones que permiten a los servicios web implementados comunicarse con los servicios de recursos y de esta manera la data capturada pueda ser almacenada y recuperada.

WSDL es un lenguaje basado en *XML* que provee la manera de describir y de comunicación utilizando servicios web.

La herramienta Globus⁵ ha sido implementada utilizando los estándares de organizaciones tales como *IETF*⁶(*Internet Engineering Task Force*), *W3C* (*The World Wide Web Consortium*)⁷, *OASIS* (*Organization for the Advancement of Structured Information Standards*)⁸ y el *GGF*. Algunos ejemplos de estos

⁴ <http://www.ggf.org>

⁵ <http://www.globus.org>

⁶ <http://www.ietf.org>

⁷ <http://www.w3.org>

⁸ <http://www.oasis-open.org>

estándares utilizados por la herramienta Globus son los siguientes:

- **SSL/TLS v1** (Secure Sockets Layer (SSL v2/v3) Transport Layer Security (TLS v1)) (IETF)
- **LDAP v3** (Lightweight Directory Access Protocol) (IETF)
- **X.509 Proxy Certificates** (IETF)
- **SOAP** (Simple Object Access Protocol) (W3C)
- **HTTP** (Hyper Text Transfer Protocol) (W3C)
- **GridFTP v1.0** (Grid File Transfer Protocol) (GGF)
- **OGSA-DAI** (Database Access and Integration Services) (GGF),
- **WS-Agreement** (Web Services Agreement Specification)
- **WSDL 2.0** (Web Service Description Language)(W3C),
- **WSDM** (Web Service Distributed Management), SAML (Security Assertion Markup Language) (OASIS),
- **XACML** (eXtensible Access Control Markup Language)(OASIS)

TLS y su predecesor *SSL* son protocolos de criptografía, estos proveen una conexión segura en la Internet para la transferencia de datos. Existen algunas diferencias entre *SSL 3.0* y *TSL 1.0* pero el protocolo básicamente sigue siendo el mismo. Entre las medidas de seguridad de este protocolo se incluyen los certificados digitales.

LDAP es un protocolo de aplicación que se utiliza para consultar y modificar servicios de directorio sobre *TCP-IP*. Un servicio de directorio se define como un conjunto de información con atributos similares organizados de manera lógica y alfabéticamente.

X.509 Proxy Certificates es un estándar para las infraestructuras de clave pública (*PKI*). Este especifica cuales son los formatos estándares para los certificados de clave pública y el algoritmo que se utiliza para la validación de los certificados.

SOAP es un protocolo para el intercambio de mensajes basados en *XML*, normalmente utilizando *HTTP/HTTPS*. Existen varios tipos de patrones de mensajes en *SOAP*, siendo el mas común las llamadas a procedimientos remotos (*RPC*).

HTTP es un protocolo de comunicación utilizado para transferir información en la Web. Su propósito original era proveer una manera de publicar y recuperar paginas *HTML*. *HTTP* es un protocolo de petición/respuesta entre clientes y servidores. La comunicación se establece utilizando *TCP*.

GridFTP es un protocolo de transferencia de datos basado en el protocolo de Internet *FTP*, el cual fue

extendido para proveer operaciones de alto rendimiento y llenar los parámetros de seguridad requeridos por el grid.

OGSA-DAI es un producto del *middleware* que permite a las bases de datos, bien sean relacionales o *XML*, ser accedidas dentro de los grids.

WSDM es un estándar de servicios web utilizado para manejar y monitorear el estatus de los servicios.

XACML es una política de control de acceso declarativa implementada en *XML*, utilizando un modelo de procesamiento para describir como interpretar estas políticas.

Entre las ventajas de utilizar estos estándares se tiene:

- Es usado ampliamente.
- Esta herramienta permite el uso de partes de los componentes, solo los que el usuario necesite.
- Es portable, corre en un gran número de plataformas.
- Se integra con sistemas por lotes o sistemas de manejo de colas de trabajos existentes, por ejemplo *Sun Grid Engine*, *PBS*, *LSF*.
- Se integra con servicios de bajo nivel como *MPI*.
- La seguridad esta presente en todos sus componentes.
- Soporta *Web Services*.
- Una gran cantidad de herramientas han sido implementadas para trabajar sobre Globus.

6. Seguridad en el Grid

La idea principal del grid es compartir recursos y datos, esto implica hacer un mayor énfasis en la seguridad de los diferentes sitios que conforman un grid.

Entre los riesgos a los que están expuestos los grids se encuentran:

- Recibir ataques de otros sitios: entre los diferentes sitios que conforman un grid se crean relaciones de confianza, por lo cual si un sitio esta comprometido el resto de los sitios también pueden estar comprometido.
- Distribución de data de manera inapropiada y acceso a información sensitiva: los usuarios tienen la posibilidad de almacenar data en diferentes sitios del grid, esta data puede ser sensitiva y confidencial.
- Explotación de huecos de seguridad: como los ambientes en el grid son dinámicos, heterogéneos y complejos, es complejo definir una política de

seguridad que englobe todos los componentes que lo conforman.

- Daños causados por virus, gusanos, etc.

Entre las necesidades de seguridad que presenta el grid están:

- Autenticación: Pueden existir múltiples mecanismos de autenticación.
- Delegación: Las políticas de delegación deben especificarse.
- Single sign-on: Se le debe permitir a un usuario el acceso continuo y correcto por un período de tiempo razonable utilizando una única autenticación.
- Renovación de credenciales: Un trabajo iniciado por un usuario puede tomar más tiempo que el tiempo provisto por la credencial del usuario. En tal caso, el usuario necesita ser notificado con anterioridad de la expiración de las credenciales, o se le debe renovar automáticamente.
- Autorización: Los recursos deben ser utilizados bajo ciertas políticas de autorización. Un proveedor de servicios puede especificar sus propias políticas de autorización.
- Confidencialidad: La confidencialidad de los mecanismos de comunicación de los mensajes o documentos debe ser soportado.
- Integridad de los mensajes: Se debe garantizar que cambios no autorizados de mensajes y documentos deben ser detectados.
- Privacidad: Tanto quien requiere el servicio como el que lo provee deben cumplir las políticas de privacidad.

La manera como se maneja la seguridad en los ambientes grids es utilizando Organizaciones Virtuales (VO) entre una o más organizaciones físicas (o dominios administrativos).

Las VOs :

- Una VO puede tener una estructura jerárquica compleja con grupos y subgrupos. Esta estructura es necesaria para dividir a los usuarios de acuerdo a sus tareas y a las instituciones a las cuales pertenecen. Estos grupos pueden ser vistos de manera independiente.
- Los usuarios pueden pertenecer a varias VOs, al igual que los proveedores de recursos pueden proveer parte de sus recursos a varias VOs.
- Una VO puede ser creada para satisfacer algún requerimiento y eliminada después que este requerimiento es alcanzado.

- Los usuarios se pueden adherir a las VOs o dejarlas.
- Los proveedores de recursos pueden adherirse o dejar las VOs.
- Los proveedores de recursos pueden cambiar dinámicamente las políticas de acceso a sus recursos.
- Los administradores de las VOs pueden manejar a los usuarios de manera dinámica.

Actualmente la seguridad dentro del grid es manejada de la siguiente manera:

Se hace una única autenticación utilizando un certificado digital firmado por una autoridad de certificación reconocida por varios proveedores de recursos, usuarios y VOs. Esto no garantiza el acceso a los recursos, pero provee un único enlace entre el identificador y el usuario,

La delegación, que es el proceso de transferir derechos de los usuarios a tareas o *proxies*, y el *single-sign-on*, una única entrada, se hace a través del uso de certificados *proxies*. Esto permite a los usuarios generar de manera remota los *proxies*, creando una nueva clave privada, utilizando el certificado y la clave original del mismo, y se evita el envío de las claves y la contraseña a través de la red.

La autorización por cada VO concede el acceso a una persona o servicio basándose en el nombre del certificado. Esto se hace mediante listas de usuarios autorizados por cada VO.

Existen varias maneras de realizar la autorización:

- **Pull Model:** Concede los derechos del usuario solo en condiciones específicas, maneja los derechos entre los usuarios y los proveedores de recursos. Ejemplo: *Akenti*
- **Push Model:** Concede los derechos de los usuarios de acuerdo a sus roles. Administra los derechos de manera centralizada. Ejemplo: *CAS*, *PERMIS*, *VOMS*.

El *CAS* y el *VOMS* son las maneras más comunes de realizar la autenticación, las características principales relacionadas se muestran a continuación.

Community Authorization Service (CAS): Permite políticas de autorización de grano fino. Los proveedores de recursos pueden generar políticas para dominios externos. También permite crear políticas para los usuarios locales. Los usuarios que realizan alguna petición obtienen las capacidades de sus *CAS* locales.

Virtual Organization Membership Service (VOMS): Provee información sobre el usuario y sus relaciones con las diferentes *VOs*.

- Permite la utilización de grupos y roles
- Se usa un único *login*: *voms-proxy-init* al comienzo de la sesión.
- Tiempo de expiración: la información de la autorización es válida solo por un período de tiempo definido.
- Compatibilidad: La información relacionada a la *VO* y al certificado del *proxy* del usuario.
- Múltiples *VOs*: El usuario puede identificarse a sí mismo en múltiples *VOs*.
- Seguridad: Todas las comunicaciones cliente servidor son seguras y autenticadas.

La infraestructura de seguridad en el grid (*GSI*) actualmente es provista por el Globus Toolkit. La cual se basa en la tecnología estándar *PKI*

- Protocolo *SSL* para la autenticación y la protección de los mensajes.
- Relaciones de confianza entre autoridades de certificación
- Certificados *X.509* para comprobar la identidad de usuarios, servicios, máquinas.
- Identidad dentro del grid: Un usuario es *mapeado* a identidades locales utilizando el **distinguished name** del certificado del usuario.

7. Conclusiones

Se describió de manera sencilla la historia del grid, una definición, se esbozaron los estándares que se están siguiendo así como los aspectos básicos y una breve descripción de como se maneja la seguridad actualmente, además de introducir el tema de las Organizaciones Virtuales, con el fin de darle al lector una introducción a los grids computacionales.

8. Referencias

- [1] Foster, Ian. What is the grid? A tree point checklist. Argonne National Lab & University of Chicago. <http://www.gridtoday.com/02/0722/100136.html>
- [2] Foster, Ian; Kesselman, Carl. The grid: Blueprint for a New Computing Infrastructure. 1998
- [3] Foster, Ian; Kesselman, Carl; Tuecke, S. The Anatomy of the grid: Enabling Scalable Virtual Organizations.