Segurança em Grades Computacionais

Edson Tessarini Pedroso

Trabalho Final de
Mestrado Profissional

Instituto de Computação
Universidade Estadual de Campinas

Segurança em Grades Computacionais

Edson Tessarini Pedroso

Julho de 2006

Banca Examinadora:

- Prof. Dr. Ricardo Dahab (Orientador)

Depto. de Teoria da Computação, IC / UNICAMP.

- Prof. Dr. Adriano Mauro Cansian

Depto. de Ciência da Computação e Estatística, IBILCE / UNESP.

- Prof. Dr. Luiz Eduardo Buzato

Depto. de Sistemas de Informação, IC / UNICAMP.

- Prof. Dr. Julio Cesar López Hernández (Suplente)

Depto. de Teoria da Computação, IC / UNICAMP.

FICHA CATALOGRÁFICA ELABORADA PELA BIBLIOTECA DO IMECC DA UNICAMP
Bibliotecária: Maria Júlia Milani Rodrigues – CRB8a / 2116

Pedroso, Edson Tessarini

P343s Segurança em Grades Computacionais / Edson Tessarini Pedroso -- Campinas, [S.P. :s.n.], 2006.

Orientador : Ricardo Dahab

Trabalho final (mestrado profissional) - Universidade Estadual de Campinas, Instituto de Computação.

1. Computação em Grade (Sistemas de computador). 2. Infra-estrutura de chaves públicas (Segurança de computador). 3. Engenharia de software. I. Dahab, Ricardo. II. Universidade Estadual de Campinas. Instituto de Computação. III. Título.

(mjmr/imecc)

Título em inglês: Security in Grid Computing.

Palavras-chave em inglês (Keywords): 1. Grid computing (Computer systems). 2. Public key infrastructure (Computer security). 3. Software engineering.

Área de concentração: Engenharia de Software

Titulação: Mestre em Computação

Banca examinadora: Prof. Dr. Ricardo Dahab (IC-UNICAMP)
Prof. Dr. Adriano Mauro Cansian (UNESP-São José do Rio Preto)
Prof. Dr. Luiz Eduardo Buzato (IC-UNICAMP)

Data da defesa: 26/07/2006

Programa de Pós-Graduação: Mestrado Profissional em Engenharia de Computação

Segurança em Grades Computacionais

Este exemplar corresponde à redação final do Trabalho Final devidamente corrigida e defendida por Edson Tessarini Pedroso e aprovada pela Banca Examinadora.

Campinas, 26 de julho de 2006.

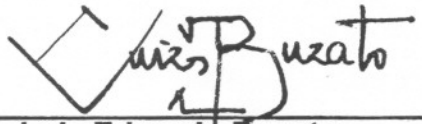
Prof. Dr. Ricardo Dahab
(Orientador)

Trabalho final apresentado ao Instituto de Computação, UNICAMP, como requisito parcial para a obtenção do título de Mestre em Computação na área de Engenharia de Software.

TERMO DE APROVAÇÃO

Trabalho Final Escrito defendido e aprovado em 26 de julho de 2006, pela Banca Examinadora composta pelos Professores Doutores:

Prof. Dr. Adriano Mauro Cansian
IBILCE/ UNESP



Prof. Dr. Luiz Eduardo Buzato
IC - UNICAMP

Prof. Dr. Ricardo Dahab
IC - UNICAMP

Agradecimentos

Agradeço em primeiro lugar a Deus por ter me dado a oportunidade de realizar este trabalho. Sem Ele não podemos nada.

Aos meus pais, Edson e Dalva, por todo o apoio que sempre me deram, me incentivando a nunca desistir. Por toda educação que me deram para a vida, permitindo que eu conquistasse mais este objetivo de maneira honesta e digna através de esforço, trabalho e muita dedicação.

À minha noiva, Cinthya, que dedicou todo seu amor, não me deixando desanimar por maiores que fossem os desafios. Por todos os momentos que esteve ao meu lado me amparando, confortando e até mesmo me auxiliando na busca do melhor. Dedico este trabalho como prova de meu amor.

À minha irmã, Érika, que soube como me amparar e fortalecer quando eu mais precisei. Sempre esteve ao meu lado transmitindo seu entusiasmo e alegria de vida como uma forma de incentivo para superar todos os obstáculos que surgem como provas para o nosso fortalecimento.

Aos meus avós, que mesmo em um outro plano sempre estiveram comigo. Mesmo através de pensamentos me mostraram o melhor caminho a seguir simplesmente por tudo aquilo que foram em vida.

Ao Professor Doutor Ricardo Dahab, pela orientação deste trabalho. Pela paciência e dedicação, sendo além de orientador, professor e amigo com suas palavras de auxílio e colaboração na busca da melhor forma de transmitir o conhecimento.

À Professora Doutora Ariadne M. B. Rizzoni Carvalho, por ter me incentivado a realizar este trabalho e sempre me manter otimista quanto aos desafios. Pela amizade demonstrada nos momentos de dúvidas e o alento proporcionado em ocasiões difíceis.

Aos meus amigos de trabalho da Softway, Eduardo T. Soares Júnior e Gustavo Andretta, pelo que me ofereceram para que eu chegasse até aqui.

Aos meus colegas de mestrado, por tudo o que aprendemos e crescemos juntos.

Aos amigos de coração que sempre entenderam a razão do meu esforço.

Ao Instituto de Computação da Unicamp, seus professores e funcionários pela atenção, amizade, contribuição intelectual e principalmente obrigado pela grande oportunidade.

Obrigado a todos que de uma forma ou outra contribuíram para este trabalho.

Dedicatória

Nesses últimos anos em que passei boas horas dedicadas ao trabalho de mestrado, uma pessoa sempre esteve a meu lado torcendo e me ajudando muito em tudo que precisei para concluir as atividades da melhor forma possível.

Nas horas em que o cansaço batia lá estava ela com seu carinho, amor e conversa agradável tentando ajudar de alguma forma. Muitas vezes não sabia nem como ajudar, mas procurava uma forma nem que fosse para oferecer um cafezinho, um suco, seja lá o que fosse e a que horas fosse.

Essa pessoa sem dúvida era uma das minhas maiores incentivadoras para a realização deste mestrado e sonhava tão intensamente com o dia da conclusão.

Minha mãe é essa pessoa que sempre esteve ao meu lado, que sempre me deu força, amor, carinho e, tudo que se possa imaginar de bom nessa vida. Tenho certeza que sem ela, não teria chegado até aqui.

Infelizmente ela não está mais aqui para ver esse trabalho concluído da forma que ela sempre torceu e sonhou.

Confesso que quando a senhora foi embora pensei em desistir, mas pensando em tudo que me ensinou criei forças para superar as adversidades e caminhar até aqui.

Dedico este trabalho especialmente à senhora, minha mãe, e agradeço por tudo que fez por mim e que sem dúvida continua fazendo. Fique com Deus, com o meu amor e com a certeza de que nunca te esquecerei. Até algum dia.

Resumo

Grade computacional é um conceito que explora as potencialidades das redes de computadores, com o objetivo específico de disponibilizar camadas virtuais que permitem a um usuário ter acesso a aplicações altamente exigentes, bem como aderir a comunidades virtuais de grande escala, com uma grande diversidade de recursos de computação e de repositórios de informações. Grades computacionais são sistemas de suporte à execução de aplicações paralelas que acoplam recursos heterogêneos distribuídos, oferecendo acesso consistente e barato aos recursos, independente de sua posição geográfica. As tecnologias de grades computacionais possibilitam agregar recursos computacionais variados e dispersos, acelerando a execução de vários processos computacionais. Para melhor entendimento das questões de segurança, principal foco deste trabalho, um estudo geral sobre a grade computacional envolvendo assuntos como arquitetura, funcionalidades, aplicações e serviços, foi realizado com o objetivo de identificar e demonstrar a complexidade existente por trás destes cenários. As exigências de segurança são fundamentais a um projeto de grade computacional. Os componentes de segurança devem fornecer os mecanismos corretos para uma comunicação segura em um ambiente de grade. Sem estes mecanismos, as informações processadas dentro da grade tornam-se vulneráveis. O propósito deste trabalho é a realização de um levantamento sobre as questões de segurança em grade computacional, identificando problemas existentes, soluções, arquiteturas, ferramentas e técnicas aplicadas. Com base nessas informações é possível entender como funcionam os mecanismos de segurança em grade, identificando o que já existe de efetivo e quais as necessidades para que a maturidade e popularidade neste ambiente possam ocorrer.

Abstract

Grid computing is a concept that exploits the power of computer networks, with the specific aim of making virtual layers available that allow users to have access to highly demanding applications, as well as to adhere to large scale, highly diverse, virtual communities. Grid computing provides support for the execution of parallel applications, grouping together distributed heterogeneous resources, offering consistent and inexpensive access to them, independently of their geographical location. The technology of grid computing allow the gathering of different and disperse computer resources, accelerating the execution of various computer processes. To better understand security issues in grids, the principal focus of this work, a general study of grid computing including architecture, functionalities, applications and services was done, with the goal of identifying and demonstrating the existent complexity behind this scenery. Security is fundamental to a grid project. Security components must supply the correct mechanisms for secure communication, without which processed information inside a grid becomes vulnerable. The purpose of this work is to survey security issues in grid computing, identifying existent problems, solutions, architectures, tools and techniques. Based on this information it is possible to understand how security mechanisms in grids work, identifying those mechanisms already in place and working, as well as what is needed for the full development of grid computing.

Sumário

Agradecimentos xi
Dedicatória xiii
Resumo xv
Abstract xvii
Lista de Figuras xxi
Lista de Tabelas xxiii
Introdução 1
1.1 Motivação 1
1.2 Objetivos da Dissertação 2
1.3 Contribuições 3
1.4 Organização do Documento 3
Grades Computacionais 5
2.1 Conceituação 5
2.1.1 Características Gerais 8
2.1.2 Organizações Virtuais 9
2.1.3 Intranets, Clusters e Grades 10
2.2 Aplicações 14
2.3 Arquitetura 21
2.4 Sistemas para Provimento de Aplicações em Grades 28
2.4.1 Globus 29
2.4.2 Legion 31
2.4.3 Iniciativas de Grades Nacionais 35
2.5 Segurança 36
2.6 Perspectivas 37
2.7 Resumo 38
Segurança em Grades 41
3.1 Problemas clássicos em ambientes distribuídos 42
3.2 Problemas Específicos em Grades 44
3.2.1 Cenário 1: Computação distribuída em alta escala 46
3.2.2 Cenário 2: Ambientes Filantrópicos 55
3.3 Propostas de Soluções de Segurança 57
3.3.1 Cenário 1: Computação distribuída em alta escala 57
3.3.2 Cenário 2: Ambientes Filantrópicos 67
3.3.3 Componentes e aspectos adicionais de segurança em grades 68
3.3.4 Políticas e Procedimentos de Segurança 71
3.4 Resumo 72
Arquitetura Aberta para Serviços na Grade 73
4.1 Desafios de Segurança em um Ambiente de Grade 73
4.1.1 Desafio de Integração 73
4.1.2 Desafio da Interoperabilidade 74
4.1.3 Desafio das Relações de Confiança 74
4.1.4 Inter-Dependência entre os desafios de segurança 75
4.2 Requisitos de Segurança para Serviços de Grade 76

4.3 Modelo de Segurança OGSA para Serviços de Grade 77
4.3.1 Invocação de Segurança 78
4.3.2 Componentes de Segurança 78
4.3.3 Binding de Segurança 80
4.3.4 Representação e Trocas de Políticas 81
4.3.5 Associação de Segurança 82
4.3.6 Mapeamento de Identidades e Credenciais 82
4.3.7 Autorização 83
4.3.8 Privacidade 84
4.3.9 Confiança 84
4.3.10 Gerenciamento de Segurança 85
4.4 Integração com Padrões de Segurança 86
4.5 Segurança como Serviço 88
4.6 Exemplo de Uso 89
4.6.1 Típicos Negócios em Serviços de Grade 89
4.6.2 Cenário envolvendo Intermediários 92
4.7 Resumo 93
Emprego de Infra-estruturas de Chaves Públicas (ICP) em Grades 95
5.1 Grades e Certificação Digital 95
5.1.1 Autoridade Certificadora (AC) 95
5.1.2 A chave privada das ACs 96
5.1.3 Certificação Cruzada da AC 97
5.1.4 Certificados Digitais 97
5.1.5 Autenticação e autorização 100
5.1.6 Autenticação Mútua 101
5.1.7 SSL Handshake 101
5.1.8 Infra-estruturas de Chaves Públicas em Grades 102
5.1.9 Vulnerabilidades da ICP 103
5.2 Confiança 104
5.2.1 Definição de Confiança e Reputação 105
5.2.2 Modelo de Confiança para Grades Computacionais 105
5.3 Delegação 109
5.3.1 Criação do proxy 109
5.3.2 Ação do procurador (proxy) 110
5.4 Globus 111
5.4.1 Características do GSI 112
5.5 Grade das Américas - TAGPMA 113
5.6 Legion 114
5.7 Resumo 115
Conclusões 117
Glossário de Siglas 123
Referências Bibliográficas 125

Lista de Figuras

1.1 Idéia base de grade computacional..... 2
1.2 Grade computacional..... 7
2.2 Analogia com a rede elétrica..... 13
2.3 SMP…….…….. 21
2.4 MPP…….…….. 22
2.5 NOW.....…….... 22
2.6 GRID..... 23
2.7 Arquitetura..... 25
3.1 Exemplo de computação distribuída em larga escala..... 47
4.1 Dependências entre os desafios de segurança em um ambiente de grade..... 78
4.2 Componentes do modelo de segurança da grade..... 81
4.3 Camadas de tecnologias para a arquitetura de segurança da grade..... 89
4.4 Prestação de serviços dentro de uma organização virtual..... 92
4.5 Prestação de serviços através das organizações virtuais..... 93
4.6 Prestação de serviços com o emprego de intermediários..... 94
5.1 Componentes de gerenciamento de confiança da grade..... 101
5.2 Certificado Digital..... 107

Lista de Tabelas

2.1 Computação Filantrópica x Ambiente Empresarial..... 18
2.2 Características das plataformas de execução..... 23
2.3 Camadas de tecnologias de grades computacionais..... 26
5.1 Exemplo de níveis de confiança..... 99
5.2 Valores previstos de suporte à confiança……..... 100

Capítulo 1

Introdução

Este capítulo apresenta o trabalho. A seção 1.1 mostra a motivação que levou ao desenvolvimento desta dissertação. A seção 1.2 descreve os objetivos. A seção 1.3 cita as contribuições oferecidas por este texto. A seção 1.4 explica como o documento está organizado.

1.1 Motivação

O conceito de grade computacional foi concebido no início da década de 90 no laboratório Argonne (Argonne National Laboratory dos Estados Unidos) por Ian Foster e Carl Kesselman. Tal conceito foi criado e justificado com base na idéia de usar de forma ótima os melhores e inúmeros recursos computacionais dispersos em várias instituições.

Existem inúmeros computadores em estado ocioso e as idéias propostas pelos sistemas de grade computacional são subsídios necessários para utilizar este tempo ocioso dos recursos que em geral estariam sendo subtilizados. Vários benefícios serão alcançados, como uma empresa em um país como o Brasil que poderá utilizar o processamento de computadores na madrugada de Tóquio, onde estes computadores poderiam estar ociosos ou até mesmo desligados.

Inicialmente a montagem das grades será feita entre partes com interesses similares interligando laboratórios, empresas e universidades e que tenham interesses semelhantes. Com sua expansão, pode-se chegar, no final, em algo como a formação de uma grade global, uma rede mundial distribuída de colaborações entre seus participantes, capaz de prover recursos talvez inatingíveis por um *cluster* isolado [BAIRD, 2002].

A popularização do sistema em grade irá atingir os usuários finais, que também farão parte desta comunidade de colaboração, beneficiando-se dos recursos desta grande rede, utilizando o poderio de processamento, compartilhamento de bases de dados e recursos diversos, sendo que, o usuário também fará sua parte, compartilhando os recursos de seu computador para benefício da rede como uma troca de favores.

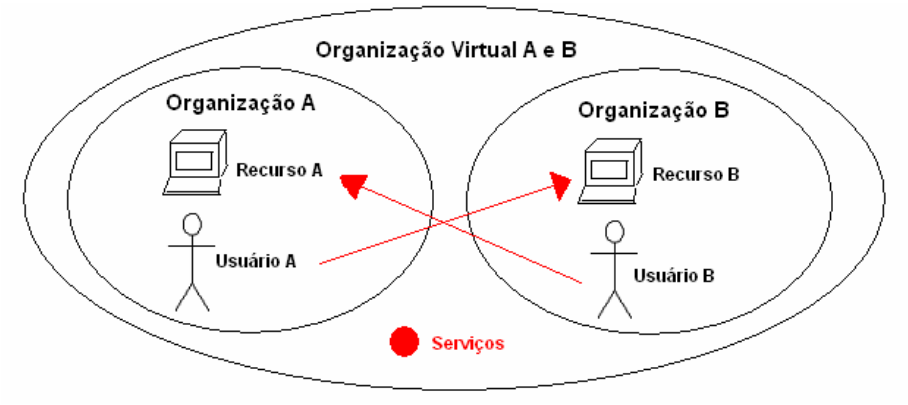


Figura 1.1: Idéia base de grade computacional.

A figura 1.1 mostra a idéia base da proposta de grade computacional em que uma organização que chamamos de A envia serviços para serem executados em recursos ociosos da organização B, sendo que o mesmo pode ocorrer de forma inversa. Essa cooperação através da grade cria o que chamaremos de organização virtual. Para que uma organização virtual possa existir, aspectos como confiabilidade, integridade, privacidade, devem ser colocados em primeiro plano.

A computação em grade pode trazer muitas vantagens, mas para que se torne realmente eficaz muitos fatores devem ser levados em consideração. O ambiente de grade computacional é muito heterogêneo e disperso existindo uma variedade de políticas de acesso, mecanismos de segurança, enorme volume de processamento e uma diversidade de tipos de usuários e aplicações.

Podemos ver que para atingirmos o cenário ideal de grade computacional muitos cuidados devem ser tomados e um dos principais deles é a questão de segurança que é a grande motivação para este trabalho.

1.2 Objetivos da Dissertação

Para que a computação em grade ocorra com sucesso, empresas, pessoas e todos aqueles que tiverem interesse nessa nova infra-estrutura computacional precisam ter a garantia de que mesmo sabendo que em seus recursos outras informações e aplicações possam ser processadas, estes não causarão danos aos recursos e informações de grande relevância aos seus interesses.

Além disso, todos aqueles que precisam ter seus dados processados em outros locais, também precisam ter a garantia de que essas informações serão processadas corretamente não sendo danificadas de alguma forma e tendo o resultado do processamento alterado.

O objetivo da dissertação é demonstrar em que estágio as questões de segurança em grades computacionais se encontram, explorando problemas, soluções e mecanismos empregados para que o conceito idealizado possa ser alcançado.

1.3 Contribuições

Para muitos pesquisadores e cientistas, a computação em grade é vista como um grande avanço tecnológico quando tratamos de processamento de dados. Os benefícios a serem alcançados pela evolução das grades computacionais podem se refletir em diferentes meios, acadêmicos, empresariais, dentre outros.

Os ganhos obtidos através das grades computacionais serão conquistados com o desenvolvimento de aplicações seguras. Portanto, esta dissertação pode auxiliar programadores, pesquisadores e todos aqueles interessados em identificar possíveis problemas, soluções e mecanismos de segurança existentes em um ambiente de grade computacional.

Assim, são apresentadas as seguintes contribuições:

- Elucidação dos conceitos de grade computacional.
- Levantamento de problemas e propostas de segurança.
- Demonstração do funcionamento da grade baseado em arquiteturas de padrões abertos para troca de serviço.
- O emprego de infra-estruturas de chaves públicas em grades computacionais.

1.4 Organização do Documento

O texto está organizado da seguinte forma: O capítulo 2 apresenta uma conceituação de grade computacional, tratando de questões como aplicações, arquitetura, sistemas para provimento de aplicações em grade, segurança, perspectivas e outros conceitos relacionados. No

capítulo 3 são levantados os problemas de segurança, os quais foram separados em duas classes: os problemas comuns, iguais aos encontrados em outras aplicações de processamento distribuído, e os problemas específicos de grades computacionais. Também serão apresentadas propostas de segurança para os problemas citados. O capítulo 4 apresenta uma arquitetura aberta para serviços em grade. O capítulo 5 mostra o emprego de infra-estruturas de chaves públicas e grades computacionais. O capítulo 6 conclui a dissertação.

Capítulo 2

Grades Computacionais

Este capítulo introduz os conceitos básicos ao entendimento desta dissertação. A seção 2.1 apresenta a conceituação de grades computacionais e faz uma analogia de grade computacional com redes de energia elétrica. A seção 2.2 descreve algumas aplicações de grades computacionais. A seção 2.3 faz um histórico e discute a arquitetura de grades. A seção 2.4 descreve os sistemas para provimento de aplicações em grades. A seção 2.5 faz uma breve introdução sobre segurança e por fim a seção 2.6 apresenta as perspectivas para grades computacionais.

2.1 Conceituação

Segundo Ian Baird [BAIRD, 2002], computação em grade é uma coleção de recursos heterogêneos e distribuídos possibilitando sua utilização comunitária em aplicações de larga escala.

Sistemas em grade integram e coordenam recursos e usuários que vivem no interior de diferentes domínios ou até diferentes unidades administrativas num mesmo domínio com diferentes políticas de segurança.

Grades são construídas com protocolos e interfaces de uso geral destinado a propósitos específicos como a autenticação, autorização, localização e acesso a recursos. É importante que estes protocolos e interfaces sejam padronizados e abertos. De outra forma, estaríamos lidando com um sistema de aplicação específica.

Sistemas em grade permitem que os recursos de máquinas-cliente sejam utilizados de forma coordenada para fornecer diferentes níveis de qualidade de serviço, como o tempo de resposta, *throughput* (quantidade de dados transmitidos em uma unidade de tempo), disponibilidade, segurança e a alocação de diferentes tipos de recursos para se adequar em complexas exigências dos usuários.

De acordo com Ian Foster [FOSTER, 2005], existem plataformas para execução de aplicações paralelas que não são grades. De maneira geral, podemos dizer que grades são mais distribuídas, diversas e complexas que outras plataformas. Alguns aspectos que evidenciam esta distribuição, diversidade e complexidade são:

- Heterogeneidade (nos componentes da grade),
- Alta dispersão geográfica (grades podem ter escala mundial),
- Compartilhamento (no sentido de que a grade não pode ser dedicado a uma aplicação),
- Múltiplos domínios administrativos (grades podem congregar recursos de várias instituições),
- Controle distribuído (tipicamente não há uma única entidade que tenha poder sobre toda a grade).

Em grades computacionais chamamos de **organização virtual** quando seus participantes desejam compartilhar recursos para poder concluir uma tarefa comum. Além disso, o compartilhamento vai além da simples troca de documentos, mas pode envolver acesso direto a software remoto, computadores, dados, sensores e outros recursos.

De acordo com Wolfgang Gentzsch [GENTZSCH, 2002], por serem uma nova plataforma com alta heterogeneidade, compartilhamento e complexidade, as grades tendem a apresentar maiores dificuldades para a execução de aplicações paralelas que plataformas tradicionais. Em geral, grades são mais propícias à execução de aplicações onde não haja muita comunicação entre tarefas pelo fato de utilizarem a Internet como o meio principal de interconexão. As melhores aplicações em grades são aquelas em que as tarefas podem ser divididas em tarefas menores independentes umas das outras e depois agrupadas para a obtenção do resultado da tarefa principal.

Apesar dessa maior dificuldade, as grades permitem a interconexão de uma vasta gama de recursos criando uma espécie de ambiente virtual, capaz de conectar pessoas, computadores, instrumentos científicos como telescópios e bancos de dados proporcionando uma nova forma de colaboração entre comunidades. A Figura 2.1 exemplifica uma grade computacional.

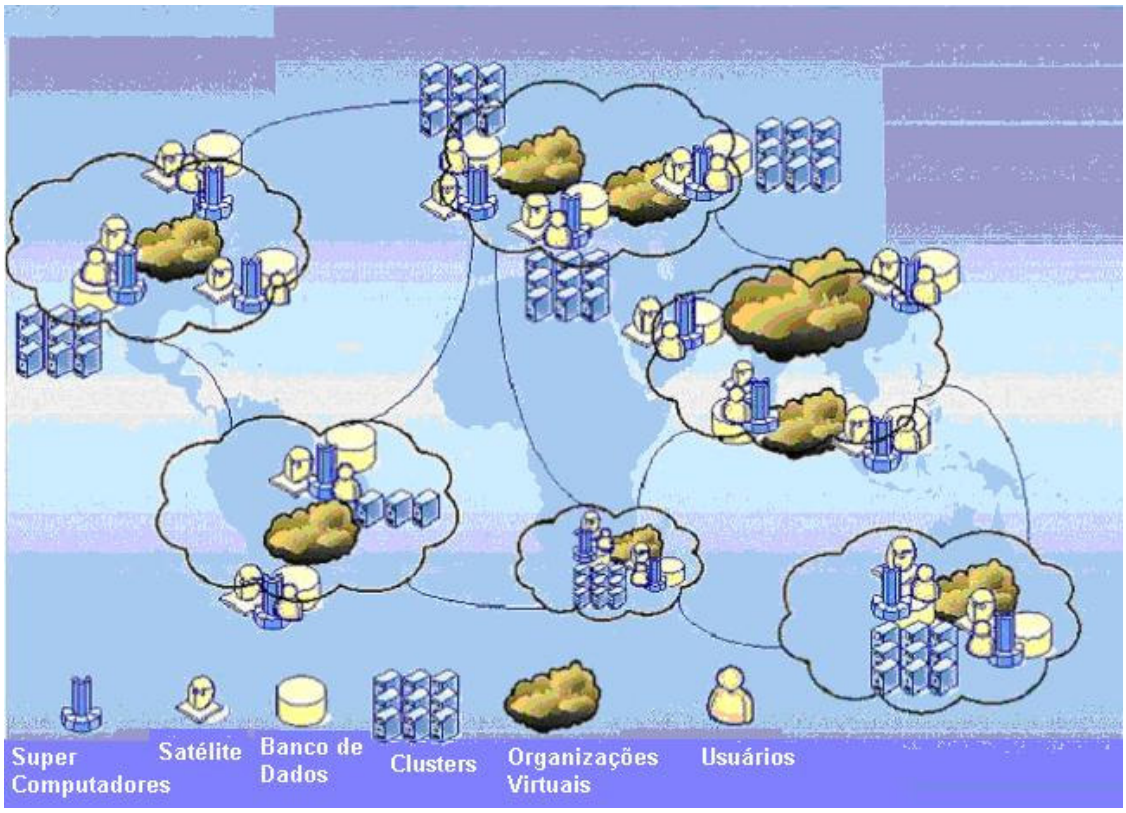


Figura 2.1: Grade Computacional [GRID_IMAGEM, 2005].

Uma grade poderia ser útil, por exemplo, para um grupo de cientistas espalhados no mundo querendo realizar pesquisas a respeito do Universo. Para tal, poderia ser utilizado um supertelescópio localizado em um determinado ponto do planeta capaz de gerar uma gigantesca quantidade de dados a serem analisados. Adquirir um computador capaz de analisá-los de forma eficiente pode ser extremamente caro, ou inviável. Uma solução alternativa seria o particionamento e distribuição dos dados para serem processados em paralelo por recursos computacionais de uma grade formada para essa finalidade. Depois de obtidos os resultados, estes poderiam ser disponibilizados na grade aos cientistas utilizando poderosas ferramentas de visualização. Dessa forma, os resultados poderiam ser avaliados e discutidos pelos cientistas em tempo real.

A discussão feita até aqui já possibilita uma definição mais precisa de uma grade, porém, alguns conceitos como **organizações virtuais**, além de **clusters** e **intranets** precisam ser melhores explorados.

2.1.1 Características Gerais

A idéia básica da computação em grade é a computação como uma utilidade sempre disponível, em qualquer ponto, na medida da demanda e com segurança. Do ponto de vista do usuário da grade, não importa onde estejam os dados e quais servidores atendem às suas requisições, e sim ter sua demanda atendida a tempo. Por outro lado, do ponto de vista dos profissionais de TI, o conceito de computação em grade se traduz em virtualização e aprovisionamento de recursos computacionais. A virtualização consiste em estabelecer abstrações que representem os recursos, de modo a permitir a sua utilização de forma flexível. O aprovisionamento consiste na capacidade de gerenciar e disponibilizar os recursos dinamicamente, com segurança, para atender aos requisitos das demandas de forma coordenada. Juntos, a virtualização e o aprovisionamento conferem a grade computacional o potencial de melhor aproveitamento dos recursos computacionais [GUEDES, 2005].

A computação em grade permite que as organizações agreguem recursos com toda a infra-estrutura dos departamentos de TI, não importando localização global. Isso elimina situações onde um site esteja sendo exigido com sua capacidade máxima, enquanto outros tenham ciclos disponíveis.

Organizações podem melhorar dramaticamente sua qualidade e velocidade de produtos e serviços disponibilizados, enquanto os custos de TI são reduzidos por habilitar a colaboração transparente dos recursos compartilhados.

Permite que empresas acessem e compartilhem bases de dados de forma remota. Isto é essencialmente benéfico para as ciências da saúde ou comunidades de pesquisa, onde volumes grandiosos de dados são gerados e analisados durante todo o dia.

Possibilita a larga dispersão das organizações para que facilmente possam colaborar em projetos pela criação da habilidade do compartilhamento de tudo, desde aplicações a dados até projetos de engenharia.

Pode aproveitar os ciclos de processamento ociosos disponíveis dos PCs de mesa que se encontram em várias localizações em múltiplas faixas de tempo.

Apesar dos computadores estarem mais rápidos do que nunca, mais poder computacional é necessário para solucionar grandes problemas. A grade computacional pode ajudar na solução desses problemas disponibilizando uma grande quantidade de recursos a um custo razoável. A

grade pode conectar recursos geograficamente distribuídos como clusters, supercomputadores, instrumentos científicos, repositórios de dados, dispositivos de visualização e pessoas.

2.1.2 Organizações Virtuais

De acordo com Cezar Taurion [TAURION, 2004], uma rede é baseada em competências especializadas, onde cada membro da equipe, pessoa física ou empresa contribui com seus conhecimentos específicos para a melhoria do todo. Neste cenário, as empresas começam a se estruturar em organizações virtuais, com composições diferentes a cada projeto ou iniciativa de negócio.

As características das organizações virtuais também demonstram profundas mudanças em relação às organizações tradicionais. As hierarquias rígidas tendem a desaparecer, pois é necessário o cruzamento de fronteiras organizacionais, para que haja a cooperação de múltiplos especialistas, pertencentes a diversas áreas. As empresas devem se complementar umas às outras com suas competências essenciais. Em cada etapa do ciclo de vida do projeto ou serviço, a composição da rede deve variar de acordo com a necessidade e a competência de cada um dos componentes. As características básicas de uma empresa no contexto das organizações virtuais são:

- Especialização nas competências essenciais;
- cooperação pró-ativa, tomando decisões com base nas suas prioridades, mas levando em consideração os parceiros da rede colaborativa;
- negócios baseados em oportunidades, explorando a velocidade e agilidade típica das redes de organizações virtuais;
- organização virtual por excelência, adotando o uso de recursos de fora da empresa. Entre os conceitos podemos citar escritórios virtuais, tele-trabalho, tele-cooperação e assim por diante;
- capacidade de integração, reunindo-se rapidamente a outras empresas para responder à flutuação da demanda.

Como exemplo de organizações virtuais, podemos citar: ASP (*Application Service Providers*), SSP (*Storage Service Providers*), *Cycle Providers*, etc.

As *Application Service Providers* [ASP, 2000] são empresas que hospedam, gerenciam e alugam software de forma compartilhada para vários clientes. Quando uma empresa contrata os serviços de um ASP, ela está terceirizando suas necessidades de TI e transferindo os aplicativos mais difíceis de gerenciar para quem possa fazê-lo com eficiência e qualidade. O acesso ao software é feito através da Internet ou de uma rede própria.

As *Storage Service Providers* [SSP, 2005] são empresas que disponibilizam espaço e serviços para o recebimento e armazenamento de dados. Oferecem estrutura redundante e extremamente segura. São conhecidas também por data centers. E assim, analogamente, existem empresas que provêm ciclos de processamento (*Cycle Providers*), soluções de negócios entre empresas, ou seja, a integração das diversas empresas que compõem um projeto (*Business Service Providers*).

2.1.3 Intranets, Clusters e Grades

Intranet

De acordo com a enciclopédia web Wikipédia [INTRANET, 2005], a Intranet é uma rede de computadores privativa que utiliza as mesmas tecnologias que são utilizadas na Internet. O protocolo de transmissão de dados de uma intranet é o TCP/IP e sobre ele podemos encontrar vários tipos de serviços de rede comuns na Internet, como por exemplo o e-mail, chat, grupo de notícias, HTTP, FTP entre outros.

Uma Intranet pode ou não estar conectada à Internet ou a outras redes. É bastante comum uma Intranet de uma empresa ter acesso à Internet e permitir que seus usuários usem os serviços da mesma, porém, nesse caso, é aconselhável a implantação de serviços e dispositivos de segurança como, por exemplo, um firewall para fazer o bloqueio do trânsito de dados indevidos entre a rede pública e a rede privativa.

Quando uma intranet tem acesso a outra intranet, caso comum entre filiais de uma empresa ou entre empresas que trabalham em parceria, podemos chamar a junção das duas ou mais redes de extranet. Algumas empresas comumente chamam de extranet a área de sua intranet

que oferece serviços para a rede pública Internet. Uma tecnologia que tem se difundido muito na área de tecnologia da informação para a criação de extranets aproveitando-se da infra-estrutura da Internet é a VPN.

O uso de redes do tipo intranet nas empresas se difundiu e consolidou em meados dos anos 90 juntamente com a popularização da Internet.

Cluster

Cluster é o nome dado a um sistema montado com mais de um computador, cujo objetivo é fazer com que todo o processamento de aplicações seja distribuído aos computadores, de forma transparente. Com isso, é possível realizar processamentos até a pouco disponíveis somente em computadores de alto desempenho.

Os computadores de um cluster, chamados de nós, devem ser interconectados, de maneira a formarem uma rede. Essa rede precisa ser criada de uma forma que permita o acréscimo ou a retirada de um nó, sem interromper o funcionamento do cluster. O sistema operacional usado nos computadores deve ser de um mesmo tipo. Isso porque existem particularidades em cada sistema operacional que poderiam impedir o funcionamento do cluster.

Independentemente do sistema operacional usado, é preciso usar um software que permita a montagem do cluster em si. Esse software vai ser responsável, entre outras coisas, pela distribuição do processamento. Esse é um ponto crucial na montagem de um cluster e é preciso que o software trabalhe de forma que erros e defeitos sejam detectados, oferecendo meios de providenciar reparos, mas sem interromper as atividades do cluster.

Grades

Um dos objetivos originais da Internet era a interligação de diferentes ambientes computacionais e geograficamente dispersos. Os web sites sempre foram interoperáveis em relação usuário-site, por meio de aplicações criadas neste contexto, em que o usuário dispõe de um menu de serviços fechados, caracterizando a Intranet. No ambiente de grade é o inverso, o usuário tem de submeter suas aplicações para serem resolvidas dentro do ambiente por ele montado.

Um ambiente de *cluster* constitui em um sistema formado por hardware e software conectados em um local apenas, servindo a usuários que estão trabalho somente em um projeto,

usado exclusivamente para resolver os problemas computacionais de uma determinada organização. Os recursos são gerenciados por uma entidade central, e os computadores agem como se fosse um único dispositivo.

Por outro lado, uma grade presta serviços de uma forma geograficamente distribuída. Nas configurações em grade, cada organização virtual faz o gerenciamento de seus recursos não tendo a visão de uma imagem única do sistema, ou seja, o usuário tem consciência dos diversos serviços disponíveis e que deverá requisitá-los para sua utilização, portanto, as grades são mais heterogêneos, complexos e distribuídos. Por isso, em grade, existe uma menor dependência de localização e uma maior dependência no canal de comunicação, pois as trocas de mensagens são constantes. O que acontece muitas vezes é a inclusão de clusters em um sistema de grade [BONBONATO, 2004].

Segundo Ian Baird [BAIRD, 2002], em se tratando de grades computacionais, as aplicações leves são as ideais, pois estas requisitam relativamente menos das redes. Mesmo tendo máquinas com uma alta conexão, estas redes com baixo fluxo de dados constituem uma espécie de gargalo ao requisito fundamental para aplicações pesadas. A multiplicidade das velocidades das diversas redes implica em alguns pontos de gargalo, e que compromete a performance da grade.

Com intuito de clarear ainda mais a definição de grade, já que este conceito está deixando o obscuro mundo acadêmico e se tornando mais popular, pois já é possível ler a respeito de *Compute Grids*, *Data Grids*, *Science Grids*, *Access Grids*, *Knowledge Grids*, *Bio Grids*, *Sensor Grids*, *Cluster Grids*, *Campus Grids*, *Tera Grids*, e comumente *Grids*, Ian Foster [FOSTER, 2005] sugeriu um checklist englobando a essência de outras definições:

- Recursos coordenados que não se sujeitam a um controle centralizado.
- Utilizar padrões abertos e com interfaces e protocolos de propósito geral.
- Prover o mínimo em qualidade de serviços.

Wolfgang Gentzsch [GENTZSCH, 2002] prefere uma definição simples que diga o que é uma grade e o que ela faz, da seguinte forma:

"Uma grade é uma infra-estrutura de hardware e software que oferece dependência, consistência e facilidade de acesso a recursos que possibilitem o compartilhamento de tais

recursos computacionais, computação utilitária, computação automática, colaboração em um grupo de organizações virtuais (VOs) e processamento de dados distribuído, dentre outros".

O estado da computação atualmente é bem parecido com o estado da eletricidade no início do século passado. Dispositivos que dependiam de energia elétrica eram desenvolvidos, mas o uso dos mesmos era dificultado porque os usuários tinham que ter seus próprios geradores. O que de fato foi revolucionário na época não foi a eletricidade. O grande avanço foi o desenvolvimento da rede elétrica e das tecnologias de transmissão e distribuição associadas. Este desenvolvimento tornou possível o fornecimento de um serviço confiável, de baixo custo e que fosse de fácil acesso aos usuários. Isso contribuiu para o surgimento de novas indústrias e o conseqüente desenvolvimento de novos dispositivos que utilizavam energia elétrica. A Figura 2.2 exemplifica uma Rede Elétrica que faz analogia com a grade computacional [EGEE, 2005].

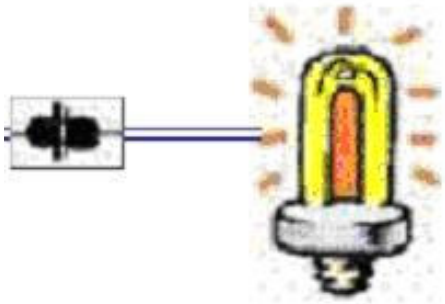


Figura 2.2: Analogia com a rede elétrica.

O termo grade computacional vem exatamente da analogia feita à rede elétrica (*Electric Power Grid*) e representa a infra-estrutura que permitirá avanços na computação, de um modo geral, como ocorreu com a rede elétrica. Essa infra-estrutura é composta por hardware que é necessário para fazer a interconexão dos recursos e por software específico para monitorar e controlar o ambiente.

A visão original estabelece uma metáfora entre a Rede Elétrica (*The Electric Grid*) e a Grade Computacional (*The Computational Grid*). A Rede Elétrica disponibiliza energia elétrica sob demanda e escondem do usuário detalhes como a origem da energia e a complexidade da malha de transmissão e distribuição. Ou seja, se temos um equipamento elétrico, simplesmente o conectamos na tomada para que ele receba energia.

A analogia é baseada em algumas das propriedades das redes de distribuição elétricas tais como:

- Não é necessário conhecer os detalhes da rede elétrica para poder usar a eletricidade.
- A rede elétrica está sempre disponível, mesmo quando existe um problema a redundância da rede permite que o serviço continue disponível.
- A eletricidade está disponível em todo lugar, bastando apenas ligar uma tomada para poder usá-la.
- A rede elétrica é um serviço em que se paga o que se consome.

O paradigma da computação em grade possui propriedades similares às da rede elétrica. Tem por objetivo a integração transparente de recursos de computação que podem pertencer a organizações independentes, escondendo as suas especificidades e apresentando uma interface homogênea aos utilizadores. Desta maneira, grandes infra-estruturas de computação podem ser criadas a partir de recursos dispersos, que surgem aos utilizadores como um único sistema.

A grade computacional, portanto, seria uma rede na qual o individuo se conecta para obter poder computacional (ciclos, armazenamento, software, periféricos, etc) [EGEE, 2005].

2.2 Aplicações

Com base na infra-estrutura de grades computacionais algumas aplicações são projetadas para resolver problemas complexos. Alguns exemplos dessas aplicações seguem abaixo:

Portais Científicos - Os portais buscam métodos para resolver problemas complexos de uma maneira mais fácil através da invocação remota de pacotes de software através de sites web dos portais. Esses pacotes podem ser executados remotamente em computadores de uma grade, por exemplo. Tais portais estão sendo desenvolvidos para auxiliar em ramos como o da biologia, física, computação, dentre outros. O GridPort, por exemplo, é um projeto do *San Diego Supercomputer Center* (SDSC) que reúne uma coleção de serviços, scripts e ferramentas permitindo que desenvolvedores conectem interfaces Web a uma grade computacional. O GridPort possui arquitetura aberta e foi desenvolvido de forma a ser flexível e capaz de usar outros serviços e tecnologias de grades computacionais [GRIDPORT, 2003].

Computação Distribuída - Matemáticos dos EUA e da Itália reuniram uma série de computadores para resolver um problema de otimização chamado Nug30. Este problema é do tipo NP-completo e o tempo necessário para resolvê-lo cresce exponencialmente com o tamanho do problema. O Nug30 requer que 30 unidades sejam atribuídas para 30 locais. Essa atribuição deve ser realizada de uma forma que seu custo, que é a distância entre os lugares e o fluxo entre as unidades, seja mínimo. Para resolver o Nug30, um computador isolado poderia levar muito tempo. Com o avanço nas redes de computadores e tecnologias de grades computacionais é possível resolver problemas cada vez maiores e mais complexos, como o Nug30, através do agrupamento de recursos computacionais [NUG30, 2003].

Análise de Dados em Larga Escala - A análise de petabytes de dados gerados por experimentos científicos vai necessitar da união de milhares de processadores e milhares de terabytes de espaço em disco para guardar os resultados parciais. Muitos problemas científicos interessantes requerem a análise de grandes quantidades de dados. Para isso, o paralelismo natural inerente em muitos procedimentos de análise de dados torna possível o uso de recursos distribuídos de forma eficiente. Em projetos dessa grandeza, várias instituições devem estar envolvidas e elas podem prover os recursos necessários para esse tipo de experimento. Essas comunidades, além de compartilhar computadores e dispositivos de armazenamento, podem compartilhar procedimentos de análise e os próprios resultados. Estações de trabalho podem ser conectadas através de redes de alta velocidade para formar um ambiente com poder computacional considerável [GRIPHYN, 2003].

Aplicações científicas que demandam alto desempenho foram os impulsionadores de grade computacional. O uso de computadores é crescente no meio científico. Uma vez que um determinado problema científico tenha sido determinado, necessita muitas vezes de experimentos, que busquem simular as situações para a busca de resultados para resolução do problema. Porém, as simulações em geral são complexas, demandando mais e mais recursos computacionais.

Algumas necessidades de processamento encontram limites impostos pelo desenvolvimento das tecnologias que possibilitem um aumento significativo no poder computacional de um único supercomputador. A idéia de paralelizar as aplicações, dividindo os

processamentos em pedaços menores que possam ser executados simultaneamente em diversos computadores, foi naturalmente aceita no meio científico.

No ambiente acadêmico, o uso de grades traz como retorno uma maior integração entre as universidades, em torno de projetos de pesquisa. Aplicações de pesquisa, muitas vezes redundantes com outras universidades, estarão disponíveis a todas pela grade computacional.

Instrumentação Contínua - Instrumentos científicos como telescópios, microscópios eletrônicos, dentre outros geram fluxos de dados brutos que são arquivados para uma posterior análise. Uma análise de dados com características próximas a uma análise em tempo real pode aumentar a funcionalidade de um instrumento substancialmente. Como exemplo pode ser citado um astrônomo estudando labaredas solares com um radiotelescópio. Os algoritmos utilizados para processar os dados e detectar as labaredas demandam muito poder computacional. Executar esses algoritmos continuamente seria ineficiente para o estudo dessas labaredas porque elas são breves e esporádicas. Se o astrônomo puder solicitar recursos computacionais em larga escala e tê-los disponíveis sob-demanda, ele pode usar técnicas de detecção automática para ampliar as labaredas solares assim que elas ocorrerem. Além disso, o acompanhamento em tempo real do fenômeno pode possibilitar que o pesquisador tome ações dependendo do que está sendo observado [IVDGL, 2003].

Trabalho Colaborativo - A resolução de problemas que contam com a colaboração de várias pessoas e análise de dados, por exemplo, são aplicações de grades computacionais de extrema importância. Um físico, por exemplo, pode realizar uma simulação que gere grande volume de dados e pode querer que colegas espalhados pelo mundo visualizem os resultados da mesma forma e no mesmo momento, assim o grupo pode discutir os resultados em tempo real. Aplicações apropriadas para grades computacionais freqüentemente vão conter alguns aspectos desse e de outros cenários. O grande atrativo dessa idéia é a possibilidade de alocação de uma enorme quantidade de recursos para executar aplicações paralelas a um custo bastante inferior das soluções tradicionais, como as baseadas em supercomputadores paralelos. Muitas aplicações já estão sendo exploradas, elevando o conceito de grades computacionais [EUDTGRID, 2003].

Computação Filantrópica - Computação Filantrópica possui a idéia de voluntariado, onde um usuário toma a decisão deliberada de ceder ciclos ociosos de seu PC para contribuir com uma determinada organização a executar uma tarefa.

Após o usuário se cadastrar como voluntário no site específico, um pequeno programa é transferido para o seu PC. Este programa é o responsável pela comunicação via internet com o servidor central, bem como pela utilização dos ciclos ociosos do PC para execução da tarefa computacional solicitada.

A sua característica principal é que demanda pouca interação com o servidor central, resumindo-se a baixar novos dados ou devolver dados já processados. A maior demanda é pelos ciclos de processador da máquina.

A aplicação também não deve interferir com a utilização diária do PC e apenas consome ciclos de processador quando o computador estiver inativo. De maneira geral ela opera como um *screen saver*.

Um projeto de grade para computação filantrópica deve se preocupar com alguns aspectos fundamentais [TAURION, 2004]:

a) Disponibilidade dos recursos: Não se pode estimar com precisão quando os resultados serão concluídos.
b) Comunicação e segurança: Os dados e programas cliente devem ser protegidos contra alterações indevidas e não autorizadas.
c) Ambientes heterogêneos: Os programas cliente devem rodar em qualquer máquina, com configurações as mais variadas possíveis.

Desde que o conceito de grade foi proposto, na década de 90, vários projetos vêm sendo realizados com tecnologia de computação em grade. Um projeto que demonstra o potencial da tecnologia de grade é o *SETI@home*. Nesse projeto, uma grande quantidade de computadores analisa os dados provenientes do telescópio Arecibo, em Porto Rico, em busca de sinais de inteligência extraterrestre. Usando a Internet, o projeto reúne o poder de processamento de mais de 3 milhões de PCs em todo o mundo, os quais formam os nós da grade. Em cada nó da grade computacional, um programa que busca informação no sistema central via Internet, efetua o processamento e envia os resultados de volta.

O projeto *SETI@home* é considerado grade computacional por compartilhar recursos distribuídos e possibilitar uma alta dispersão geográfica em termos de abrangência dos recursos envolvidos.

Ambiente Empresarial - De maneira geral, considera-se que o cenário empresarial não pode depender de PCs para executar tarefas críticas ao negócio, e, portanto, grades deverão ser construídas em torno de servidores. Os PCs são muito mais instáveis que servidores, com freqüentes *boots* e muito mais facilidade de serem desligados. Um ambiente de negócios depende essencialmente de disponibilidade e segurança para a execução das tarefas com previsibilidade.

Entretanto, os inúmeros PCs espalhados pelas empresas representam uma imensa fonte de recursos computacionais disponível quase sempre ociosos.

O uso de PCs em ambiente empresarial apresenta aspectos diferentes dos projetos típicos de computação filantrópica. Algumas das principais diferenças são apresentadas a seguir:

	Computação Filantrópica	**Ambiente Empresarial**
Conexões	*Conexão remota, de baixa velocidade.*	*Conectados em redes de alta velocidade.*
Participação	*Voluntária.*	*Pode ser compulsória.*
Administração e segurança	*Não existem políticas nem obrigatoriedade de adoção.*	*Existem políticas a serem seguidas.*
Homogeneidade	*Ambientes diversos.*	*Padronizadas.*

Tabela 2.1: Computação Filantrópica x Ambiente Empresarial [TAURION, 2004].

O ambiente empresarial para grades é uma arquitetura basicamente cliente-servidor, com centenas ou milhares de PCs se conectando diretamente aos servidores centrais. Por estarem dentro de uma barreira física é possível implementar procedimentos e políticas corporativas, que minimizam os problemas e incertezas de um ambiente aberto, como os da computação filantrópica.

Data Grids - Outra configuração de grades computacionais é chamada de *Data Grid*. *Datas Grids* são usadas para prover acesso seguro a dados armazenados remotamente, em

sistemas diferentes, sem que os usuários precisem se preocupar se estão acessando dados locais ou remotos. Uma *Data Grid* permite que os dados sejam distribuídos geograficamente, proporcionando mecanismos de acesso que permitam que os usuários, de forma transparente, localizem e acesse estes dados.

Uma das primeiras alternativas que surgiram com o propósito de permitir acesso remoto a arquivos foi o NFS (*Network File System*), que se popularizou com o Unix. Infelizmente, o NFS apresenta diversas limitações, sendo que a principal é ser um protocolo típico de LAN, não escalando bem para redes amplas (WANs).

Outra solução é o FTP, mas que também apresenta inúmeras limitações para atuar em grade. Uma adaptação do FTP, o GridFTP, construído em cima do Globus Toolkit, foi desenvolvida para transferir arquivos em grades computacionais. Resolve problemas de privacidade e integridade, pois criptografa senhas e dados. Além disso, fornece um desempenho superior ao tradicional FTP.

Entre as aplicações deste produto podemos citar o desenvolvimento de *Data Marts* e *Data Warehouses*. Com Data Warehouse é possível criar aplicações de garimpagem ou mineração de dados.

Services Grids, OGSA e Web Services - As bases tecnológicas dos *Services Grids* são os *Web Services*. *Web Services* são softwares que se propõem a conectar uma empresa com seus clientes e parceiros de negócio.

Com *Web Services*, as empresas podem se conectar de maneira muito mais fácil, interligando seus sistemas aos dos parceiros de negócio. Estas aplicações podem ser interfaces computacionais dos seus processos de negócio, permitindo que estes sejam acessados diretamente pelos parceiros.

Web Services são programas projetados para executar uma função simples, com interfaces baseadas nos padrões XML (*Extensible Markup Language*). XML é um padrão independente de fornecedor que implementa uma meta-linguagem para criação de *tags*, que funcionam como nome de campos em uma descrição de bancos de dados, apontando ou marcando onde elementos específicos de dados residem em uma mensagem XML.

Para implementar *Web Services* são adotados três padrões baseados em XML: o SOAP (*Simple Object Access Protocol*), WSDL (*Web Services Description Language*) e UDDI (*Universal Description Discovery and Integration*).

O objetivo do SOAP é formatar mensagens entre serviços consumidores e produtores, implementando um mecanismo de RPC (*Remote Procedure Call*) entre aplicações. Pode ser definido como um conjunto de regras que facilitam o intercâmbio de mensagens XML entre aplicações. Em resumo, é uma peça de código que se comunica com outra peça de código.

Com o protocolo UDDI são criados os diretórios que permitem aos *Web Services* descobrirem-se uns aos outros, e interagirem pela Internet, sem intervenção humana. Pode ser definido como um conjunto de especificações para criação de serviços de diretórios. Basicamente é constituído por dois componentes: o UDDI *registry*, que é o servidor físico que armazena as descrições dos *Web Services*, e a interface UDDI, APIs que permitem aos programas interagirem através dos servidores de *registry*.

O WSDL descreve as características e detalhes do *Web Service* de modo que possa ser compreendido por qualquer outro *Web Service*. Pode ser definido como um *framework* para descrever *Web Services* e suas funções.

Segurança também merece atenção e alguns protocolos visando criar mecanismos de segurança em *Web Services* estão sendo implementados, como o SAML (*Security Assertion Markup Language*), para permitir a troca de informações de autenticação e autorização.

Envelopando XML em torno de uma aplicação cria-se um *Web Service*. O envelope XML descreve, torna visível e acessível o serviço implementado pelo *Web Service* para outras aplicações.

O programa Web Service é definido via WSDL e disponibilizado de maneira pública ou restrita por um diretório UDDI. Este diretório pode ser de uso interno, acessível apenas por *Web Services* da empresa, ou externo.

Com *Web Services* implementamos uma visão arquitetônica que é denominada SOA (*Services Oriented Architeture*), que permite que uma aplicação seja composta por componentes independentes, distribuídos e cooperativos.

Esta visão é compartilhada pelos conceitos de grades computacionais. Portanto, juntar *Web Services* e grades computacionais nos permite criar o que chamamos de *Services Grid*, grades constituídas de aplicações baseadas em *Web Services* [TAURION, 2004].

2.3 Arquitetura

A execução das aplicações em grade é realizada de forma paralela. As tarefas que compõem uma aplicação paralela executam em vários processadores, o que reduz o tempo de execução. As características de uso dos processadores qualificam a plataforma de execução da aplicação. Plataformas de execução de aplicações paralelas variam em diversos aspectos, dos quais destacamos conectividade, heterogeneidade, compartilhamento, imagem do sistema e escala.

Entender as diferenças entre plataformas é fundamental, porque cada aplicação paralela tem requisitos que podem ser atendidos de forma melhor ou pior em uma dada plataforma. Podemos agrupar as plataformas de execução hoje existentes em quatro grandes grupos: *SMPs*, *MPPs*, *NOWs* e grades computacionais [CIRNE, 2002].

- SMPs ou multiprocessadores simétricos são máquinas em que vários processadores compartilham a mesma memória. Multiprocessadores possibilitam um fortíssimo acoplamento entre os processadores e executam uma única cópia do sistema operacional para todos os processadores. Portanto, apresentam uma imagem única do sistema e excelente conectividade.

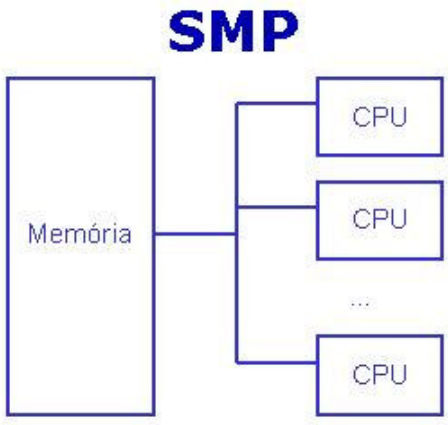


Figura 2.3: SMP. [CIRNE, 2002]

- MPPs ou processadores maciçamente paralelos são compostos por vários processadores e memórias independentes, interconectados por redes dedicadas e muito rápidas. MPPs incluem supercomputadores paralelos, como também clusters menores.

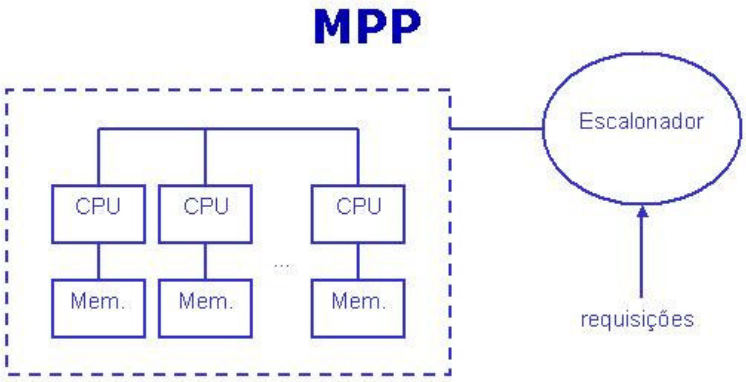


Figura 2.4: MPP [CIRNE, 2002].

- NOWs, ou redes de estações de trabalho, são um conjunto de estações de trabalho ou PCs, ligados por uma rede local. NOWs são arquiteturalmente semelhantes aos MPPs. Ambas plataformas são formadas por nós que agregam processadores e memórias. Uma diferença entre NOWs e MPPs é que os nós que compõem uma MPP tipicamente são conectados por redes mais rápidas que as que conectam os nós de NOWs.

A principal diferença entre ambas arquiteturas é que NOWs não são escalonadas de forma centralizada, ou seja, não há um escalonador para o sistema como um todo, mas cada nó tem seu próprio escalonador local.

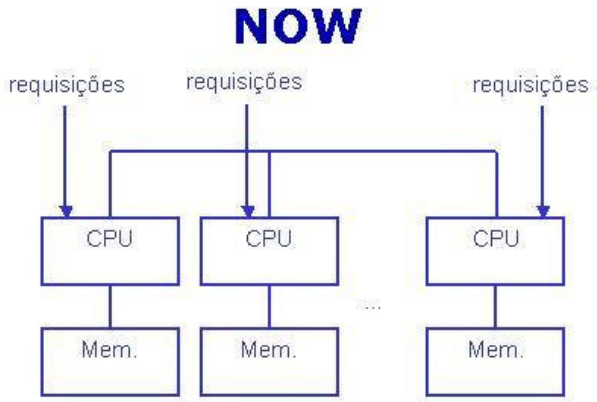


Figura 2.5: NOW [CIRNE, 2002].

- As grades computacionais são o passo seguinte aos NOWs no sentido de mais heterogeneidade e distribuição. Grades não fornecem uma imagem comum do sistema para seus usuários. Os componentes de uma grade podem estar sob controle de diferentes entidades e, portanto, em domínios administrativos diversos.

Um dado usuário pode ter acesso e permissões bastante diversas nos diferentes componentes de uma grade. A grade computacional não pode ser dedicada a um usuário, embora seja possível que algum componente possa ser dedicado. Uma aplicação de grade computacional deve estar preparada para lidar com todo dinamismo da plataforma de execução, adaptando-se ao cenário que se apresenta com o intuito de obter a melhor performance possível no momento.

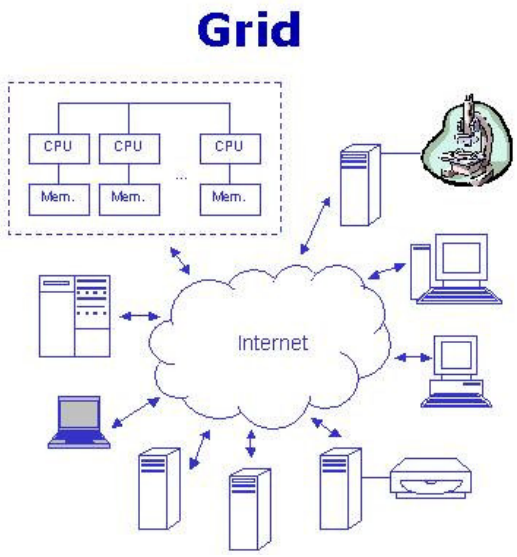


Figura 2.6: GRID [CIRNE, 2002].

A tabela 2.2 resume as características das várias plataformas:

	SMPs	MPPs	NOWs	Grids
Conectividade	excelente	muito boa	boa	média/ruim
Heterogeneidade	nula	baixa	média	alta
Compartilhado	não	não	sim	sim
Imagem	única	comum	comum	múltipla
Escala	10	1.000	1.000	100.000

Tabela 2.2: Características das plataformas de execução [CIRNE, 2002].

Analisando a tabela 2.2 podemos verificar que a arquitetura de grade avançou em características como heterogeneidade, compartilhamento, imagem e escala, mas piorou em relação à conectividade. O problema da conectividade em grades computacionais ocorre por causa da diversidade da topologia, largura de banda, latência e compartilhamento dos canais de comunicação.

De acordo com Foster, et al. a arquitetura de grades computacionais está organizada em 5 (cinco) camadas e foi dividida da seguinte maneira: Base (*Fabric*), Conectividade, Recursos, Coletiva e de Aplicações [FOSTER, 2001a].

- Base - É a interface para controle local dos recursos. Os componentes da camada base implementam localmente operações específicas de cada recurso, seja físico ou lógico. Existe então uma interdependência forte entre as funções implementadas no nível da camada base e as operações de compartilhamento suportadas. Funcionalidades mais ricas da camada base habilitam operações de compartilhamento mais sofisticadas. Se exigirmos menos dos elementos da camada base, a distribuição da infra-estrutura da grade será simplificada.

- Conectividade - Esta camada define o núcleo dos protocolos necessário de comunicação e autenticação para transações pela rede, específicas para a grade. Protocolos de comunicação habilitam a troca de dados entre recursos da camada base. Protocolos de autenticação são construídos sobre os serviços de comunicação para poder prover mecanismos criptografados e seguros para verificar a identidade de usuários e recursos.

- Recurso - Define protocolos para negociação, monitoramento, controle de operações compartilhadas em recursos individuais de forma segura. As implementações da camada de Recursos desses protocolos chamam as funções da camada Base para acessar e controlar recursos locais. Protocolos de informação são usados para obter informação sobre a estrutura e o estado dos recursos compartilhados. Protocolos de gerenciamento são usados para negociar acesso a recursos compartilhados.

- Coletiva - Camada que busca interações entre coleções de recursos. Por essa razão tem esse nome. Implementa uma ampla variedade de comportamentos de compartilhamento sem colocar novos requisitos nos recursos sendo compartilhados. Serviço de Diretório - permite aos usuários pesquisar por recursos pelo nome ou atributos como tipo, disponibilidade ou carga, podendo ser alocação conjunta e agendamento, serviços de monitoração e diagnóstico, serviços de replicação de dados, sistema de programação para grades, sistemas de gerenciamento de carga de trabalho, serviços de descobrimento de software, serviços de autorização de comunidade, serviço de colaboração, dentre outros. As funções da camada Coletiva podem ser implementadas como um serviço persistente, com protocolos e interfaces associadas projetadas para serem linkadas com aplicações. Em ambos os casos, sua implementação pode ser sobre a camada de Recursos.

- Aplicações – São as aplicações que rodam na infra-estrutura de grades.

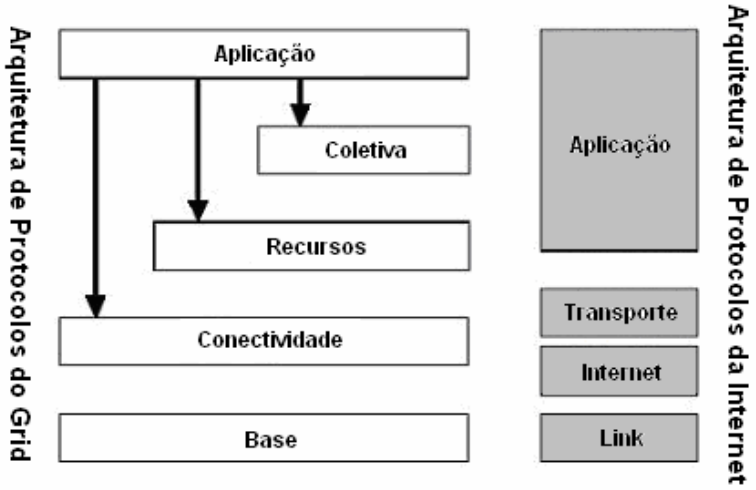


Figura 2.7: Arquitetura [FOSTER, 2001a].

Uma grade computacional tem algumas características próprias, que definem o seu conjunto de tecnologias. Estas características são a distribuição geográfica dos sistemas, a heterogeneidade dos sistemas, a escalabilidade e a adaptabilidade.

Para Cezar Taurion [TAURION, 2004] as tecnologias que compõem uma grade computacional podem ser visualizadas em camadas, com cada uma sendo composta por um conjunto de tecnologias.

1. A camada mais baixa é constituída pelos elementos básicos que são os recursos computacionais, como servidores, PCs e seus sistemas operacionais.
2. A seguir temos o middleware, que oferece serviços básicos como gestão de recursos distribuídos.
3. Logo após temos uma camada de serviços orientados aos usuários, como ambientes de programação em grade.
4. Finalmente vemos a camada de aplicações, onde estão os portais de acesso e os programas que exploram a potencialidade das grades computacionais.

Camada de Aplicações
Portais de acesso e programas que exploram a potencialidade das grades
Serviços Orientados aos Usuários
Ambientes de programação em grade
Middleware
Serviços básicos como gestão de recursos distribuídos
Elementos Básicos
Recursos computacionais, como servidores, PCs e seus sistemas operacionais.

Tabela 2.3: Camadas de tecnologias de grades computacionais [TAURION, 2004]

Para entendermos o modelo computacional de grade computacional, é preciso compreender quais são os desafios que esta tecnologia enfrenta. Vamos imaginar que estamos diante de uma situação em que um usuário de uma grade pretende executar um programa A usando os recursos localizados em um site B, sujeito às políticas de segurança de acesso P, e que necessita acessar dados localizados no site C, estes sujeitos a uma outra política de acesso, Q. Para que tudo isso funcione, é necessário implementar uma série de funcionalidades:

a) Identificação e autenticação: É necessário garantir que o usuário seja realmente quem ele diz ser. Compreende funções como *sign-on* único e integração com as soluções de segurança e integridade de cada computador.
b) Autorização e aderência a políticas: Procuram garantir que os usuários estejam autorizados e possam executar programas e acessar recursos em outros domínios diferentes do seu.

c) Localização dos recursos: Essa funcionalidade tem por objetivo buscar a localização dos recursos disponíveis e dados a serem analisados em determinado momento.
d) Caracterização dos recursos: Tem por objetivo identificar se determinado recurso é apropriado para executar determinada tarefa.
e) Alocação dos recursos: Busca definir quanto de cada recurso pode ser usado e qual a prioridade para a execução destes.
f) Contabilização e Nível de Serviço: Essas funcionalidades têm ligação com o monitoramento das tarefas para garantir que os serviços estão sendo executados de acordo com o definido.
g) Segurança: Verifica se as políticas de segurança estão sendo respeitadas.

A base de uma grade computacional é a capacidade de interoperabilidade entre seus componentes.

Para implementar as funcionalidades que permitem interoperabilidade é necessário criar uma série de protocolos e mecanismos que permitam aos usuários acessarem sistemas distribuídos por diversos locais. Os protocolos e mecanismos devem ser flexíveis o suficiente para aceitar na grade computadores de diferentes tecnologias e sistemas operacionais. De maneira geral devem ser implementados em camadas, de forma similar às camadas que compõem a Internet.

Os protocolos definem como os elementos distribuídos interoperarão uns com os outros e que informação será intercambiada durante estas interações. Protocolos são conjuntos de regras e definições. Os serviços implementam na prática estes protocolos. O protocolo descreve como determinada interação pode ocorrer entre dois ou mais sistema, sendo que os serviços por sua vez, implementam estas interações. Exemplificando, o protocolo define como estruturar as informações para que o estado de um determinado recurso, como um computador que pertence a grade computacional, seja reconhecido e os serviços implementam as rotinas que acessam tal computador e recuperem estas informações.

Os protocolos devem definir os serviços de interface de programação (API – *Application Program Interfaces*) que permitam desenvolver aplicações que explorem as funcionalidades embutidas nos próprios protocolos da grade. Os sucessos da grade computacional serão determinados não pela sofisticação técnica dos seus protocolos, mas pela importância dos

problemas a que se propõem resolver. Claramente, nem todas as aplicações poderão extrair benefícios da tecnologia de grade computacional. As que mais aproveitarão grades serão as que puderem trabalhar em paralelo. Quanto maior o grau de paralelismo conseguido por uma aplicação, maior será o seu potencial de exploração das grades [TAURION, 2004].

Na utilização de grade computacional a tecnologia pode ser empregada de diferentes maneiras com o objetivo de atender o objetivo proposto pela aplicação.

2.4 Sistemas para Provimento de Aplicações em Grades

Vários sistemas para suporte à computação em grades surgiram nos últimos anos, tanto através de esforços de instituições de pesquisa (*Globus*, *Legion*, *Condor*, *MyGrid*), quando decorrentes de empreendimentos comerciais (*Entropia*, *distributed.net*). Dentre os esforços de instituições de pesquisa, Globus e depois Legion foram os projetos que mais evoluíram. Para ilustrar alguns serviços para computação em grade, os sistemas Globus e Legion serão apresentados.

Globus possui um conjunto de serviços que facilita a computação em grade. Esses serviços facilitam o desenvolvimento de aplicações, pois não precisam se preocupar com questões como descoberta de recursos, movimentação de dados e segurança no ambiente da grade. Os serviços Globus já estão implementados para isso.

Apesar de Legion apresentar muitas possibilidades, os programadores de aplicações precisam desenvolver os objetos que são integrados a grade computacional como objetos Legion para fornecer o serviço desejado.

Os sistemas para provimento de aplicações em grades computacionais são *middlewares* que fazem o gerenciamento de recursos, permitindo o aproveitamento de estruturas já existentes.

Em alusão a arquitetura de grades computacionais, os *middlewares* estão acima da camada de aplicação fazendo todo o gerenciamento de recursos e integrando as atividades entre as camadas.

Em computação, *middleware* consiste em um conjunto de agentes de software agindo como intermediário prestando serviços às aplicações distribuídas e fornecendo uma visão unificada do sistema que se apresenta ao usuário como um único sistema distribuído.

2.4.1 Globus

Características Gerais - Globus é um conjunto de serviços que facilitam computação em grades. Os serviços Globus podem ser usados para submissão e controle de aplicações, descoberta de recursos, movimentação de dados e segurança na grade. Alguns dos principais serviços Globus disponíveis são GSI - Segurança, autenticação única na grade, GRAM (*Globus Resource Allocation Manager*) - Submissão e controle de tarefas, Nexus - Comunicação entre tarefas, MPI-G - MPI (*Message Passing Interface*) sobre Nexus, MDS - Informações e diretórios, GASS - Transferência de arquivos e GridFTP - Transferência de arquivos.

Segurança e Autenticação - Um aspecto crítico no uso de grades na prática é a autenticação de usuários em diferentes domínios administrativos. Em princípio, o usuário tem que se autenticar em cada domínio administrativo de uma forma determinada pelo administrador do domínio. Este esquema coloca uma grande carga no usuário. No contexto de computação em grades, os problemas de múltipla autenticação são agravados, pois queremos ter programas que possam efetuar ações que exigem autenticação.

GSI (*Globus Security Infrastructure*) é o serviço que ataca este problema. GSI viabiliza o login único na grade. O usuário deve se identificar junto ao GSI, após tal identificação todos os demais serviços saberão de forma segura, que o usuário é de fato quem diz ser. Uma vez que um serviço sabe a identidade Globus do usuário, resta estabelecer quais operações tal usuário pode realizar. Isto é feito mapeando a identidade Globus para um usuário local [CIRNE, 2002].

Alocação e Descoberta de Recursos - Grades não têm um escalonador que controla todo o sistema. Quando um usuário submete uma aplicação para execução na grade, o usuário utiliza um escalonador de aplicação que escolhe os recursos a utilizar, particiona o trabalho entre tais recursos, e envia tarefas para os escalonadores dos recursos. Em Globus, os escalonadores de recurso são acessados através do serviço GRAM. GRAM fornece uma interface única que permite submeter, monitorar e controlar tarefas de forma independente do escalonador de recursos. Assim sendo, escalonadores de aplicação não precisam entender dos detalhes particulares de cada escalonador de recurso.

Uma idéia bastante interessante em Globus é que escalonadores de aplicação podem usar os serviços de outros escalonadores de aplicação. O escalonador que recebe a solicitação do

cliente lida com a especificação em mais alto nível. Ele refina tal especificação e, para implementá-la, submete novas solicitações a escalonadores de recurso.

Processamento Paralelo - Globus suporta bem a hierarquia de escalonadores através da linguagem RSL (*Resource Specification Language*). RSL é capaz de expressar tanto solicitação de alto nível, como também solicitações concretas. Portanto, o trabalho de um escalonador de aplicação em Globus pode ser descrito como sendo o de refinar solicitações RSL. Em Globus existe um realocador que é um escalonador de aplicação especializado em garantir que tarefas localizadas em máquinas distintas executem simultaneamente.

O realocador é fundamental para execução em grades de aplicações fortemente acopladas. Em aplicações fortemente acopladas, as tarefas precisam se comunicar para que a aplicação faça progresso. Portanto, todas as tarefas da aplicação têm que ser executadas paralelamente. É importante ressaltar que uma boa implementação de realocação depende da implementação do serviço de reserva adiantadas por parte dos escalonadores de recursos. Reservas adiantadas permitem a escalonadores de aplicação obter garantias de escalonadores de recurso que determinados recursos estarão disponíveis para aplicação em um intervalo de tempo preestabelecido permitindo que as tarefas em execução continuem a serem executadas de forma paralela [GLOBUS, 2005].

Comunicação Intertarefas – O conflito entre generalidade e performance é um eterno problema quando se trata de comunicação, e não é diferente em grades computacionais.

Globus ataca este problema com o Nexus. Nexus fornece uma interface de baixo nível, mas uma implementação adaptável que escolhe, dentre as tecnologias de comunicação disponíveis, a que vai oferecer melhor performance. Por exemplo, se ambas tarefas estão em uma máquina de memória compartilhada, Nexus utilizará a memória para efetuar a comunicação [CIRNE, 2002].

Transferência de Dados - A necessidade de acesso remoto e transferência de dados é uma constante na computação em grade. Globus disponibilizou GASS (*Global Access to Secundary Storage*), um serviço para acesso remoto a arquivos sob a tutela de um servidor. O cliente GASS é uma biblioteca C que é vinculada à aplicação usuária do serviço. Com o intuito de fornecer boa

performance, o serviço GASS implementa as otimizações típicas de acesso remoto como *caching* e *pré-fetching*. GASS encontrou problemas de implantação e a dificuldade encontrada foi de interoperabilidade.

Essa realidade motivou a introdução do GridFTP por parte da equipe Globus. GridFTP estende o popular protocolo FTP para torná-lo mais adequado para as necessidades da computação em grade. Uma vez que GridFTP é uma extensão do FTP, o problema de interoperabilidade fica resolvido.

2.4.2 Legion

Características Gerais - Legion foi concebido em Agosto de 1993, mas permaneceu no papel até 1996. Nesse período os pesquisadores do projeto refinaram o modelo de objetos e de segurança, arquitetura e outras metas principais. Ele é um projeto de software de um metasistema baseado em objetos da Universidade da Virgínia, para um sistema com milhões de hosts com vários objetos juntos para serem compartilhados em alta velocidade. Nessa época um protótipo do Legion foi desenvolvido, baseado em um projeto anterior chamado Mentat. Nesse projeto os pesquisadores colocaram em prática todos os conceitos e problemas que o Legion poderia solucionar como, por exemplo, aqueles relacionados ao gerenciamento dos recursos compartilhados.

De uma maneira geral, o Legion é um *middleware* que conecta redes, estações de trabalho, supercomputadores e outros recursos computacionais com arquiteturas, características e sistemas operacionais diferentes, todos espalhados por localidades fisicamente distribuídas, em um único e poderoso sistema computacional. Não há uma central que controla e fiscaliza cada recurso disponível; em vez disso, cada recurso é, por si só, um elemento independente na grande rede Legion.

O Legion integra uma variedade de recursos de hardware e software. Cada um destes itens é representado como um objeto Legion, que é um processo ativo que responde a invocações feitas por outros objetos. O Legion não define uma linguagem de programação ou um protocolo de comunicação. O Legion apenas define um formato de mensagens e um protocolo de alto nível.

A definição e o gerenciamento de objetos Legion é feito por seu objeto de classe, que por si só, é um objeto Legion ativo. Esse tipo de objeto atua em nível de sistema, sobre suas

instâncias, realizando criação de novas instâncias, agendamento de execuções, ativação e desativação das instâncias, e fornece informações sobre sua localização atual para que clientes possam comunicar-se com esta. Classes cujas instâncias são classes dela mesma são denominadas metaclasses.

O Legion permite que os programadores escrevam os seus próprios objetos de classe e assim, eles podem escolher qual adapta-se melhor as suas necessidades. As versões mais recentes do Legion já contêm implementação de muitos tipos de classes e metaclasses, mas os usuários do Legion não são obrigados a utilizá-las [LEGION, 2003].

Segurança e Autenticação – Quanto ao serviço de identificação dos objetos denominado *naming*, o Legion implementa-o em três níveis. Em alto nível um usuário refere-se a um objeto por um nome (*string*) que seja entendível a este, denominado *context name*. Quando um objeto deve ser localizado, este nome é mapeado, por *objetos context*, para LOIDs (*Legion Object Identifiers*), que são ainda identificadores que não apontam diretamente para o endereço físico em que se encontra o objeto. Os LOIDs contém informações como: identificador da classe, número de uma instancia e chave pública.

Os LOIDs não contém informações suficientes quanto à localização do objeto e logo esses têm que ser mapeados para um outro tipo de objeto: o LOA (*Legion Object Address*). Um objeto LOA contém o endereço físico que é informação suficiente para que outro objeto possa comunicar-se com esse.

As várias possibilidades que o Legion oferece são atrativas, mas os usuários só adotarão o Legion se eles se sentirem confiantes de que ele protegerá a privacidade e integridade de seus recursos.

Para o Legion há quatro requisitos principais que devem ser satisfeitos:

Não prejudicar um host: A instalação do Legion em um host não deve comprometer as políticas e metas de segurança do computador. Em geral, Legion não deve permitir acesso não autorizado aos recursos do sistema.

Fornecer uma estrutura de controle de acesso: Todos os recursos locais são representados no Legion como objetos, e o recurso fundamental do Legion é a capacidade de chamar um método de um objeto. Os objetos devem ter mecanismos de controle de acesso flexíveis para autorizar chamadas de métodos.

Manter e proteger identidades: Objetos e usuários possuem identidades que podem ser usadas para autenticar e autorizar outro objeto. Essas identidades são representadas através de chaves privadas e credenciais de vários tipos.

Proteger a comunicação: Um sistema Legion pode se espalhar por redes públicas ou semipúblicas. Os objetos devem ser capazes de se comunicar com garantia de integridade e privacidade.

Processamento Paralelo - Um dos maiores desafios computacionais vem sendo o alcanço de uma alta performance através das técnicas de paralelismo. O projeto Legion vem basicamente fornecer um alto índice de desempenho através do processamento paralelo, dando ao usuário a impressão de estar utilizando um grande computador virtual com um alto poder de processamento.

Diferentemente da clusterização o Legion conecta redes, estações de trabalho, supercomputadores e outros recursos computacionais, espalhados por localidades fisicamente distribuídas (regiões, países, continentes etc.), constituindo assim um poderoso sistema computacional.

O Legion suporta o processamento paralelo de quatro maneiras:

Suporte a bibliotecas paralelas populares, como MPI: Isto significa que os usuários podem fazer uso dos benefícios do Legion bastando para isso recompilar as aplicações para que funcionem no mesmo, visto que a maioria das aplicações atual é escrita em MPI.

Suporte a linguagens paralelas, como MPL: O MPL é uma linguagem paralela do C++, onde o usuário especifica as classes que são muito complexas computacionalmente. As instâncias das classes são usadas como instâncias do C++. Dessa forma o compilador e o sistema de run-time examinam e fazem gráficos de computação paralela e a executam em paralelo nos diferentes processadores.

Suporte a componentes paralelos agrupados (encapsulamento de componentes): É feito um encapsulamento dos componentes nos chamados objetos do Legion. O objeto encapsulado parece seqüencial, mas executa mais rapidamente. Vale lembrar que as aplicações compartilhadas podem ser encapsuladas num objeto do Legion.

Exportação das interfaces das bibliotecas de run-time para outras bibliotecas: Como o Legion é aberto para suportar desenvolvimento de terceiros, a biblioteca *run-time* completa está disponível. As bibliotecas podem facilmente manipular a biblioteca *run-time*.

Alocação e Descoberta de Recursos – A alta performance é garantida pelo Legion baseado em dois fatores:

Seleção dos recursos: Por mais simples que seja uma tarefa, a escolha do recurso é importante. Nesse caso o Legion tem a função de distribuir uma tarefa optando pelo *host* que estiver com a menor carga de trabalho ou aquele que possui o maior poder de processamento.

Uso do paralelismo: O Legion suporta um modelo de computação paralela distribuída, estando os objetos do Legion em *hosts* diferentes, talvez a quilômetros de distância. Isso torna o Legion pouco apropriado para computação paralela de granulação fina.

O Legion baseia-se num processo de negociação entre quem solicita um recurso e quem irá prover este recurso. O escalonamento de recursos neste caso é feito em nível de aplicação. Dessa forma a autonomia torna-se fundamental nesse processo.

Autonomia do site: É fundamental para atrair os provedores de recurso, mas vale ressaltar que cada site quer que suas políticas sejam respeitadas pelo sistema, sendo assim, a autorização final para o uso de um recurso depende do proprietário do recurso.

Autonomia do usuário: Também é importante para se conseguir alta performance. Uma única política de escalonamento não seria a melhor solução para todos os casos. Cada usuário deve ter a autonomia para escolher aquelas que melhor se ajustam ao seu problema. É de vital importância que o usuário tenha seus escalonadores de aplicação, pois estes melhoram consideravelmente a performance de uma aplicação paralela.

Comunicação Intertarefas – Os objetos Legion comunicam-se com outros por meio das interfaces (métodos de acesso público) oferecidas por estes. Essas interfaces podem ser descritas em uma linguagem de descrição de interfaces (IDL - *Interface Description Language*) sendo suportadas pelo Legion.

Transferência de Dados - O LegionFS é um sistema de arquivos desenvolvido para o projeto Legion. O LegionFS está inserido dentro do ambiente Legion que é um *middleware* que

provê a ilusão de uma única máquina virtual e a segurança para acesso indevido e processamento distribuído.

LegionFS representa um arquivo por uma abstração denominada *BasicFileObject*. Os métodos de *BasicFileObject* são semelhantes a chamadas de sistemas do UNIX como *read*, *write* e *seek*. *ContextObjects* gerenciam o contexto dos nomes. Além disso, são usados *ProxyMultiObjects* como um processo que encapsula vários arquivos e contextos que residem na mesma máquina para obter eficiência. Um *ProxyMultiObject* busca as requisições para o arquivo ou contexto correspondente. É implementado como um processo leve, funcionando como uma camada intermediária entre o sistema de arquivo e a aplicação [WHITE, 2001].

2.4.3 Iniciativas de Grades Nacionais

No Brasil temos diversas iniciativas de grade em andamento como o OurGird, CBPF, LNCC (Laboratório Nacional de Computação Científica), experiências na Embrapa, no CPTEC (Centro de Previsão de Tempo e Estudos Climáticos), e vários outros. A seguir, apresentaremos como exemplo algumas destas iniciativas.

OurGrid

OurGrid é uma solução de grade para execução de aplicações *Bag-of-Tasks*. Aplicações *Bag-of-Tasks* são aquelas aplicações cujas tarefas são independentes umas das outras. Não é viável fazer computação em grades de aplicações fortemente acopladas na Internet de hoje.

Exemplos de Aplicações *Bag-of-Tasks*: *Data mining*, pesquisa massiva (como quebra de chave e sequenciamento de gens), varredura de parâmetros, simulações, *fractais* (como *Mandelbrot*) e manipulação de imagem (como tomografia).

Componentes do OurGrid: MyGrid Broker permite que um usuário utilize todas as máquinas a que tem acesso. OurGrid Community fornece acesso a máquinas dentro de um grupo estabelecido para determinada funcionalidade em comunidade e SWAN cuida da Segurança.

Um dos principais objetivos do OurGrid é funcionar como uma rede de favores em que os usuários locais sempre têm prioridade aos seus recursos. Recursos ociosos são doados para comunidade. O maior problema enfrentado pelo OurGrid é como fazer esta doação de forma justa [OURGRID, 2005].

CBPF

No CBPF (Centro Brasileiro de Pesquisas Físicas) está em andamento um projeto experimental de grade. A Física computacional demanda, pela pesada utilização de simulações, um ambiente de computação massivo.

O objetivo do projeto é permitir a solução de problemas da física, utilizando uma infra-estrutura computacional maior do que a que existe localmente, compartilhando recursos e informações com outros pesquisadores e centros de pesquisa [CBPF, 2005].

LNCC

No LNCC (Laboratório Nacional de Computação Científica), o projeto GRADE tem por objetivo a construção de uma grade computacional composta por uma infra-estrutura escalável de clusters, distribuídos geograficamente por diversos centros do país. A estrutura proposta inclui a RNP (Rede Nacional de Ensino e Pesquisa), centros de computação de alto desempenho e centros institucionais, divididos em cluster nacional, clusters regionais e clusters institucionais. O uso de clusters de diversas topologias permitirá a convivência de aplicações que requeiram modelos computacionais diferentes, como computação fortemente acoplada (como meteorologia) e fracamente acoplada, como bioinformática e física de alta energia [LNCC, 2005].

2.5 Segurança

As aplicações em grades computacionais são distintas das aplicações tradicionais pelo uso simultâneo de um grande número de recursos, domínios administrativos múltiplos, estruturas complexas em termos de comunicação, altas exigências de desempenho, dentre outras.

Todas essas características conduzem a problemas de segurança que não podem ser gerenciados somente por tecnologias de segurança existentes para sistemas distribuídos comuns.

As computações paralelas em grades, que adquirem recursos computacionais múltiplos, têm a necessidade de estabelecer não simplesmente relacionamentos de segurança entre um cliente e um usuário, mas entre centenas de processos que utilizam coletivamente muitos domínios administrativos. Além disso, a natureza dinâmica da grade computacional pode tornar impossível de se estabelecer relacionamentos de confiança entre locais diferentes antes da execução da aplicação.

As exigências de segurança são fundamentais a um projeto de grade. Os componentes básicos de segurança devem fornecer os mecanismos para a autenticação, a autorização, e confidencialidade de uma comunicação entre grades de computadores. Sem estas funcionalidades, a integridade e confidencialidade dos dados processados dentro da grade estariam em risco.

Dentro de todo o ambiente de rede, há alguns riscos e exposições envolvidas com a segurança de sua infra-estrutura. A menos que os computadores estejam desconectados em uma sala fechada, há a possibilidade de alguém burlar a segurança e acessar os recursos teoricamente protegidos. Se existem estes riscos, um dos principais objetivos da segurança é ajudar a reduzir esse risco para um nível aceitável.

O objetivo da segurança é examinar os requisitos e utilização de ferramentas e processos necessários para reduzir os riscos envolvidos. O grau de segurança necessário é baseado no tipo de topologia da grade e os dados que se estará protegendo. As exigências da segurança para um projeto de grade dentro de um banco serão completamente diferentes daquelas de uma instituição acadêmica.

Os componentes da infra-estrutura de segurança (*firewalls*, IDS, antivírus, criptografia, etc...), os riscos envolvidos e os processos para controlar estes componentes são objetos de estudo para a segurança em grade, e serão discutidos em maior detalhes no capítulo 3.

2.6 Perspectivas

Segundo Ian Baird [BAIRD, 2002], a criação e a consolidação da infra-estrutura de grades computacionais será feita, aproximadamente, em 3 fases.

Fase 1 - Na primeira fase temos o desenvolvimento dos chamados *Enterprise Grids*, que são as implementações comerciais de grades por corporações que possuem presença global ou que precisam acessar recursos fora de uma simples corporação local.

Fase 2 - Para a segunda fase teremos os *Partner Grids*, que irão surgir das operações das organizações com industrias similares ou áreas de interesse colaborativo em projetos de objetivo em comum.

Fase 3 - Chamada de *Service Grids*, irá acontecer quando os usuários adotarem o sistema de grades computacionais para a realização de serviços.

De forma equivalente, ocorrerá uma expansão gradualmente da seguinte forma:

Regional - empresas, universidades ou organizações se juntam em uma mesma região em um intuito de colaboração para aumentar seu poder de processamento através do compartilhamento de recursos.

Estadual - quando as grades computacionais regionais começarem a crescer será mais fácil à interligação das grades entre estados, novamente com um objetivo de colaboração.

Global - cria-se um interesse de interligar estas grades entre os paises, possibilitando um poder de colaboração muito grande.

As fases não serão executadas necessariamente em ordem, pois já existem esforços e interligações sendo montadas entre grades de diferentes países, como EUA, Europa e Austrália, além de outros. Mas haverá um crescimento natural, principalmente pela maior facilidade física e operacional na montagem.

2.7 Resumo

Este capítulo apresentou alguns conceitos básicos de computação em grades trazendo uma visão necessária para o entendimento do restante do texto: aspectos relevantes sobre grades computacionais como aplicações, arquitetura, sistemas para provimento de aplicações em grades, uma breve introdução sobre segurança, e por fim perspectivas são apresentadas. O próximo

capítulo discute problemas de segurança em grades Computacionais, finalizando com propostas de soluções para tais problemas.

Capítulo 3

Segurança em Grades

O assunto segurança por si só já nos permite perceber a importância que deve ser dada a este tema. Quando tratamos de segurança em ambientes computacionais, muitos fatores estão envolvidos, dentre eles autenticação, controle de acesso, integridade, privacidade, não repúdio e outros.

No ambiente de computação em grades isso não é diferente, porém a complexidade é bem maior uma vez que novos problemas e situações são encontrados.

Um dos principais problemas está relacionado à construção de soluções capazes de coordenar e garantir segurança em diversas políticas de controle de acesso. Um outro ponto é em relação à capacidade de operar em uma infinidade de ambientes totalmente heterogêneos, levando-se em consideração software, hardware, protocolo e todo tipo de estrutura.

A computação em grade é composta de um grupo dinâmico de processos que funcionam em recursos e em locais diferentes. As grades computacionais permitem que os recursos ociosos adquiram processos, comecem processos e ainda liberem recursos dinamicamente durante sua execução devido à prioridade de processamento.

A complexidade envolvida nessa troca dinâmica de recursos e processos pode tornar o ambiente da grade vulnerável, instável e propício a ataques.

Este trabalho está baseado em um estudo sobre questões de segurança em grades computacionais. Este capítulo faz um levantamento sobre segurança em grades. A seção 3.1 apresenta os problemas clássicos, iguais aos encontrados em outras aplicações de processamento distribuído e questões necessárias para a resolução destes problemas. Na seção 3.2 serão abordados os problemas específicos a grades computacionais e propostas para a resolução de tais problemas e a seção 3.3 conclui o capítulo.

3.1 Problemas clássicos em ambientes distribuídos

Muitas ferramentas aplicadas em outros ambientes de computação distribuída também são utilizadas em ambientes de grade. Desta forma, os problemas de segurança encontrados em outras aplicações de processamento distribuído tem seus similares em cenários de grades.

Sabemos que os problemas que classificamos como clássicos também são encontrados em aplicações de grades. Em grades computacionais estes problemas apresentam uma complexidade muito maior devido as principais características que são: heterogeneidade, escalonamento e conectividade. A diversidade de plataformas e recursos envolvidos torna questões de autenticação, por exemplo, muito mais complexas.

Ataques

Ataques como *spoofing*, negação de serviço, *buffer overflow* e outros, possíveis de ocorrer em qualquer aplicação, também podem acontecer em aplicações de grade. Alguns exemplos seguem abaixo:

Spoofing: O *spoofing* é uma técnica na qual o endereço real do atacante é mascarado, de forma a evitar que ele seja encontrado ou fazer-se passar por outro usuário. Essa técnica é muito utilizada em tentativas de acesso a sistemas nos quais a autenticação tem como base endereços IP, como a utilizada nas relações de confiança em uma rede interna. O *Spoofing* não é exatamente uma forma de ataque, mas sim uma técnica que é utilizada na grande maioria dos ataques, pois ele ajuda a esconder a identidade do atacante. A técnica de *spoofing* é também utilizada juntamente com um ataque de negação de serviço, o qual consiste em disparar algum processo que sobrecarregue a máquina ou algo que ela não consiga finalizar [COLE, 2001].

Negação de Serviço: Uma forma de prejudicar o processamento em grades é a sobrecarga de processos gerados de forma intencional que comprometam a disponibilização de serviços alimentados pela computação em grades. Isso ocorre quando um determinado recurso nunca se torna disponível uma vez que seus ciclos de processamento sempre estarão ocupados devido a algum tipo de ataque. No caso das grades isso pode causar perdas pelo fato de muitos recursos

que poderiam ser usados em quanto ociosos, não são possíveis de serem identificados pela grade para a execução de serviços.

Buffer Overflow: O ataque *buffer overflow* funciona inserindo muitos dados dentro da pilha de memória do computador, o que causa que outra informação que está na pilha seja sobrescrita. Como você pode imaginar, informações importantes, são armazenadas e acessadas a partir das pilhas da memória, por exemplo, todas as informações manipuladas pelo sistema operacional.

As dificuldades de proteção contra esses ataques em grades computacionais existem pela grande diversidade de domínios administrativos, o que dificulta a localização da origem dos ataques, uma vez que várias formas de escalonamento e busca de recursos são empregados, cruzando as barreiras entre os domínios. Na transição dessas fronteiras, requisitos como identificação, autorização, integridade, confidencialidade, não repúdio, privacidade, devem ser considerados.

Identificação

Um dos maiores problemas é a identificação de pessoas e recursos dentro de um ambiente de grade computacional. É preciso garantir que determinado usuário ou recurso seja realmente quem ele diz ser para que as informações de um determinado domínio não fiquem sobre riscos.

Autorização

Além da identificação um outro aspecto é a autorização para que uma pessoa ou recurso possa operar dentro de um determinado domínio. Muitas vezes o acesso é aberto para que pessoas utilizem um determinado recurso por apenas um período e depois este acesso é fechado. Este controle deve ser realizado para que uma pessoa ou recurso não utilize um recurso ou informações sem o consentimento das entidades administradoras do domínio.

Integridade

A integridade das informações é um outro aspecto de segurança muito importante principalmente se levarmos em conta a diversidade de recursos, usuários e processos que podem estar envolvidos neste ambiente.

Confidencialidade

Da mesma forma que a integridade, a questão da confidencialidade também é um fator importante para se ter atenção. Uma informação só é válida para uma empresa, órgão cientifico ou acadêmico se esta for integra e confiável.

Não Repúdio

A garantia de que o emissor de uma mensagem ou a pessoa que executou determinada transação de forma eletrônica, não poderá posteriormente negar sua autoria é mais um ponto de grande importância. Para cada informação transmitida ou processo executado é necessário um responsável.

Privacidade

À medida que a computação se torna cada vez mais presente, o mesmo ocorre com a quantidade de informações e dados que as pessoas transmitem. As pessoas têm o direito de controlar suas informações pessoais, o direito de não serem importunadas e o direito de ter uma experiência segura na qual confiem nas tecnologias, serviços e soluções.

3.2 Problemas Específicos em Grades

Como já sabemos, uma grade computacional é uma coleção de computadores heterogêneos e de recursos espalhados através de diferentes domínios administrativos com a intenção de fornecer acesso fácil aos usuários a estes recursos. Há muitas maneiras de alcançar os

recursos de uma grade computacional, e cada uma com exigências distintas de segurança e de implicações para o usuário do recurso e o fornecedor do recurso.

Em grades computacionais existe a necessidade de mecanismos para assegurar a estabilidade de processos. Essa necessidade torna-se muito mais complexa, sendo que os processos podem ser interrompidos em um local por algum motivo (prioridade de processamento, por exemplo), transportados para outro recurso que se encontra ocioso e reiniciados do mesmo ponto em que pararam. No ambiente de computação em grades é muito importante a atribuição de responsabilidade aos geradores de informações utilizadas nos processos envolvidos.

As grades computacionais permitem a partilha e a coordenação de diversos recursos distribuídos em organizações virtuais (VOs). As VOs são compostas por múltiplas instituições com diferentes políticas e tecnologias de controle e segurança. Cada instituição poderá ter recursos muito valiosos e não deseja que esses recursos sejam mal utilizados. A natureza de um sistema deste tipo traz problemas de segurança de difícil resolução. As organizações clássicas apenas têm políticas que apenas tratam de utilizadores locais. O acesso da VO tem de ser estabelecido e coordenado entre o utilizador e a sua organização.

Os mecanismos de segurança da grade devem trabalhar com a infra-estrutura das organizações e não substituí-las.

Sistemas em grade podem requerer praticamente todos os padrões de segurança existentes em virtude da diversidade de cenários.

No desenvolvimento da arquitetura de segurança para aplicações em grade é verificado o encontro de inúmeros requerimentos de segurança. O Redbook da IBM faz algumas recomendações como a seguir [IBM_REDBOOK, 2003]:

- Acesso Sign-on: Um usuário é autenticado uma única vez quando o processo é iniciado e busca recursos, usa recursos, libera recursos, e se comunica internamente sem serem exigidas autenticações futuras do usuário.

- Proteção de Credenciais: Credenciais de usuários (senhas e chaves privadas) devem obrigatoriamente ser protegida.

- Interoperabilidade com outras políticas de segurança: Enquanto soluções de segurança podem prover mecanismo de acessos entre domínios conhecidos, o acesso a domínios com políticas de segurança distinta é praticamente inviável. Para isso usa-se de agentes que agem em nome do usuário uma vez que é impossível a alteração das políticas de segurança já que cada domínio tem por objetivo se alto proteger.

- Exportação de código: É desejado que um código gerado em um local seja executado em outro. É necessário identificar a maneira que o código deve ser apresentado para que possa ser exportado e executado em outro local.

- Uniformidade de Infra-estrutura de Credenciais e Certificados: Acesso entre domínios requer, no mínimo, um caminho comum para a realização de identificação entre usuários e recursos. Então deve se empregado um padrão, tal como X.509, para a codificação de credenciais aplicados nos princípios de segurança.

Partindo destas premissas é que iremos identificar problemas específicos de segurança em grades em razão da grande diversidade de cenários, políticas, variações de execução das tarefas e pontos vulneráveis existentes.

Para facilitar esta explanação apresentaremos dois cenários e através dos quais ilustraremos os principais problemas de segurança específicos ao ambiente de grade computacional. O primeiro cenário, mais controlado, em que as aplicações distribuídas em grades de alta escala são providas por sistemas como Globus e Legion. O segundo cenário, aberto, como ambientes filantrópicos de auxílio a pesquisas.

3.2.1 Cenário 1: Computação distribuída em alta escala

Para começar a explanação de problemas de segurança específicos em grades computacionais ilustraremos um problema de segurança conforme o exemplo da figura abaixo (figura 3.1). Este exemplo demonstra uma situação real de aplicação em grades [FOSTER, 2001b].

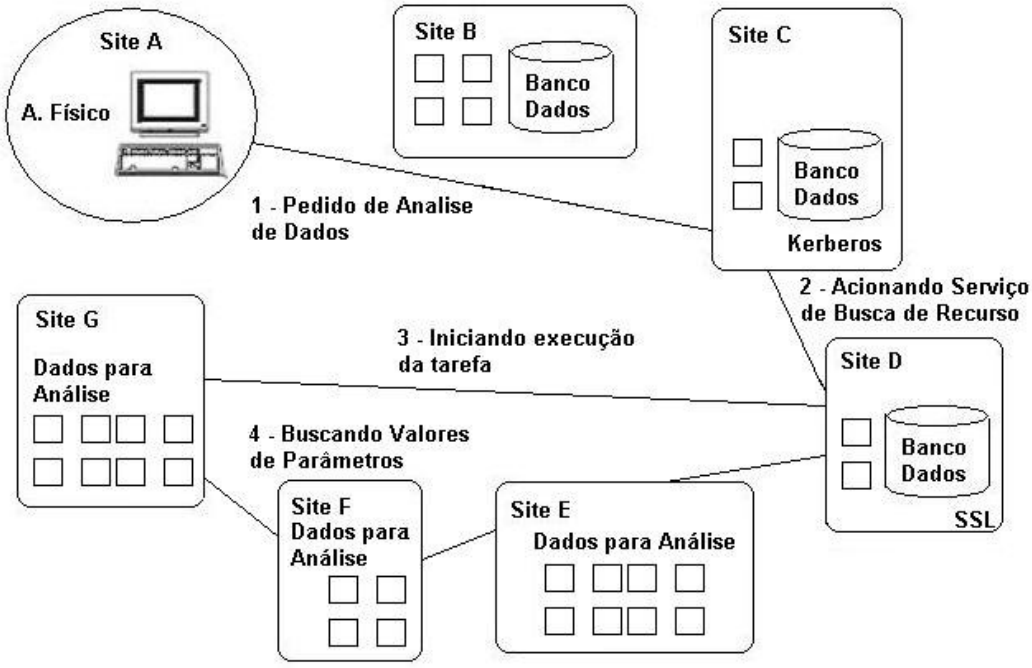


Figura 3.1: Exemplo de computação distribuída em larga escala [FOSTER, 2001b].

Um físico precisa iniciar uma análise que realize o processamento de dados em diferentes locais. Para isso, o físico faz com que um programa de análise seja iniciado enviando a solicitação de análise para uma localização remota onde os dados estão armazenados (chamaremos de local C). Uma vez iniciada a análise o programa determina que é necessário executar simulações para comparar os resultados do experimento com valores previamente estabelecidos.

Como o volume de informações a serem analisadas é enorme, um serviço de busca de recursos mantido pela colaboração (local D) é ativado com o objetivo de encontrar recursos ociosos que possam ser utilizados para executar as simulações de comparação dos resultados obtidos nas análises com o objetivo de diminuir o tempo de processamento.

O serviço de busca é iniciado em computadores localizados em (E e G). Mas por sua vez os computadores devem acessar parâmetros de controle de acesso de usuário e aplicação em servidores localizados em um outro local (F), por exemplo. Isso ocorre devido às distintas políticas de controle de acesso.

Toda essa comunicação pode ser realizada utilizando-se protocolos especializados, orientados pelo serviço de busca de recursos.

Para o cenário apresentado por Foster, et al., este exemplo ilustra algumas características de um ambiente de computação em grades e por conseqüência alguns problemas [FOSTER, 2001b].

- A população de usuário é grande e dinâmica. Os participantes em organizações virtuais como esta colaboração científica incluem membros de muitas instituições e são trocados constantemente.
- A gama de recursos é grande e dinâmica. As instituições e os usuários individuais são responsáveis por decidir quando e com quais recursos contribuir, por isso a quantidade e a posição de recursos disponíveis podem mudar rapidamente.
- Os processos criados por uma computação podem adquirir, começar processos e ou liberar recursos dinamicamente durante sua execução. Mesmo em nosso exemplo que é simples, a computação adquiriu, iniciou e posteriormente liberou recursos em cinco locais distintos. Durante toda sua vida, uma computação em grade é composta de um grupo dinâmico de processos que funcionam em recursos e locais diferentes.
- Os processos que constituem uma computação em grade podem se comunicar usando uma variedade de mecanismos.
- É possível que seja necessário a utilização de mecanismos de autenticação diferentes para acessos aos recursos espalhados por locais distintos, que por sua vez podem ter diferentes políticas de autorização. Na figura 3.1, isto é exemplificado, quando os computadores que disponibilizam recursos acessam parâmetros de controle de acesso de usuário e aplicações em servidores localizados em um outro local (F).
- Um usuário pode ser associado a diferentes locais e com diversas credenciais com o propósito de controle de acesso.
- Os recursos utilizados pela computação em grades podem estar situados em diferentes países.

Um dos principais problemas é fornecer soluções de segurança para as situações acima, principalmente para coordenar diversas políticas de controle de acesso e executar operações com segurança em ambientes heterogêneos. Dentro de uma gama de possibilidades de uso das grades computacionais, novas necessidades apareceram e com elas novos problemas surgiram.

Alguns dos principais problemas provenientes dos objetivos de grades ou apresentados em razão de sua arquitetura são:

1. Execução imediata dos processos.
2. Alocação antecipada de recursos.
3. Verificação de uso de recursos da grade.
4. Necessidade de controle dos processos.
5. Serviços de informações de acesso em grades.
6. Parâmetros de segurança.
7. Delegação.
8. Mapeamento de identidades.
9. Firewalls e Redes Virtuais Privadas (VPN)
10. Ataques maliciosos.

1. Execução imediata dos processos

Um primeiro problema a ser analisado é aquele em que um usuário necessita combinar recursos de diferentes locais em um único, coordenando trabalhos para a execução imediata de processos. Por exemplo, um usuário poderia gerar uma grande quantidade de dados de um grande instrumento compartilhado (por exemplo, um microscópio), que então necessitam serem armazenados em um grande disco e que por sua vez pudessem ser alcançados por um potente processador. Uma vez que a análise preliminar dos dados ocorreu, dados intermediários necessitam serem guardados e enviados para outros computadores para analises futuras.

Um recurso específico para as analises pode ser selecionado por um agente que age em nome do usuário baseado em métricas definidas tal como rapidez, disponibilidade, etc... A escolha é feita por um serviço terceirizado, mas o usuário pode especificar um grupo de recursos que deseja escolher. A execução remota de trabalhos, especialmente em locais múltiplos, provavelmente vai requerer a leitura e a escrita dos arquivos em locais remotos. E nesse cenário, para a execução imediata dos trabalhos, surgem alguns problemas de segurança [HUMPHREY, 2001]:

- No caso do cenário acima, necessita-se a identificação dos possíveis recursos para o processamento dos trabalhos em múltiplos locais.

- Além da identificação dos recursos é preciso saber se estes estão propensos a executar os processos naquele instante.
- Para manter a execução imediata dos processos que analisam os dados, é preciso uma busca ininterrupta de recursos.
- A identificação do usuário em um domínio diferente do seu pode ser um outro problema que atrapalhe a execução imediata dos processos.
- Após a identificação do usuário no domínio do recurso, a continuidade da execução dos processos, depende que estes sejam executados como se fossem de um usuário local reconhecido.
- Os processos podem precisar ler ou escrever arquivos em nome do usuário que solicitou a execução dos processos.

2. Alocação antecipada de recursos

Se um grande instrumento gera uma grande quantidade de dados e fluxo para serem processados em tempo real sem dúvida será necessário reservar recursos antecipadamente. Os recursos podem ser espaço para armazenamento de dados, largura de faixa de rede e principalmente ciclos computacionais. Para a alocação antecipada de recursos temos as seguintes necessidades:

- Todo processo de autorização do usuário deve ser realizado antecipadamente.
- Com a antecipação é possível que o recurso alocado previamente não esteja mais disponível quando surgir a necessidade de utilização dos mesmos.
- Dificuldade de estabelecer antecipadamente espaços para armazenamento, largura de faixa de rede e ciclos computacionais em detrimento das prioridades do momento, principalmente quando a requisição é feita em domínios administrativos diferentes.

3. Verificação de uso de recursos da grade

Um administrador local ou um administrador da grade podem necessitar monitorar todos os acessos aos recursos locais ou até mesmo a outros recursos da grade. Estas informações monitoradas podem ser úteis para finalidades de controle de usuários e recursos, para uma revisão

rotineira de segurança, ou para um procedimento em tempo real de detecção de intrusos. Os administradores (local e da grade) podem desejar verificar os acessos permitidos e os acessos rejeitados.

Este cenário implica que a monitoração pelos administradores deve ser capaz de colher informações em diferentes domínios administrativos, ou os diferentes locais devem confiar nos administradores para ceder tais informações e essa troca de informações deve ser realizada com cuidado e segurança, uma vez que informações inexatas podem gerar imagens falsas do que está sendo monitorado e dessa forma fazer com que os administradores tomem ações indevidas prejudicando o bom funcionamento da grade computacional e de seus próprios recursos.

4. Necessidade de controle dos processos

A execução de processos de longa duração de um determinado usuário em domínios diferentes do seu, deve possuir a habilidade de desconectar de uma estação e posteriormente reiniciar em uma posição diferente. Porém nesta situação um usuário pode querer monitorar o progresso do processamento ou então inserir informações a serem processadas em momentos específicos. Monitorar a execução dos processos é bastante simples quando se sabe onde os arquivos de registros estão sendo escritos tendo o acesso para lê-los.

Mas no caso do cenário em que os processos são migrados de um local para outro, sempre que este tiver que ceder o poder de processamento para outros processos mais prioritários, a dificuldade de monitoramento torna-se bem grande uma vez que inúmeras políticas de acesso podem existir durante a troca de recursos realizada pelos processos.

Neste caso, não basta o usuário se identificar ao outro domínio administrativo, mas por questões de segurança, deve ser capaz de se identificar ao próprio processo e por esse e outros motivos que o monitoramento e a inserção de novas entradas de dados durante a execução dos processos de longa duração pode gerar sérios problemas de segurança.

5. Os serviços de informação de acesso em grades

A capacidade de encontrar serviços, de determinar o status e identificar a disponibilidade dos serviços, é crucial para o bom funcionamento da grade. Na maioria das arquiteturas de grades os serviços de informação existem para ser um repositório centralizado de informações. Muitos

serviços requerem acesso às informações que o repositório guarda, como por exemplo o status atual e quem podem usar cada serviço. Os usuários podem ler estas informações diretamente do repositório tendo acesso a informações de outras entidades como máquinas e processos que são monitorados.

Esse acesso se não for controlado pode gerar problemas como alteração de informações de forma indevida, fazendo com que a confiabilidade nas mensagens fique ameaçada. Nesse sentido a confiança deve ser mútua, ou seja, do repositório de informações para o usuário e do usuário para o repositório de informações.

6. Parâmetros de segurança

Há um grande número de parâmetros que podem afetar a segurança na interação de um usuário com serviços e recursos das grades computacionais. Estes parâmetros precisam ser ajustados pelos usuários e pelas partes interessadas. A integridade e a confiabilidade de ambas as partes, além dos dados gerados e armazenados, são controlados pelos parâmetros.

A exata configuração destes parâmetros é essencial para que a integridade e a confiabilidade estejam garantidas. O parâmetro de integridade deve garantir que nada deve ser alterado entre o momento da escrita e o momento da leitura pelas partes interessadas. O parâmetro de confiabilidade deve garantir que ninguém pode compreender os dados, com exceção do escritor e do leitor pretendido.

A configuração incorreta ou a utilização de parâmetros desconhecidos aos softwares que suportam as aplicações em grades podem criar aberturas para que os processos sejam alterados, danificados ou utilizados de forma indevida.

A complexidade da utilização destes parâmetros é enorme devido à heterogeneidade de plataformas, a quantidade de usuários e a diversidade de políticas de controle de acesso.

7. Delegação

Em muitos cenários, o usuário principal não tem autorização para acessar recursos em domínios administrativos diferentes do seu. Por isso é necessário delegar suas necessidades a entidades que possam executá-las. Porém a delegação não é algo simples de ser mensurada no sentido de dimensionar os direitos de acesso.

Para todas as delegações que ocorrem em uma grade computacional, uma das mais cruciais é a determinação dos direitos que devem ser concedidos pelo usuário para a execução dos serviços e às circunstâncias em que tais direitos são válidos.

Delegar direitos demasiadamente pode conduzir ao abuso, e quando delegar com poucos direitos pode impedir que a tarefa seja terminada inesperadamente.

A delegação ideal para a execução de cada tarefa em um ambiente tão heterogêneo é um dos maiores problemas que existem quando falamos em processamento em grades, pelo fato da grande variedade de políticas de controle de acesso e da diversidade de níveis de controle de acessos que envolvem os usuários.

8. Mapeamento das Identidades

Mapear a grade ao usuário com identificação local é uma maneira de permitir que um usuário tenha acesso com um único acesso (*sign-on*), não tendo que se identificar a cada novo recurso acessado. Isso implica em dizer que o administrador local e o administrador da grade concordam em traçar o mapeamento para o usuário a fim de utilizar os recursos sem ter que se identificar a todo instante.

Esse modelo requer algumas implicações como:

- Requerer que os usuários tenham contas de acesso locais em todas as máquinas que pretendem utilizar.
- É possível que seja dado aos usuários mais acesso que eles realmente necessitam.
- É preciso que o administrador local confie seus recursos aos usuários que desejam utilizar os serviços.

9. Firewalls e Redes Virtuais Privadas (VPN)

Os *firewalls* ou VPN entre os usuários e os servidores dos recursos ou entre diferentes servidores da grade apresentam um desafio sério às medidas de segurança.

As grades computacionais que incentivam a busca dinâmica de recursos não se beneficiam da segurança estática proporcionadas pelos *firewalls* e VPN. Os *firewalls* permitem acesso aos usuários nas portas especificadas e isso impediria o dinamismo da busca de recursos. Os processamentos dos serviços na grade podem ser perdidos caso os *firewalls* presentes não

estejam configurados de maneira a autorizar a passagem de usuários em busca de recursos com o intuito de dar continuidade na execução da tarefa e ampliar o tempo de resposta no processamento.

10. Ataques maliciosos

Se fizermos uma pesquisa aprofundada e detalhada em aplicações de grades computacionais é possível vermos que existe uma infinidade de possibilidades para se realizar ataques maliciosas devido a quantidade de recursos, serviços e mecanismos envolvidos. Não é nenhuma pretensão dizer que ataques maliciosos em grades computacionais é assunto para uma nova dissertação de mestrado. Mas para que possamos ter uma visão desses ataques vamos apresentar alguns deles:

- Troca de Favores: No ambiente de grades, uma das idéias é a troca de favores em que uma organização ou usuário cede recursos ociosos e depois, quando necessitar, utiliza os recursos ociosos da organização ou usuário que utilizou seus recursos. Porém nesse cenário podem existir intenções maliciosas, uma vez que as tarefas emitidas para serem executados podem ser simuladas gerando dados de resposta falsificados. Assim uma organização ou usuário utiliza os recursos ociosos do seu "parceiro" e quando chega sua vez de retribuir apenas simula essa ação não gastando ciclos de processamento de seus próprios recursos a favor de outros.
- Desestabilização da Execução dos Processos: Temos os casos em que o atacante pode emitir mensagens aos vários componentes com a intenção de comprometer a execução dos processos a fim de mudar a execução da aplicação para destruir a integridade das informações, por exemplo.

Os ataques não ocorrem apenas de entidades externas, mas podem ocorrer através de entidades mal intencionadas que fazem parte do próprio processo. Esses ataques podem ser realizados de um servidor para os mecanismos responsáveis pela troca de dados e tarefa que chamaremos de agentes, dos agentes para os servidores e ainda podem ocorrer ataques entre os próprios agentes.

- Servidores contra agentes: No caso do ataque dos servidores contra os agentes, os principais problemas de segurança são a personificação, a negação de serviço e a alteração de dados e códigos.
- Agentes contra servidores: Os mesmos tipos de ataques também podem ser causados dos agentes para os servidores além de existir a possibilidade do acesso não autorizado o que possibilita ao mesmo realizar funções que não faziam parte de seu escopo.
- Agentes contra agentes: Um outro tipo de ataque seria os que acontecem de agente para agente, em que um agente pode assumir a posição de outro para atingir objetivos de forma errada.

Dentre os ataques citados a maioria já foi estudado e já existem formas de proteção, porém em quase toda a literatura sobre o assunto é bastante enfatizado como grande problema de segurança os ataques que partem dos servidores mal intencionados contra os agentes.

3.2.2 Cenário 2: Ambientes Filantrópicos

Inúmeros projetos em grade estão neste momento espalhados pelo mundo, o mais popular atualmente e um dos primeiros a se difundir na rede é o *SETI@home* da Universidade de Berkeley. Para o projeto SETI (*Search for Extraterrestial Intelligence*) milhares de pessoas no mundo todo rodam um protetor de tela que analisa os sinais de rádio capturados pelo radiotelescópio de Arecibo. O *SETI@home* atualmente tem conseguido alcançar a marca de 1 zettaflop (um sextilhão de operações aritméticas por segundo). Na prática o usuário que possui um computador, que não necessita estar constantemente ligado à internet, baixa um cliente da grade *SETI@home* compilado para sua plataforma e o instala. O usuário cria uma identificação e recebe neste momento um pequeno pacote contendo os dados que serão processados durante as horas ociosas do computador. Neste momento não é necessário estar conectado, o programa vai processando localmente e colecionando os resultados em um pacote de retorno. Concluída a operação, quando o usuário volta a se conectar na rede, o programa envia o pacote resultado do processamento e solicita um outro pacote continuando a operação *off-line*. Normalmente os requisitos de grades deste tipo são que os pacotes de dados e resposta sejam pequenos para não ocupar a banda de dados dos clientes e que o processamento seja leve para não afetar as tarefas na máquina cliente [SETI, 2004].

Ambientes Filantrópicos são aqueles em que as pessoas, através de seus recursos, concordam em ceder ciclos de processamento ociosos de seus equipamentos em nome de uma determinada pesquisa. A analise destes em um único computador, por mais poderoso que seja, levaria anos. Mas com o auxílio de milhares de computadores espalhados pelo mundo esse processo pode se tornar muito mais rápido.

A livre e espontânea iniciativa dos proprietários de cederem seus ciclos ociosos em nome da ciência têm seus riscos, pois pode deixar suas informações particulares expostas a ataques.

No decorrer do projeto vulnerabilidades foram e estão sendo identificadas tornando milhares de computadores expostos a uma série de problemas de segurança. Como descrito do site do projeto *SETI@home*, uma das mais recentes vulnerabilidades identificadas ocorre da seguinte forma:

- O aplicativo *SETI@home* utiliza o protocolo HTTP para realizar o download de novas massas de dados para serem processadas e também utiliza este protocolo para enviar informações de usuários e registrar novos usuários. Através dessa implementação foram identificadas algumas vulnerabilidades, sendo elas:
    - Toda informação é emitida através de um pacote através da rede. Esta informação inclui o tipo do processador e o sistema operacional da máquina que está executando os processos;
    - A aplicação possui a identificação, o endereço e o caminho do servidor a qual precisa devolver a informação processada, e estas informações ficam expostas nos pacotes que trafegam pela rede;
    - Pacotes cifrados de forma assimétrica, de forma fraca, podendo ter sua chave facilmente quebrada em razão de sua grande disseminação;

Através destas vulnerabilidades, muitos ataques maliciosos passaram a ocorrer:

- Como as informações expostas pelo aplicativo cliente do *SETI@home* são simples e de fácil acesso a uma pessoa maliciosa que planeja um ataque em uma determinada rede é possível realizar uma varredura através destas informações e obter os dados necessários para o ataque;

- O atacante altera as informações trafegadas na rede podendo redirecionar todos os dados processados que deveriam ser enviados a um servidor da aplicação para uma máquina de seu controle;
- Com os dados em mãos o atacante pode alterar o conteúdo processado com a intenção de causar falsa impressão através de dados irreais;

Além desse tipo de ataque, um atacante pode gerar uma enorme quantidade de pacotes com estruturas semelhantes tanto do servidor para as máquinas clientes como das máquinas clientes para os servidores com intenção de desestabilizar os processos que estão sendo executados. Nesta categoria entram alguns ataques classificados como comuns, que é o caso da Negação de Serviços, por exemplo.

3.3 Propostas de Soluções de Segurança

Conseqüência do planejamento, projeção e inúmeras pesquisas que são realizadas voltadas para o desenvolvimento do conceito de grades computacionais, muitas propostas de soluções para os diversos problemas de segurança têm sido oferecidas, e discutidas a seguir.

3.3.1 Cenário 1: Computação distribuída em alta escala

Neste capítulo foram apresentados problemas de segurança específicos em grades. Para esses problemas existem propostas de soluções que não estão necessariamente vinculados a sua forma de implantação. Cada sistema de suporte em grades pode implementar sua solução da melhor maneira que convier, ou seja, Globus utiliza uma técnica, Legion utiliza outra, e assim por diante. A seguir apresentamos algumas propostas de solução para cada um dos problemas específicos citados anteriormente neste capítulo como abaixo:

1. Execução imediata dos processos.
2. Alocação antecipada de recursos.
3. Verificação de uso de recursos da grade.
4. Necessidade de controle dos processos.

5. Serviços de informações de acesso em grades.
6. Parâmetros de segurança.
7. Delegação.
8. Mapeamento de identidades.
9. Firewalls e Redes Virtuais Privadas (VPN)
10. Ataques maliciosos.

1. Execução imediata dos processos

Como forma de agilizar a busca por recursos os ambientes de grades possuem um agendador ou scheduler que interage com os componentes dos serviços de informações da grade.

O scheduler identifica através dos serviços de informação da grade se o recurso encontrado permite que sejam executados serviços do usuário, caso não consiga esta resposta, o scheduler após identificar a máquina busca a permissão diretamente no próprio recurso.

Para que a execução dos processos ocorra de forma imediata, um agente controlado pelo scheduler fica responsável por garantir recursos para que a execução não pare. Assim, se por qualquer motivo, ocorrer de uma execução ter que ser interrompida em um recurso, este poderá reiniciar imediatamente em outro previamente determinado pelo scheduler.

Conforme o agente controlado pelo scheduler pré-agenda um recurso, este deve providenciar a autenticação entre o usuário e o novo recurso antes que o trabalho necessite ser executado na nova máquina.

Com a finalidade de facilitar estes processos, pode ser submetido ao usuário um identificador local para que este seja reconhecido como um usuário local autorizado respeitando os direitos cedidos. Dentro destes direitos, pode ser necessária a autorização de ler e escrever em arquivos remotos [HUMPHREY, 2001].

2. Alocação antecipada de recursos

Para a execução imediata dos processos uma técnica utilizada é a alocação antecipada de recursos deixando os mesmos disponíveis no momento da execução dos trabalhos.

Por um lado tem a necessidade de garantir ao processo que reservou o recurso que este estará disponível quando o mesmo for ser utilizado e por outro lado é preciso possibilitar que a

reserva do recurso seja desfeita caso o mesmo não seja utilizado. Isso é necessário para que o recurso não fique reservado e ao mesmo tempo ocioso aguardando que quem o reservou faça a utilização deste.

O responsável que fez a reserva deve ser capaz de se identificar ao recurso. Isso é necessário, pois a reserva pode ter sido feita em nome de um grupo e aí o usuário que deseja utilizar o recurso deve ser capaz de provar que faz parte deste grupo. A reserva em nível de grupo é feita com o objetivo de maximizar o aproveitamento de um recurso em prol da execução de um determinado trabalho.

Uma maneira de assegurar que o recurso seja utilizado por um membro do grupo é a utilização de bilhetes de reserva que possam ser transferidos entre os membros do grupo. Neste caso o recurso deve poder verificar que a reivindicação foi transferida de forma legal de quem fez a reserva para um reivindicador atual.

No caso da reserva ser feita para um grupo, existe todo um controle através do bilhete de reserva que garante o não repúdio por parte do recurso, ou seja, garante que o recurso não negue que esteja reservado para a execução das tarefas necessárias.

Esse bilhete de reserva deve ser concedido pelo scheduler que faz a busca de recursos e deve ser protegido para que o mesmo não seja utilizado indevidamente. A proteção do bilhete envolve questões de autenticação utilizando assinaturas digitais e criptografia.

De forma geral, quando a reserva é feita em nome de um único usuário o controle é bem mais simples já que a identificação pode ser realizada de forma direta tornando o gerenciamento da reserva mais simples. Já quando a reserva é feita em nome de um grupo o controle deixa de ser trivial.

A reserva quando concedida possui algumas restrições sendo uma delas o tempo de duração da reserva. Isso se faz necessário já que pode ocorrer a reserva e o recurso nunca chegar a ser utilizado. Neste caso a reserva do recurso se expira dentro de um tempo pré-determinado. O recurso torna-se disponível para que qualquer outro processo possa utilizá-lo. O cancelamento da reserva é informado aos serviços de informação da grade computacional para que todos tomem conhecimento.

Pode ocorrer do usuário precisar utilizar o recurso e este não se encontrar mais disponível devido à expiração da reserva. Por isso que quando à expiração da reserva ocorrer, o usuário que

a fez deve ser avisado para que tome as providências novamente solicitando a reserva caso ainda necessite.

Em resumo, a reserva antecipada de recursos requer a delegação de direitos do usuário ao representante denominado scheduler para que o mesmo faça as reservas em nome do usuário.

Esse controle deve garantir que uma reserva de recurso concedida a um usuário deverá estar disponível a quem fez o pedido do recurso e caso não seja possível um gerenciamento deve ser feito para que o usuário tome providências para assegurar um novo recurso quando precisar utilizar.

3. Verificando o uso de recursos da grade

Para que os administradores locais e administradores da grade possam ter condições de monitoramento que resultem em informações precisas e ações eficientes algumas práticas devem ser adotadas.

Essas práticas consistem em controlar basicamente o que foi usado por quem e em que momento. Dessa forma é possível traçar estratégias para otimizar a usabilidade dos recursos da grade melhorando aspectos de acessibilidade e segurança para todos os interessados.

Em geral, este monitoramento se dá através de registros de todos os acessos utilizando-se da identificação original do usuário na época do acesso. Estes registros de acesso em geral são negociados entre os administradores locais e os administradores da grade para que haja um padrão que facilite a identificação. Por questões de segurança os acessos às informações monitoradas devem ser restritos somente aos interessados que necessitam ver as entradas solicitadas e realizadas em seus recursos.

Com esse controle realizado através do monitoramento da utilização dos recursos é possível realizar a detecção em tempo real de intrusos, uma vez que o monitoramento pode reconhecer pedidos estranhos de acesso, se comparar às identificações de acesso já realizadas.

Dessa forma os administradores locais e administradores da grade podem ter um controle mais apurado através do monitoramento podendo tomar ações de forma mais rápida possibilitando maior segurança às informações dos diferentes locais administrativos presentes na grade computacional.

4. Necessidade de controle dos processos

A necessidade de controle dos processos é importante principalmente para os administradores locais e os administradores da grade no sentido de administrar os processos que estão sendo executados.

Como em uma computação local em que processos podem ser terminados forçadamente, no ambiente de grade isso também deve ser possível e para que isso seja viável alguns detalhes devem ser considerados:

O administrador da grade deve ter condições de rastrear e identificar em que máquina o processo está sendo executado. Neste caso o recurso que está executando o processo e é protegido por uma política de acesso local deve permitir que o administrador da grade tenha direitos que possibilitem a monitoração dos processos, inclusive o cancelamento dos mesmos.

No caso do encerramento forçado do processo, o administrador da grade é notificado por um software de monitoração que detecta o local de execução do processo.

Depois de identificado o local de execução do processo, o administrador da grade informa o administrador local da necessidade de cancelamento do processo, qual o processo que precisa ser cancelado e em que máquina ele está sendo executado.

Assim que o administrador local concluir o cancelamento do processo este avisa o administrador da grade.

Para realizar o trabalho de cancelamento e outros, como a inserção de informação para seqüência do processamento em nome de um usuário da grade, os administradores locais devem receber determinados direitos. Desde que os serviços de uma grade computacional são usados por usuários locais e por usuários da grade, não é óbvia a identificação da origem de um processo. Conseqüentemente, o software que monitora a grade deve manter registros de verificação ou ao menos fornecer os meios para que os trabalhos locais possam ser identificados. No exemplo de terminação forçada, geralmente não haverá uma única pessoa que tenha o poder de matar uma computação típica da grade, porque existirão domínios administrativos múltiplos. Por isso, um esforço deve ser feito para que um processo possa ser terminado forçadamente.

5. Os serviços de informação de acesso em grades

Para garantia de qualidade das informações de acesso em grades que são administradas pelos repositórios de serviços de informação, em primeiro lugar deve ser considerada a confiança entre os usuários e os serviços. Para isso é preciso a autenticação que deve ocorrer entre o usuário e os serviços de informação.

Os serviços de informação devem tratar as informações respeitando as políticas de acesso de cada recurso da grade computacional. Dessa forma, se um determinado recurso, através de sua política de acesso diz que apenas determinados serviços para determinados usuários podem ser executados nele, o repositório de informações deve respeitar este controle sendo capaz de informar quem pode ou não acessar tal recursos para executar serviços e quais serviços podem ser executados.

Antes de uma informação ser publicada no repositório de informações, um acordo de confidencialidade segundo as políticas de acesso e a garantia da integridade das mensagens é celebrado entre o repositório e os usuários que solicitam a publicação. Essa celebração é feita através de autenticação mútua.

Quando os serviços de informação requerem a autenticação do usuário, não é estritamente necessário que os serviços de informação autentiquem para leitura. Por exemplo, se um usuário tem subseqüentes acessos a uma determinada informação, o serviço de informação já reconhece o usuário validando o mesmo para a leitura das informações.

O custo da autenticação mútua pode ser grande em termos de processamento, mas por outro lado vai de encontro a garantias de segurança, protegendo o ambiente de informações maliciosas.

No que diz respeito aos serviços de informação que fornecem informações reais para as solicitações é provável que os repositórios e os responsáveis pelas publicações estabeleçam uma política apropriada de acesso para cada cenário. Entretanto um cenário mais geral deve buscar uma política de acesso mais comum para que o papel do repositório de informações que é gerar informações em busca de recursos na grade não seja prejudicado em razão das restrições impostas pelas políticas de acesso.

6. Parâmetros de segurança

Como já dito, há uma grande quantidade de parâmetros que afetam a interação entre usuários e recursos.

Normalmente muitas partes interessadas na utilização da grade desenvolvem e executam APIs (interfaces) e funcionalidades totalmente diferentes umas das outras. Para que essas APIs e funcionalidades funcionem adequadamente em um ambiente de grade será necessário que as relações a qual estão submetidas sejam flexíveis e conhecidas para outros usuários.

As exigências de confiabilidade e integridade são características que não podem ser impostas de modo exclusivo por cada usuário ou servidor. É necessário que os dados sejam protegidos, mas para que o ambiente da grade seja válido é preciso uma intermediação entre as partes envolvidas para que os protocolos de segurança possam ser capazes de se comunicar uma vez que a diversidade de ambientes é enorme.

Em geral, propor o gerenciamento de chaves de autenticação é uma das possibilidades de se obter a interação entre as partes envolvidas respeitando os parâmetros de cada usuário em relação à integridade e confidencialidade e as políticas de acesso.

Esse gerenciamento de chaves deve ser feito através do administrador da grade no momento de transição entre os domínios administrativos ou até mesmo na mudança do esquema de segurança adotado entre as partes.

Um momento em que se utiliza uma grande quantidade de parâmetros de segurança é na identificação da política de acesso empregada para cada recurso. Neste exemplo supõe-se que existe um interprete da política de autorização para cada recurso que possa ser questionado. É esse interprete que irá esclarecer aos interessados, todas as questões de segurança pertinentes ao recurso, informando o que pode ser executado neste recurso e de que forma.

Um usuário pode necessitar determinar seu próprio acesso a um recurso antes de tentar usá-lo, e para isso é necessário saber os direitos de acesso. Dependendo dos direitos de acesso pode ser que um usuário tenha necessidade e precise de autorização para ajustar configurações para a execução de serviços.

Para que todas essas ações possam ser executadas existem parâmetros que precisam estar configurados de forma precisa pelo administrador da grade a fim de determinar um bom funcionamento.

Os parâmetros, além de guardar as regras e políticas de segurança, determinam outros itens relacionados à execução de serviços na grade. Entre esses itens gerenciados através da configuração dos parâmetros podemos destacar o tempo de vida de um processo executando em determinado recurso, o tempo de duração da autorização de um determinado usuário fora de seu ambiente administrativo, quais protocolos devem ser utilizados para a comunicação entre as partes, quais esquemas de autenticação serão empregadas entre as partes para se garantir integridade e confidencialidade das informações, entre outros.

Em fim, é através dos parâmetros que os administradores locais e da grade irão configurar e estabelecer diretivas para que suas informações fiquem sempre seguras e a comunicação seja eficiente.

O mau uso dos parâmetros pode gerar grandes perdas dentro de um cenário de grades computacionais. O uso exagerado dos parâmetros pode impossibilitar que a maior e melhor característica da grade, que é a liberação de recursos ociosos para que processos pesados possam ser otimizados, não ocorra da forma esperada pelas partes envolvidas. Já o uso frouxo dos parâmetros pode fazer com que as informações de maior importância fiquem vulneráveis e expostas a partes da grade resultando em problemas de confiabilidade e integridade.

7. Delegação

A questão da delegação é um outro ponto bastante crítico e problemático de se chegar a um consenso. Muitas vezes a busca de recursos para a execução de uma determinada tarefa deve ser feita através de delegação.

Delegação nada mais é que ordenar a um agente que execute as atividades em seu nome, uma vez que o próprio usuário não tem acesso para isso. Esse agente age em nome do usuário munido de uma procuração (em geral, um certificado proxy) no qual estão estabelecidos as atividades e os direitos necessários para tal execução. Para que o agente delegado responda em nome do usuário este também recebe direitos para que possa agir.

Porém a coerência para a atribuição de tais direitos é um fator complicador na delegação das tarefas, uma vez que direitos em excesso ou a falta deles pode causar problemas no processamento da tarefa.

Para evitar esses problemas é preciso saber qual o conjunto mínimo de direitos requerido para a execução de um determinado trabalho. Neste ponto um dos problemas é saber como os direitos são classificados pelos vários servidores que fazem parte da grade.

Sabendo qual o nível de delegação requerido o usuário realiza a delegação para que um agente possa agir em seu nome. Nesse primeiro passo o usuário procura estabelecer um nível de direitos de seu interesse que seja compatível com a realização das tarefas.

Porém o direito de delegação não está somente com o usuário, pois este pode não ser consciente das relações de confiança entre todos os recursos do sistema. Assim um pedido que a princípio parece legal pode ser rejeitado por um recurso uma vez que não permite determinado direito.

Para que isso não ocorra, no momento da passagem na grade, o domínio que autoriza a ação do agente delegado deve avaliar os direitos consentidos pelo usuário interessado identificando se houve abuso ou não. Além de identificar quais direitos solicitados podem ser delegados ao agente, é especificado um período em que o certificado de delegação é válido, passado este tempo o agente perde o poder de agir em nome do usuário não podendo completar as atividades que a ele foi delegada.

Para que a tarefa não seja encerrada antes do tempo ou não haja abuso na utilização dos recursos, a definição dos direitos e a validade do certificado de delegação são muito importantes.

8. Mapeamento das Identidades

Na prática, a utilização de mapeamento de identidades para que um usuário de um determinado domínio administrativo possa executar tarefas em recursos de um outro domínio administrativo se dá quando existe confiança mútua entre os domínios administrativos. Em geral, isso ocorre quando são ambientes administrativos diferentes, mas de uma mesma organização.

No caso, as ACs emitem certificados com identificadores locais que possibilitam aos usuários executar seus processos como se estivessem em seu próprio domínio administrativo.

Nesse cenário, o principal ponto de segurança está na garantia de reconhecimento e confiança mútua entre os diferentes domínios administrativos através dos seus ACs.

9. Firewalls e Redes Virtuais Privadas (VPN)

Uma das soluções em relação às barreiras de segurança imposta pelos *Firewalls* e VPN que acabam sendo prejudiciais à busca de recurso e execução de processos através do ambiente de grades é a configuração de portas conhecidas que os usuários da grade utilizam para rastrear informações e recursos.

Porém muito cuidado deve ser dado, pois a abertura de portas erradas pode tornar o *Firewall* inoperante possibilitando aberturas que tornem portas de entrada disponíveis para ataques maliciosos.

Dessa forma muitas informações antes protegidas pelo *Firewall*, podem ficar expostas podendo ser roubadas, destruídas, alteradas sem que sejam percebidas.

Um monitoramento deve ser realizado no ambiente a fim de se adequar o *Firewall* ao objetivo da grade computacional sem que a segurança das informações seja prejudicada.

10. Ataques maliciosos

Dentro desta categoria podemos encontrar diversos tipos de ataques maliciosos que certamente irão aumentar com a expansão e difusão do conceito de grades computacionais, principalmente quando envolver questões de negócios financeiros.

Dentro dos casos apresentados, uma das soluções propostas que atende mais de um problema é enviar junto com as tarefas que serão processadas um código binário que seja capaz de identificar se a tarefa foi realizada ou não. Com esse código é possível realizar uma engenharia reversa para identificar se houve ou não fraude na execução da tarefa podendo assim verificar a integridade dos dados produzidos.

Uma outra idéia é utilizar diferentes tipos de algoritmos de criptografia para diferentes tarefas a serem executadas com o objetivo de dificultar a ação maliciosa de simular a execução de determinada tarefa gerando dados falsos. É muito difícil evitar que uma organização "parceira" simule a execução dos serviços, mas com base nas idéias acima é possível identificar tal malícia e passar a desconsiderar esse "parceiro" classificando-o como não confiável.

Os ataques realizados entre entidades do próprio processo é um tanto complicado de ser resolvido, pois a descoberta do ataque só ocorre depois que ele ocorreu. É semelhante aos vírus

de computadores que só podem ser controlados depois de identificado a forma como ele age. Identificando a forma de ação é que se estabelecem meios de proteção.

Para os casos identificados algumas soluções foram inclusão de códigos binários, criação de tabelas de confiança para divulgação da reputação de cada usuário na grade, mecanismos de autenticação e autorização.

3.3.2 Cenário 2: Ambientes Filantrópicos

As aplicações desenvolvidas para fins de filantropia como ocorre com o projeto *SETI@home* após apresentar algumas vulnerabilidades passaram a desenvolver estratégias com o objetivo de se alcançar mais segurança e não perder credibilidade dos usuários que se propuseram a auxiliar nas pesquisas.

Como ocorre com os sistemas operacionais existentes como Windows, Linux, dentre outros, constantemente falhas de segurança são descobertas por hackers a todo instante com o objetivo de realizar ataques dos mais diversos possíveis.

Da mesma forma que a Microsoft lança patchs de correções para diversos problemas encontrados nas versões do Windows, inclusive para os problemas de segurança, o projeto *SETI@home* também desenvolve atualizações para sua aplicação com o intuito de aprimorar e garantir maior segurança.

Segundo o site do projeto *SETI@home*, para os últimos problemas de segurança identificados na transição de informações do projeto *SETI@home*, os pacotes passaram a ser criptografados com algoritmos mais fortes e, além disso, o pacote passou a ter uma identificação digital que garanta a propriedade da informação.

Essa identificação, que é um código binário embutido no pacote, pode garantir que as informações são exatas e confiáveis. Sendo identificada alguma diferença no código binário saberá que ocorreu uma tentativa de ataque e que aquela informação deve ser desprezada, sendo considerada não processada pelo servidor central do projeto.

Toda aplicação possui seu grau de maturidade, e quanto mais usuário e mais tempo, mais estável e seguro estarão já que constantes pesquisas de soluções para problemas de segurança são realizadas pelas empresas que desenvolvem as aplicações de grades computacionais voltadas para questões filantrópicas [SETI, 2004].

3.3.3 Componentes e aspectos adicionais de segurança em grades

Além dos componentes e das tecnologias já citados, há muitos outros componentes de segurança de infra-estrutura que são necessárias para dar segurança a qualquer ambiente computacional incluindo as grades. Nos próximos itens exploraremos alguns aspectos adicionais de segurança e veremos como cabem em uma infra-estrutura de grades computacionais. O Redbook da IBM faz algumas recomendações como a seguir [IBM_REDBOOK, 2003].

Servidor

Qualquer servidor ou estação de trabalho que participarem do ambiente, seja ele grade ou qualquer outro ambiente de computação distribuída, estão propensos a ataques de um hacker externo ou interno através de suas vulnerabilidades. Sabendo isto, é muito importante proteger e isolar todo o computador do ambiente de quaisquer redes ou recursos que não necessitarem do acesso explícito ao mesmo. A maneira mais comum de se isolar ou proteger os computadores de acessos desautorizados podem ser feitas através de políticas e de procedimentos de segurança. Não há nenhum certificado ou *firewalls* mágicos para proteger os computadores dos perigos existentes em um ambiente de computação distribuída, mas o bom senso na utilização dos recursos existentes pode permitir que as informações fiquem seguras. As seguintes áreas dentro do servidor do ambiente devem ser protegidas:

- Uma boa segurança física deve limitar o acesso de qualquer um dentro da sala do servidor, controlando de forma otimizada sua utilização.
- Proteger todos os diretórios dos sistemas de gerenciamento do ambiente distribuído.
- Roubo de certificado digital e da chave privada (é de grande importância que a senha de identificação da chave privada não fique junto com a chave privada).
- Tomar cuidado e identificar aplicações ou processos que podem ser vulneráveis a ataques no servidor do ambiente.
- Toda a modificação dos arquivos de mapeamento do ambiente deve ser analisada.
- Manter sempre atualizado o pacote de segurança do servidor.

Segurança física

A infra-estrutura de segurança de grades computacionais é baseada em fundamentos comuns de segurança. Por isso, práticas de segurança físicas devem ser aplicadas para todos os computadores da grade. O ambiente físico de um sistema também é considerado uma parte da infra-estrutura. Se os servidores forem mantidos em um quarto aberto, não importa como estão seguras as aplicações projetadas ou como os algoritmos são cifrados, pois os serviços do servidor podem ser facilmente interrompidos, sendo simplesmente desligados ou de outra forma qualquer. Conseqüentemente, o acesso físico deve ser controlado e é parte das políticas de segurança que necessitam de definição.

Os servidores devem ficar em salas específicas e fechadas. Todos os acessos devem ser registrados e controlados de modo que somente o pessoal relacionado possa entrar na sala. A fonte de alimentação aos servidores nunca deve ser interrompida. Isto significa que uma fonte de alimentação ininterrupta (UPS) deve ser usada. Um UPS pode funcionar fora da eletricidade por um período prolongado. Em tal caso, os servidores devem ser capazes de guardar os dados automaticamente antes de serem desligados. A entrada da sala deve também ser monitorada para verificar quem entrou. Para máxima segurança, os segmentos de rede onde as máquinas servidoras são instaladas devem ficar separados fisicamente e logicamente do restante da rede. Idealmente, a separação é feita através de um *firewall* que seja transparente somente para tráfego entre infra-estruturas relacionadas.

Segurança dos Sistemas Operacionais

Uma revisão de configurações de arquivos para cada sistema operacional e para os componentes *middleware* com o escopo do projeto determina como cada acesso de usuário deve ser efetivamente autorizado com base em sua política de segurança prevenindo e detectando acessos não autorizados ao mesmo tempo. Para isso é necessário:

- Remover todos os processos desnecessários dos servidores. Se o servidor não necessitar enviar e-mail, por exemplo, este processo deve ser desabilitado.
- Remover todos os usuários ou grupos desnecessários.
- Todos os usuários do ambiente devem usar senhas fortes.

- Manter os servidores e todos os softwares protegidos com a última versão do pacote de segurança.
- Restringir o acesso aos diretórios de controle do servidor.
- Usar o host IDS (detecção de hosts intrusos) para monitorar diretórios importantes.
- Permitir criação de logs e auditorias no servidor.
- Usar sistemas operacionais uniformes quando possível.
- Permitir níveis de restrições em importantes arquivos do servidor.
- Fazer revisões periódicas dos sistemas para assegurar que nada de importante tenha sido alterado.
- Ativar proteção de antivírus.

Detecção de hosts intrusos

Uma opção recomendada para dar mais segurança aos computadores de qualquer ambiente de computação distribuída, inclusive da grade, é investir em um produto de detecção de hosts intrusos (IDS). Como qualquer aplicação pode armazenar dados importantes dentro de uma estação de trabalho local, IDS pode adicionar uma grande defesa para qualquer movimentação de arquivo na estação de trabalho que não deva ocorrer. Se o host com o IDS detectar uma alteração desconhecida no servidor, pode emitir um alerta a uma estação de trabalho que faz a monitoração e alertar os responsáveis pela grade computacional. Uma das funções é recolher e analisar informações de várias áreas de um computador ou de uma rede para identificar as possíveis brechas de segurança, que incluem intrusões (ataques de fora da organização) e mau uso dos recursos (ataques de dentro da organização). Os IDS usam a avaliação da vulnerabilidade, que é uma tecnologia desenvolvida para avaliar a segurança de um sistema computadorizado ou de uma rede, para melhorar sua performance.

As funções de detecção de hosts intrusos incluem:

- Monitorar e analisar as atividades de usuários e do sistema.
- Analisar as configurações do sistema e as possíveis vulnerabilidades.
- Avaliar o sistema e a integridade de arquivos.
- Habilitar reconhecimento de ataques típicos.
- Analisar atividades anormais.
- Identificar violações de políticas de usuários.

Detecção de intrusos na rede

Em alguns casos pode ser bastante significativo o uso de IDS para redes de ambientes computacionais, mas dentro de um ambiente de grade alguns de seus benefícios seriam perdidos devido à criptografia dos dados entre os usuários da grade.

Quando um sistema IDS da rede não pode ver uma parte dos dados significativos por estar cifrado, o IDS pode responder aos eventos com base no encabeçamento do pacote que não é criptografado. O IDS para redes é mais útil para os casos em que os dados não estejam criptografados. O uso de IDS é opcional dentro de uma arquitetura, mas é extremamente recomendado de acordo com boas práticas de segurança.

3.3.4 Políticas e Procedimentos de Segurança

As políticas e os bons procedimentos de segurança são usados para complementar a variedade dos componentes de segurança que constituem uma infra-estrutura de segurança. Em um ambiente de grades computacionais isto também é válido, principalmente para garantir a segurança das informações locais de cada organização. Para ajudar a gerenciar os riscos existentes para a segurança das informações de uma organização participante de uma grade a definição de políticas e bons procedimentos de segurança são fundamentais.

Em primeiro lugar as organizações devem compreender e identificar o quanto estão seguras suas informações com base na infra-estrutura de segurança adotada. Somente então políticas e procedimentos de segurança devem ser incorporados.

Com a diversidade de políticas e procedimentos que podem existir em uma grade, é necessário adotar critérios que permitam identificar e enquadrar um usuário externo da organização que necessite fazer uso de algum recurso desta.

Ao construir um novo ambiente ou ao executar uma nova aplicação sempre ocorrem várias mudanças, e é sempre uma boa prática realizar uma verificação sobre os componentes de segurança. Uma verificação de segurança ajudará a determinar como estas novas mudanças afetarão a segurança total do ambiente e de todas as outras áreas. Isto pode ajudar a uma melhor condução no uso total dos controles de segurança com o objetivo de verificar se estes controles estão corretos dentro do ambiente estabelecido. Uma revisão dos controles de segurança pode

ajudar a verificar melhor como a segurança está trabalhando em relação ao uso de senhas, auditoria, administração do ambiente, enfim monitorando todos os aspectos que possam conduzir o ambiente a um cenário de menor risco.

3.4 Resumo

Neste capítulo foi apresentado um levantamento dos diversos problemas de segurança existentes. Também foi relatada proposta de soluções para os problemas específicos de grades. Após essa abordagem veremos no próximo capítulo uma arquitetura para serviços em grades computacionais.

Capítulo 4
Arquitetura Aberta para Serviços na Grade

As perspectivas apresentadas por grades computacionais abrem um leque de oportunidades voltadas para a prestação de serviços dos mais diversos segmentos possíveis. A integração de *Web Services* com as grades computacionais proporcionará o que chamamos de **Grade de Serviços**.

Dentro deste contexto, ocorrerá a interoperabilidade entre domínios de forma transparente, integrando e compatibilizando as políticas e mecanismos de segurança existentes nas organizações virtuais e reais. Essas condições são conseguidas somente com uma arquitetura de segurança com padrões abertos e esse será o enfoque deste capítulo.

4.1 Desafios de Segurança em um Ambiente de Grade

Segundo as proposições do grupo de trabalho OGSA do Globus Alliance [OGSA, 2002], que trata da arquitetura aberta para serviços na grade, desafios de segurança enfrentados como integração com sistemas e tecnologias, interoperabilidade com diferentes ambientes e relacionamentos de confiança existentes na interação dos ambientes terão que ser superados. Os relacionamentos entre estas três categorias de desafios serão vistos na seqüência.

4.1.1 Desafio de Integração

Uma única tecnologia de segurança não seria capaz de atender aos desafios de segurança da grade e ser adotada em cada um dos diferentes ambientes possíveis. As infra-estruturas existentes de segurança devem ser mantidas e utilizadas. O mesmo deve ocorrer com os mecanismos de autenticação aplicados nestes ambientes que devem ter seus critérios de segurança respeitados.

Uma arquitetura de segurança de grade computacional necessita se integrar com arquiteturas e modelos existentes. Isto significa que a arquitetura deve ser capaz de se adaptar aos

mecanismos existentes de segurança (por exemplo, Kerberos ou ICP) e ser extensível de modo que possa incorporar novos serviços de segurança quando se tornarem disponíveis.

4.1.2 Desafio da Interoperabilidade

A completa execução de um determinado serviço pode ser realizada em diversos domínios, o que introduz a necessidade de se operar em vários níveis estabelecendo o desafio da interoperabilidade:

- No nível de protocolos, são requeridos os mecanismos que permitam que os domínios troquem mensagens.
- No nível de política, é requerido que cada parte envolvida possa especificar o que deseja a fim de alcançar uma conversação segura e que as políticas expressadas por partes diferentes possam ser compreendidas entre si.
- No nível da identidade, são requeridos mecanismos que identifiquem um usuário de um domínio em um outro domínio. Esta exigência vai além da necessidade de definir relacionamentos de confiança e conseguir a padronização entre mecanismos da segurança (por exemplo, dos bilhetes do Kerberos aos certificados X.509). Para toda invocação feita ao domínio, é preciso determinar as identidades e as credenciais.

4.1.3 Desafio das Relações de Confiança

A confiança entre pontos extremos da grade pode ser presumida, baseando em suposições topológicas (por exemplo, VPN), ou explícita, sendo especificados como políticas de acesso e trocas de credenciais. O estabelecimento de confiança pode ser uma atividade única por sessão ou pode ser exigida dinamicamente a cada pedido feito. A natureza dinâmica da grade pode tornar impossível de se estabelecer relacionamentos de confiança antes da execução da aplicação. Os domínios participantes podem ter tecnologias de segurança diferentes em sua infra-estrutura (por exemplo, Kerberos ou ICP) o que torna então necessário realizar os relacionamentos requeridos de confiança através de alguma entidade que possa dar garantia a todos os mecanismos de segurança envolvidos.

Os problemas de relacionamentos de confiança são mais complexos em ambientes de grade já que existe uma grande dinâmica de eventos envolvidos, um grande número de usuários a serem controlados, e um grande volume de transições de serviços.

O acesso controlado aos recursos e aos serviços da VO é claramente um aspecto crítico de um ambiente seguro de grade.

Dada a natureza dinâmica das grades e da escala do ambiente, sérios desafios existem e necessitam constantemente de pesquisas nas áreas de detecção, de análise, e de recuperação de pontos expostos de segurança.

4.1.4 Inter-Dependência entre os desafios de segurança

Em resumo, os desafios de segurança em um ambiente de grade podem ser categorizados em áreas de solução:

- As soluções de integração onde os serviços existentes necessitam evoluírem;
- Soluções de interoperabilidade de modo que os serviços hospedados em organizações virtuais diferentes, com mecanismos e políticas de segurança diferentes possam ser executados;
- As soluções que controlam e reforçam políticas de confiança dentro de um ambiente dinâmico de grade.

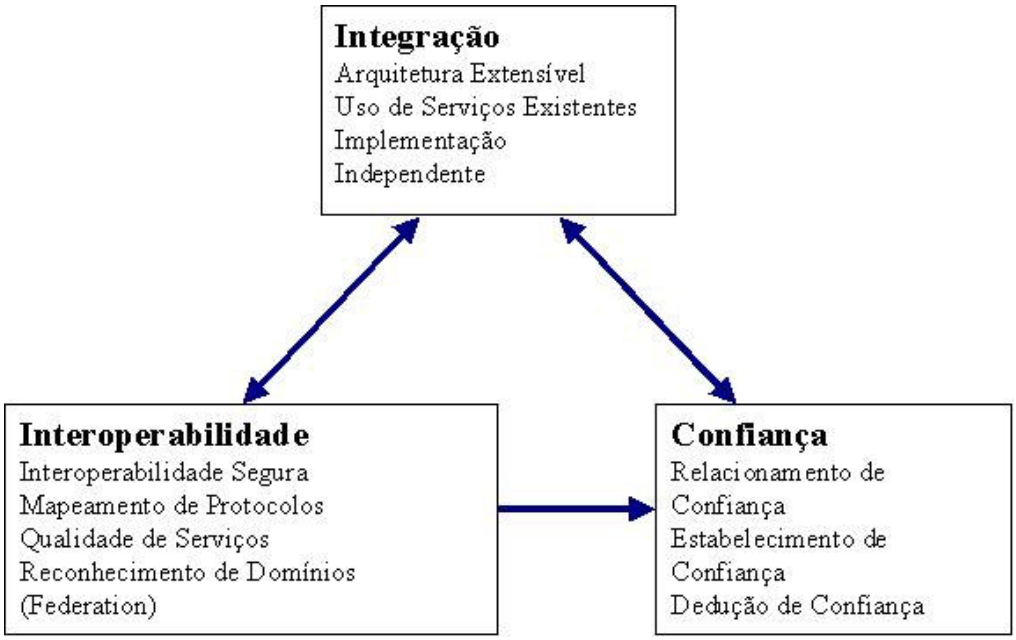


Figura 4.1: Dependências entre os desafios de segurança em um ambiente de grade. Inspirado em [OGSA, 2002].

Uma solução dentro de uma determinada categoria dependerá freqüentemente de uma solução em uma outra categoria. A dependência entre estas três categorias é ilustrada na figura 4.1. Toda a solução para que as entidades consigam a interoperabilidade será dependente dos modelos de confiança definidos dentro dos domínios participantes e do nível de integração dos serviços dentro de um domínio. Definir um modelo de confiança é a base para a interoperabilidade, mas o modelo de confiança é independente de características da interoperabilidade.

4.2 Requisitos de Segurança para Serviços de Grade

Como já vimos, alguns problemas de segurança são específicos a grades computacionais, e é focado nestes problemas que veremos algumas das exigências neste sentido.

O objetivo e a finalidade de tecnologias de grades é suportar, compartilhar e usar de maneira coordenada, os diversos recursos em VOs dinâmicas e distribuídas. As exigências básicas de um modelo de segurança para uma arquitetura aberta de serviços de grades são que os mecanismos de segurança sejam ativados e desativados pelo requisitante do serviço de acordo com a descrição apresentada do serviço.

A segurança da arquitetura OGSA (*Open Grid Services Architeture*), por exemplo, deve ser completa aos usuários da aplicação e dos dados, e permitir o reconhecimento de identidades aos mecanismos de segurança não somente em locais intermediários, mas em todas as plataformas que hospedam os serviços que estão sendo alcançados.

O modelo básico de segurança de OGSA deve contemplar aspectos de segurança como autenticação, delegação, acesso único (*sign-on*), renovação de credenciais, autorização, privacidade, confidencialidade, integridade das mensagens, troca de políticas, acesso seguro, garantia e meios para o gerenciamento da segurança.

A computação em grade está evoluindo para atender exigências de e-business em diversos cenários comerciais. As exigências e as funções discutidas até aqui dão um direcionamento para um padrão de interoperabilidade baseado não somente entre organizações reais dentro de uma mesma VO, mas também através das organizações que pertencem a VOs diferentes.

Com estes fundamentos as aplicações podem ser construídas para estabelecer os relacionamentos de confiança que são requeridos para a computação distribuída dentro de um ambiente de grade.

4.3 Modelo de Segurança OGSA para Serviços de Grade

Do ponto de vista de segurança, a virtualização de serviços define a necessidade de abrangência das exigências de segurança para alcançar tais serviços. A necessidade surge para que componentes de segurança padrão (por exemplo, autenticação, controle de acesso, etc) e maneiras variadas de identificação através de mecanismos diversos de segurança existentes possa ocorrer. Os benefícios de se ter uma maneira mais fraca, neutra, e independente de ligar e de fixar aplicações dentro das organizações, através das empresas, e através da internet são fundamentais para a resolução de problemas em uma arquitetura aberta de grade. Conseqüentemente, a abstração dos componentes de segurança como um único modelo de segurança permite às organizações usar suas tecnologias de segurança ao comunicar-se com outras organizações mesmo que estas utilizem tecnologias diferentes.

Como é evidente nas exigências de segurança da grade, definir serviços seguros é fundamental para os modelos de segurança. Ao fornecer uma infra-estrutura de segurança, um ambiente pode usar as funções e os componentes de segurança, que podem ser expostos como

serviços da grade. Assim, os princípios do modelo de segurança da grade podem ser categorizados como:

- Um modelo de segurança para serviços gerais em grades computacionais;
- Serviços de segurança construídos para fornecer funcionalidades específicas;

4.3.1 Invocação de Segurança

A arquitetura de segurança de serviços da grade deve assegurar-se de que os serviços, quando invocados por um requisitante, aderem às regras do nível de política do ambiente que está hospedando.

Tal política pode incluir um tipo específico de exigências de credencial, de integridade e de confidencialidade, e assim por diante, para a invocação bem sucedida do serviço. Esta arquitetura deve também permitir aos requisitantes de selecionar dinamicamente os serviços que se vinculam com as regras da política, de selecionar um fornecedor de serviço que reúna as melhores exigências dentro das políticas requeridas.

Um serviço de grade deve ser capaz de definir os atributos de segurança que incluem os protocolos e mecanismos de segurança suportados pelo serviço, pelo tipo de credencial esperado do requisitante do serviço, por exigências da integridade, confidencialidade, etc.

Algumas políticas podem requerer que o fornecedor de serviço permita somente a invocação de um serviço depois que o requisitante do serviço tenha se autenticado, e forneça uma credencial apropriada ao invocar o serviço.

Estas exigências destacam a necessidade de estabelecer mecanismos-padrão e reforçar a qualidade da proteção dos atributos de segurança e das políticas de acesso associadas com os serviços e os requisitantes.

4.3.2 Componentes de Segurança

Grandes esforços em torno de *Web Services* (WS) estão sendo feitos, para tornar a arquitetura OGSA hábil a integrar e interoperar soluções. O modelo de segurança proposto para a

segurança dos *Web Services* suporta infra-estruturas de chaves públicas (ICP) e de mecanismos de autenticação baseados no *Kerberos*, e pode ser estendido para aceitar mecanismos adicionais de segurança.

A segurança de um ambiente de grade deve verificar vários aspectos envolvidos em uma invocação de serviço. Isto é ilustrado na figura 4.2.

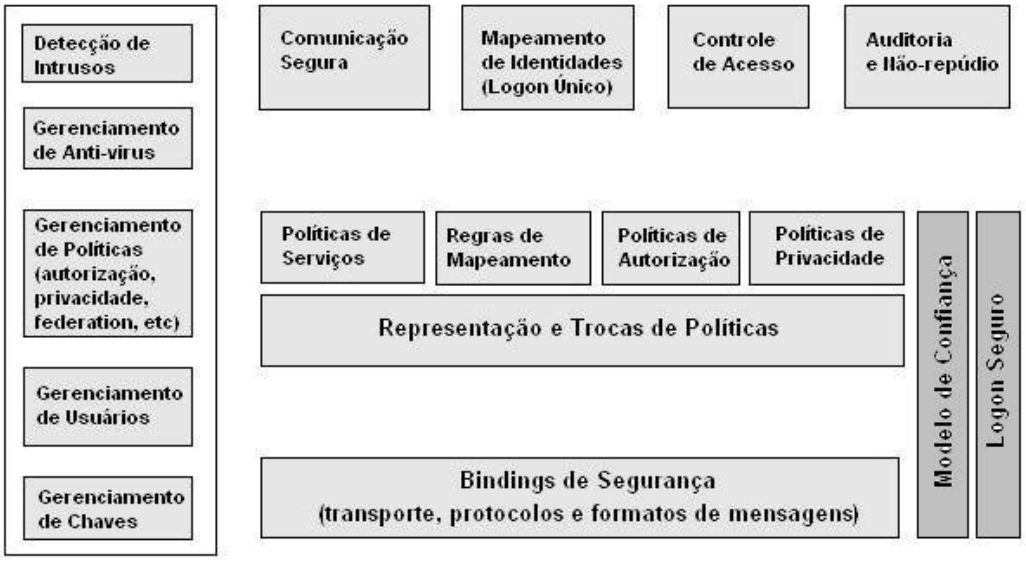


Figura 4.2: Componentes do modelo de segurança da grade. Inspirado em [OGSA, 2002].

O acesso a serviços Web é feito usando uma variedade de protocolos e de formatos de mensagens, como definido por seus *bindings,* que descrevem os formatos de protocolo e de mensagens e devem fornecer suporte para a qualidade do serviço, incluindo requisitos de segurança como a confidencialidade, a integridade, e a autenticação.

Cada participante da grade pode aplicar a política que deseja ao engajar em uma conversação segura com um outro ponto da grade. As políticas podem especificar mecanismos que suportam autenticação, integridade e a confidencialidade requerida, as políticas de confiança, as políticas de privacidade, e outras características de segurança. Dada a natureza dinâmica da invocação dos serviços da grade, os pontos de extremidade freqüentemente vão descobrir as políticas de um serviço desejado e estabelecer relacionamentos de confiança dinamicamente.

Uma vez que um requisitante de serviço e um fornecedor de serviço determinem e reconheçam as políticas adotadas, eles podem estabelecer um canal seguro por onde as operações subseqüentes possam ser invocadas. Tal canal deve reforçar várias qualidades de serviços

incluindo identificação, confidencialidade, e integridade. O modelo de segurança deve fornecer um mecanismo para que as credenciais de autenticação do domínio do requisitante possam ser traduzidas ao domínio do fornecedor do serviço e vice-versa. Esta tradução é requerida para avaliação mútua das políticas de acesso baseadas nas apresentadas estabelecidas e na qualidade do canal estabelecido.

4.3.3 Binding de Segurança

A segurança de um *binding* (formatos de protocolos e mecanismos de segurança) é baseada nas características de segurança do protocolo associado e do formato da mensagem. Se novos protocolos ou novos formatos de mensagem forem introduzidos, um cuidado deve ser tomado às exigências de segurança de modo que, no mínimo, a autenticação, a integridade, e a confidencialidade apropriados possam ser obtidas.

O HTTP é um protocolo importante a se considerar por causa de sua transparência aos *firewalls* e a sua adoção. No caso de *bindings* sobre HTTP, os pedidos podem ser emitidos sobre SSL (isto é, HTTPs) e assim o SSL pode fornecer a autenticação, a integridade e a confidencialidade. Entretanto o SSL assegura estas qualidades somente entre participantes de pontos extremos da conexão. Se um pedido necessitar atravessar múltiplos intermediários (*firewalls*, *proxies*, etc...), então a segurança ponta a ponta necessitara ser reforçada por uma outra camada acima do protocolo SSL.

No caso de mensagens de SOAP (*Simple Object Access Protocol*), a informação de segurança pode ser carregada dentro da própria mensagem do SOAP no formulário de segurança definido na especificação de segurança. As mensagens do SOAP podem também ser integras e confiáveis sendo protegidas usando como suporte a assinatura de *XML Digital* e de criptografia de XML respectivamente. As assinaturas e criptografias de bindings definidos em *WS-Security* também podem ser usadas para esta finalidade.

Existem documentos de padronização para a construção e integração de *Web Services* que devem ser respeitados por todas as tecnologias que pretendem interoperar com estes. Dentro destes documentos existem padrões que descrevem aspectos de segurança (*WS-Security*), de autenticação (*WS-Autentication*) e assim por diante para cada mecanismo e componente.

O modelo de segurança da grade deve ser capaz de adotar um nível de segurança independente dos protocolos ou formatos da mensagem. As exigências de segurança para um acesso a um *Web Services* serão especificadas e cumpridas com base no conjunto de políticas associadas aos participantes existentes. Por exemplo, uma política associada com um *Web Services* pode especificar que as mensagens do SOAP serão assinadas e criptografadas.

Dirigindo-se às exigências de segurança dos bindings relacionadas à integridade e a confidencialidade das mensagens, maiores facilidades e segurança em ações como delegação e na transposição de *firewalls* serão conquistados.

4.3.4 Representação e Trocas de Políticas

Os *Web Services* têm determinadas exigências que devem ser apresentadas para que haja interação com eles. Por exemplo, um serviço pode suportar formatos de mensagem específica ou pode requerer credenciais próprias de segurança para executar uma determinada ação. Um ambiente que hospeda serviços tem o acesso associado às políticas de um serviço do *Web Services* de modo que possa reforçar as exigências de invocação quando o serviço é alcançado. É importante para os requisitantes do serviço conhecer as políticas associadas com o serviço desejado. Uma vez que o requisitante do serviço conhece as exigências e as potencialidades suportadas de um serviço desejado, este pode avaliar as potencialidades e os mecanismos que o fornecedor de serviço suporta. Dessa forma, o requisitante do serviço e o fornecedor de serviço selecionam junto um conjunto ideal de *bindings* para conversar um com o outro. Em um ambiente dinâmico como o de grades computacionais, é importante para os requisitantes do serviço descobrir dinamicamente estas políticas e fazer decisões em tempo real.

Além das políticas do fornecedor de serviço que necessitam ser exposta a um requisitante do serviço ou vice-versa, podem existir outras políticas que um requisitante do serviço ou um ambiente de fornecedor de serviço necessitam saber, mas não necessariamente expõe as informações a fim de garantir um ambiente seguro.

Baseado nos documentos padrões e guias dos *Web Services*, o *WS-Policy* descreve como os fornecedores de serviços e os requisitantes de serviços podem especificar suas exigências e potencialidades.

4.3.5 Associação de Segurança

Para que as mensagens sejam trocadas com segurança, as políticas podem requerer que o requisitante de serviços e os fornecedores de serviços se autentiquem. Um mecanismo é requerido de modo que possam executar a autenticação e estabelecer um contexto de segurança. Este contexto de segurança pode ser usado para proteger a troca de mensagens subseqüentes. Como um benefício adicionado, usando o contexto de segurança estabelecido, o desempenho de trocas de mensagens seguras será melhor. O período de tempo em que um contexto é usado é considerada uma associação ou uma sessão entre os pontos da interação. O estabelecimento e a manutenção do contexto de segurança devem ser baseados em um contexto do *Web Service* definido dentro de serviços web ou especificações dos serviços da grade.

A noção de contexto é fortemente ligada com os bindings. Muitos protocolos existentes (por exemplo, IPSEC, SSL, IIOP) e os mecanismos (por exemplo, Kerberos) já suportam contextos seguros de associação. Os *WS-SecureConversation* descrevem como um serviço de *Web Service* pode autenticar mensagens do requisitante do serviço, como os requisitantes dos serviços podem autenticar fornecedores de serviço, e como estabelecer contextos mutuamente autenticados de segurança.

Facilitar a associação segura ajuda a estabelecer a identidade de um requisitante ao fornecedor de serviço e vice-versa de modo que o fornecedor do serviço e o requisitante possam satisfazer às exigências de autenticação da identidade e reforçar as políticas de autorização e de privacidade baseadas na identidade estabelecida. As identidades dos requisitantes e dos fornecedores de serviços são requeridas de acordo com as finalidades, de modo que os registros de verificação contenham a informação sobre a identidade de acesso.

4.3.6 Mapeamento de Identidades e Credenciais

Um ambiente de grade computacional consiste em domínios múltiplos de confiança e segurança. As operações entre entidades em domínios diferentes requerem tipicamente a autenticação mútua. A identidade de requisitante do serviço e fornecedores, assim como suas respectivas credenciais, muitas vezes não podem ser reconhecidas pelo outro domínio. Permitir a

interoperação requer o reconhecimento dos domínios envolvidos e seus respectivos mecanismos de segurança, por exemplo, Kerberos ou ICP.

O reconhecimento dos domínios envolvidos é denominado de *federation* e será realizado tipicamente no mapeamento ou na tradução das identidades e das credenciais que é requerida através dos *proxies* ou dos intermediários confiáveis. Os componentes de mapeamento e tradução nesta camada são responsáveis por implementar estas funções conforme as políticas correspondentes. A estrutura resultante do *federation* dá base para as exigências de delegação e autenticação única.

O *WS-Federation* definirá como construir cenários de confiança usando as especificações de *WS-Security*, de *WS-Policy*, de *WS-Trust* e de *WS-SecureConversation*. O modelo de segurança da grade deve executar o *federation* de acordo com a especificação do *WS-Federation*.

4.3.7 Autorização

As políticas requeridas no modelo de segurança da grade incluem também políticas de autorização. A autorização é uma parte chave de um modelo de segurança e requer uma menção especial. Cada domínio terá tipicamente seu próprio serviço de autorização para tomar suas próprias decisões de acesso. Em um ambiente de Internet, a autorização é associada tipicamente com um fornecedor de serviço que controla o acesso a um recurso baseado na identidade do requisitante do serviço. Os clientes, ou os requisitantes do serviço confiam no servidor, ou no fornecedor de serviço. A autenticação do fornecedor de serviços através do SSL é um mecanismo para estabelecer a confiança do requisitante do serviço para o fornecedor de serviço. Em um ambiente de grade, regras mais restritas aplicam-se do lado do requisitante do serviço. Os requisitantes do serviço avaliam seu relacionamento com o ambiente do fornecedor de serviço antes de decidir se confiam no fornecedor de serviço para assegurar o pedido.

A implementação de mecanismos de autorização em cada domínio pode seguir modelos diferentes (por exemplo, autorização baseada em necessidades, autorização baseada em regras, potencialidades, listas do controle de acesso, etc...). O guia *WS-Authorization* descreve como as políticas de acesso para um serviço de *Web Services* são especificadas e controladas. Cada domínio poderá ter seus próprios modelos de autorização, autoridades de autorização e

facilidades de gerência. Definindo um modelo de autorização será fornecido um ambiente seguro controlando o acesso aos serviços da grade.

A autorização baseada em identidade é típica na maioria dos gerenciamentos de recursos. É necessário que toda identidade exibida por um requisitante de serviço seja reconhecida e válida no domínio de fornecedor de serviço, facilitando a identificação e credenciando funções mapeadas.

Existem circunstâncias onde um usuário pode querer permanecer anônimo, ou usar uma identidade (possivelmente compartilhada) diferente. Uma identidade apresentada pode ser associada a um conjunto de atributos, privilégios ou direitos que podem ser avaliados no sentido de tomar decisões de acesso, não importando se a identidade é ou não de um usuário real.

4.3.8 Privacidade

Manter o anonimato ou a confidencialidade da informação são importantes em determinados serviços do ambiente. As organizações que criam, controlam, e usam serviços de grades computacionais necessitarão com freqüência indicar suas políticas de privacidade e requerer que os pedidos façam reivindicações sobre a adesão do fornecedor de serviço a estas políticas. A especificação de *WS-Privacy* descreverá um modelo para detalhar como a privacidade pode ser encaixada nas descrições da *WS-Policy*. O modelo de segurança da grade deve adotar a *WS-Privacy* além da *WS-Policy* para reforçar políticas de privacidade em um ambiente de grade.

A imposição das políticas como o *federation*, a autorização e a privacidade dos serviços devem ser construídas em cima de *WS-SecureConversation*, de *WS-Federation*, de *WS-Authorization* e de *WS-Privacy* na arquitetura de segurança dos *Web Services*.

4.3.9 Confiança

Cada membro de uma VO provavelmente tem uma infra-estrutura de segurança que incluí serviços de autenticação, registros de usuários, mecanismos de autorização, proteção de camadas de rede e outros serviços de segurança. As políticas de segurança, as credenciais de autenticação

e as identidades que pertencem à organização de um determinado membro são controladas, emitidas e definidas dentro do espaço da organização - isto é, dentro do domínio de segurança. A fim de processar de forma segura pedidos que transitam entre membros de uma VO, é necessário para as organizações o estabelecimento de confiança entre os relacionamentos. Tais relacionamentos de confiança são essenciais para que os serviços alcançados entre os membros possam atravessar pontos de verificação da rede (por exemplo, *firewalls*) e satisfazer às políticas de autorização associadas aos serviços conseguidas por credenciais traduzidas de um domínio ao outro (por exemplo, Kerberos a ICP) e mapeando identidades através dos domínios de segurança. Conseqüentemente, definir e estabelecer estes relacionamentos de confiança em um ambiente de grade é fundamental no modelo de segurança. Tal modelo necessita definir relacionamentos diretos ou mútuos de confiança entre dois domínios, assim como os relacionamentos indiretos de confiança são conseguidos através de intermediários. Estes relacionamentos são definidos como regras para mapear identidades e credenciais entre os domínios envolvidos da organização.

O modelo de confiança da grade deve ser baseado na especificação *WS-Confiance* dos *Web Services*. Devido à natureza dinâmica das grades, relacionamentos de confiança podem também necessitar ser estabelecido dinamicamente usando os *proxies* de confiança que agem como intermediários. A confiança pode ser estabelecida e reforçada com base em políticas de confiança definidas a priori ou dinamicamente. Uma vez que tal modelo é definido, este executará um roteiro que demonstra como as afirmações de confiança devem ser utilizadas por um fornecedor de serviço ou por um requisitante conforme as circunstâncias. O modelo dará forma também à base para satisfazer às exigências de acesso único a uma sessão baseado na confiança de afirmar a autoridade ou na confiança do pedido feito pelo membro de uma VO.

4.3.10 Gerenciamento de Segurança

O modelo da segurança da grade agrupa todas as funções de gerência de segurança aplicáveis aos vários aspectos de *binding*, de política e de *federation*. Estes incluem a gerência para funções criptográficas, a gerência do registro do usuário, a autorização, a gerência da política da privacidade e confiança e a gerência de mapeamento de regras que permitem o *federation*. Pode também incluir a gerência de detecção de intrusão, dos serviços de antivírus e de

informações de garantia permitindo requisitantes de serviços de descobrir que mecanismos e garantias de segurança um ambiente hospedeiro pode oferecer.

O gerenciamento de segurança deve estar dirigido a vários aspectos de infra-estrutura de segurança que satisfará às exigências no ambiente de grades computacionais.

4.4 Integração com Padrões de Segurança

De acordo com o documento OGSA, o ambiente e as tecnologias da grade integram os serviços com os recursos existentes do domínio que será executado o serviço. Como visto no modelo de segurança da grade, o ambiente de grades computacionais possui estrutura flexível e que maximiza investimentos existentes na infra-estrutura de segurança. Permite o uso de tecnologias existentes tais como os certificados de chave pública "X.509", bilhetes (*tickets*) de compartilhamento de segredos Kerberos e outros mais. Conseqüentemente, é importante para a arquitetura de segurança adotar e suportar qualquer padrão existente que seja relevante. Muitos serviços da grade são baseados em *Web Services*, e por isso o modelo de segurança da grade adota padrões de segurança dos *Web Services* para suas funcionalidades.

Dado que OGSA é um serviço orientado baseado na arquitetura de *Web Services*, o modelo de segurança de OGSA necessita ser consistente com o modelo de segurança dos *Web Services*.

A figura 4.3 ilustra as camadas de tecnologias e dos padrões que existem hoje e de como são adequadas aos modelos de segurança da grade.

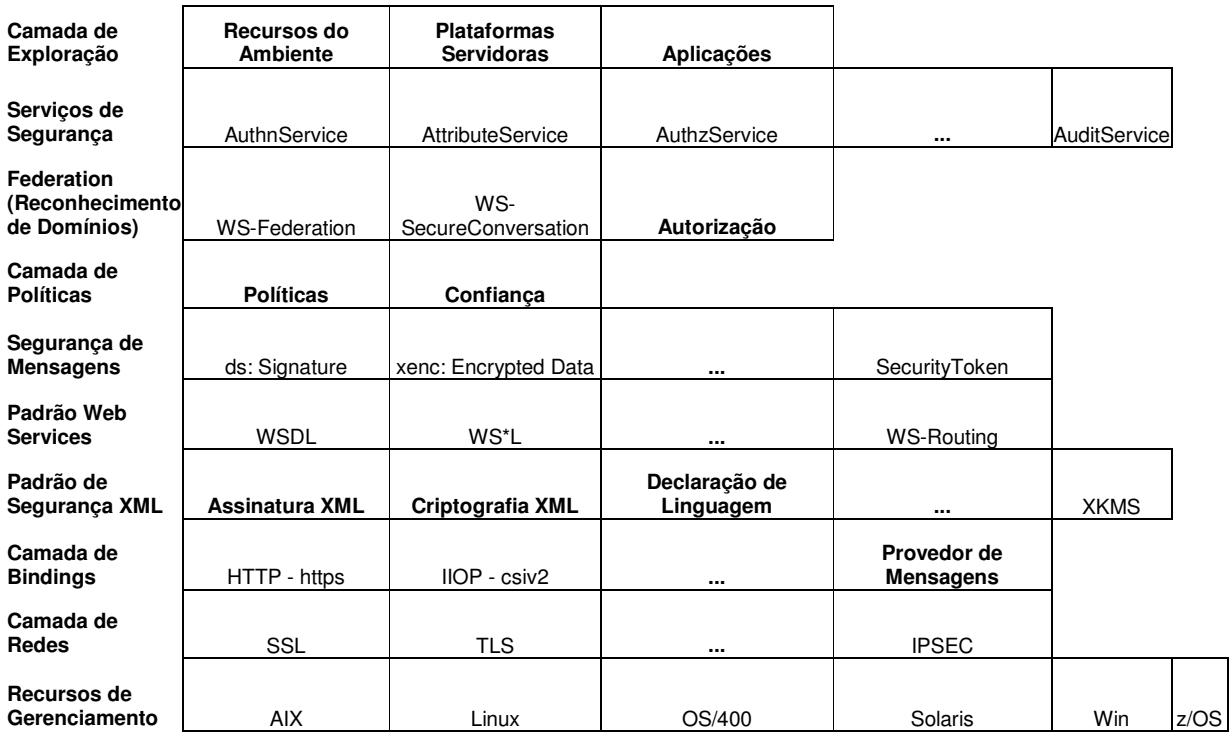


Figura 4.3: Camadas de tecnologias para a arquitetura de segurança da grade [OGSA, 2002].

Em um ambiente orientado a serviços, cada mecanismo de decisão deve ser construído como um serviço. Estes serviços são hospedados em um ambiente da grade, sendo que o próprio ambiente necessita usar tecnologias de segurança e soluções. Há um conjunto limitado de apoio a confiança em que o sistema se baseia para a construção dos serviços. Tais apoios de confiança podem ser autoridades do certificado, domínios confiados do Kerberos, autoridades de autorização ou um conjunto mínimo das partes de segurança que são anexados ao ambiente que hospeda.

A segurança pode ser uma parte inerente de uma rede e de uma camada obrigatória. No exemplo da camada de rede, protocolos como IPSec, SSL ou TLS é que fornecem a segurança. No caso da camada obrigatória, pode ser fornecida por HTTPs, por exemplo.

O nível de segurança das mensagens fornece meios de conseguir manter a segurança ponta a ponta em vez de depender de tecnologias de segurança subjacentes de parte a parte como o SSL. No caso do SOAP, a segurança é baseada na *WS-Security* e nas áreas que se dirige: assinatura digital, criptografia e símbolos de segurança. Como descritas no modelo de segurança

da grade, a camada de política e a camada de *federation* serão construídas baseadas nas camadas e nas tecnologias subjacentes de segurança.

Os serviços de segurança podem ser construídos com base em tecnologias e nas soluções fundamentais. Estes serviços podem ser construídos para um caso específico ou alternativamente utilizar funcionalidades existentes de segurança expostas como um serviço de autorização. Tais serviços de segurança podem ser apresentados como *Web Services* que prestam serviços de manutenção a eles mesmos - de modo que possam ser descobertos, direcionados e invocados.

Diferentes ambientes necessitarão realizar a interoperabilidade, assim o uso de diferentes tecnologias em ambientes hospedeiros deve ser compreendido como parte da política de modo que a interoperabilidade possa ser alcançada.

4.5 Segurança como Serviço

Para realizar a integração e a interoperabilidade ao assegurar os serviços da grade, as tecnologias de segurança existentes podem ser usadas e reusadas. Soluções de segurança existentes são expostas como serviços, assim a construção de novas funcionalidades de segurança requerida será concebida como serviços em um nível de abstração dando maior flexibilidade às questões de segurança no ambiente de grades computacionais. Como todo serviço, os serviços de segurança devem ser apresentados conforme *Web Services* e devem expor as funcionalidades enquanto oculta detalhes da implementação. Os serviços devem ter os aspectos de segurança baseados no modelo de segurança da grade (por exemplo, um pedido de serviço deve ser protegido usando o *WS-Security*) e assim serem protegidos usando uma variedade de mecanismos de segurança suportados em ambientes hospedeiros que serão adotados.

Uma infra-estrutura de OGSA pode usar um grupo de funções primitivas de segurança do seu próprio escopo de serviços. Um conjunto de serviços de segurança da grade pode incluir serviço de autenticação, mapeamento da identidade do serviço, serviço de autorização, serviço de política da VO, serviço de conversão da credencial, serviço de auditoria, serviço de perfil (profile), serviço de privacidade e outros.

4.6 Exemplo de Uso

Esta seção discute uma aplicação de grade que utiliza os padrões apresentados para a realização de prestação de serviços. Neste exemplo um fluxo de segurança ponta a ponta do requisitante do serviço a um fornecedor de serviço é apresentado mostrando como os mecanismos de segurança são empregados em uma aplicação de grade computacional.

4.6.1 Típicos Negócios em Serviços de Grade

Bob é um viajante que utiliza a web para procurar as melhores taxas de hospedagem de hotéis através de uma agência de viagem, chamada de "Pacote de Viagem". A fim de pedir serviços de viagem para a agência, que inclui reservas automáticas de hotéis com base em critérios fornecidos pelo usuário, Bob abre uma conta de cliente com a agência de viagem. Isto é realizado pela invocação de um serviço que chamaremos de "RegistraNovoUsuario", que deve fornecer um símbolo de autorização de pagamento (por exemplo, número de cartão do crédito) e os critérios de seleção do hotel. O símbolo de autorização de pagamento tem um tempo de vida limitado, que permite a agência trabalhar em nome de Bob por um determinado período. Bob fornece os critérios que incluem as datas programadas de chegada e de partida, escalas de preço que está disposto a pagar pelo quarto, e outras informações de seu interesse.

Como mostrado na figura 4.4, Bob submete o pedido sobre uma conexão amigável de protocolo do *firewall* (por exemplo, HTTPs). Bob, como o requisitante do serviço, confia em sua conexão à agência de viagem baseada na política de acesso estabelecida previamente. O serviço de "RegistraNovoUsuario" hospedado pelo departamento de vendas por sua vez invoca com segurança o serviço de criação de novos usuários que chamaremos de "CriaNovoUsuario" do departamento de contabilidade. A base de confiança para a invocação é o relacionamento de confiança estabelecido através do mecanismo de autenticação do Kerberos usado por ambas as organizações (departamento de vendas e departamento de contabilidade) na agência "Pacote de Viagem" [OGSA, 2002].

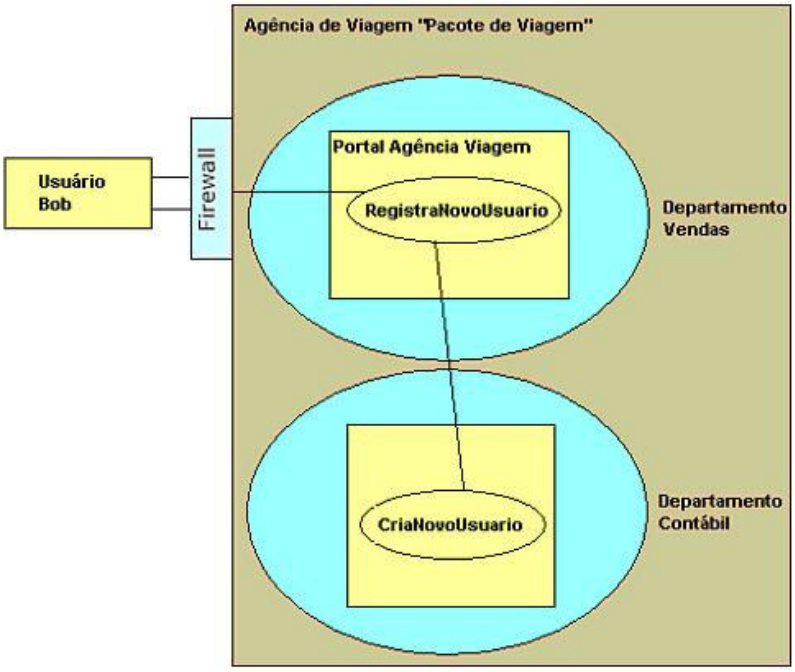


Figura 4.4: Prestação de serviços dentro de uma organização virtual. Inspirado em [OGSA, 2002].

Como parte do processo da criação do cliente, o serviço de "CriaNovoUsuario" avalia o símbolo de autorização fornecido. Para as transações futuras que envolvem a reserva de quartos do hotel, Bob fornecerá no pedido do serviço apenas uma autorização, mais os critérios requeridos para realizar a reserva do hotel.

Os departamentos de vendas e de contabilidade da agência de viagem compartilham de recursos e de serviços a fim de executar com eficácia seu negócio. Mesmo que possam ter os limites organizacionais refletidos através dos domínios da rede ou de segurança, dão forma a uma organização virtual baseada em seu relacionamento de confiança que compartilham seus recursos. Este relacionamento de confiança é definido estaticamente usando o Kerberos como o mecanismo que estabelece a confiança entre estas duas organizações na agência. Entretanto, entre Bob e a agência de viagem, um relacionamento dinâmico de confiança foi estabelecido. Esta confiança é baseada em um símbolo de autorização - uma credencial com tempo de vida limitada, que é delegada à agência de viagem que permite a esta reservar um quarto de hotel em seu interesse. Nesta interação entre Bob e a agência de viagem uma organização virtual foi criada dinamicamente para que limites organizacionais fossem cruzados.

Uma vez que o símbolo de autorização é validado com sucesso, a agência "Pacote de Viagem" processa os critérios e seleciona um hotel que se enquadre aos critérios estabelecidos.

Como ilustrado na figura 4.5, o serviço de "LivroReservaHoteis", que é uma lista de reserva de hotéis, hospedado pela agência de viagem, examina dinamicamente um serviço de reserva de hotel, que chamaremos de "ReservaHotel", a fim de concluir o pedido. Entre um conjunto possível de valores resultantes do serviço de reserva de hotel, o serviço de "LivroReservaHoteis" escolhe um deles, que é o "Grande Hotel", baseado em um conjunto de regras da política que podem ter sido refinados pela união dos critérios que Bob forneceu e as políticas específicas da agência de viagem.

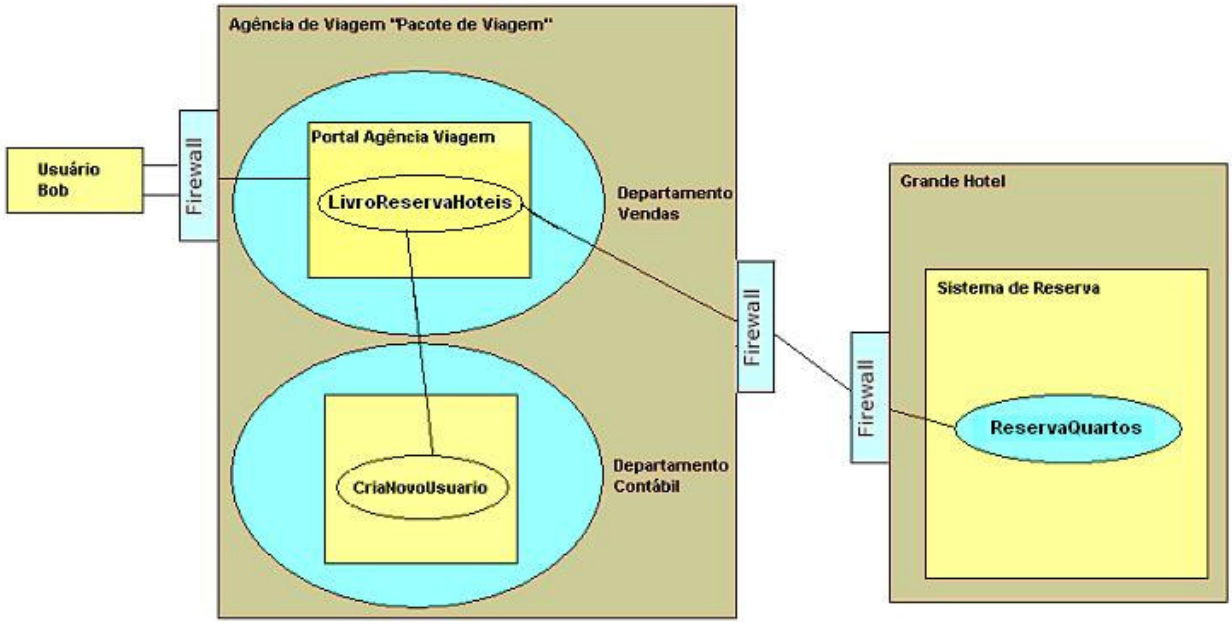


Figura 4.5: Prestação de serviços através das organizações virtuais. Inspirado em [OGSA, 2002].

Os serviços do "Grande Hotel" são assegurados usando uma infra-estrutura de chave pública (ICP). Os serviços podem aceitar os certificados X509 emitidos por uma Autoridade Certificadora confiável a fim de permitir que seus serviços possam ser alcançados. Estas informações são publicadas juntamente com as políticas de segurança associadas com o serviço de reserva de quartos.

A fim de invocar os serviços do "Grande Hotel", um certificado X509 é trocado por um *ticket* (bilhete) do Kerberos da agência de viagem. O pedido de reserva do quarto do hotel é submetido usando o certificado X509 emitido pelo serviço de troca, e inclui com o pedido o símbolo da autorização de pagamento fornecido por Bob em seu pedido inicial à agência de viagem. Tendo a validação bem sucedida do certificado e a política de autorização imposta, o

serviço "ReservaHotel" faz uma reserva de quarto para Bob. O resultado é emitido de volta ao usuário com o serviço de "LivroResrvaHotel".

A agência de viagem e a cadeia de hotéis têm formado sua própria organização virtual onde compartilha de recursos e realiza a prestação de serviços. As diferenças em mecanismos de segurança e os modelos de confiança entre estas VOs e a invocação dinâmica ilustrados neste caso destacam um conjunto de desafios de segurança a serem inseridos no contexto de ambiente da grade.

4.6.2 Cenário envolvendo Intermediários

A Figura 4.6 ilustra um fluxo de um pedido de usuário a um serviço determinado onde o pedido passa através de intermediários para atingir seu destino. Suponha que um usuário deseja invocar um serviço da grade, por exemplo, através de um nó intermediário que resulte eventualmente em alcançar algum recurso em algum fornecedor de serviço remoto. (lado direito da figura 4.6).

O usuário pode obter uma credencial autenticada por um serviço de autenticação existente em seu domínio local, e apresenta a credencial como parte do serviço solicitado. Quando o pedido segue uma rota através de um *gateway*, este pode consultar um servidor de atributos para obter os atributos e os direitos de privilégios do usuário.

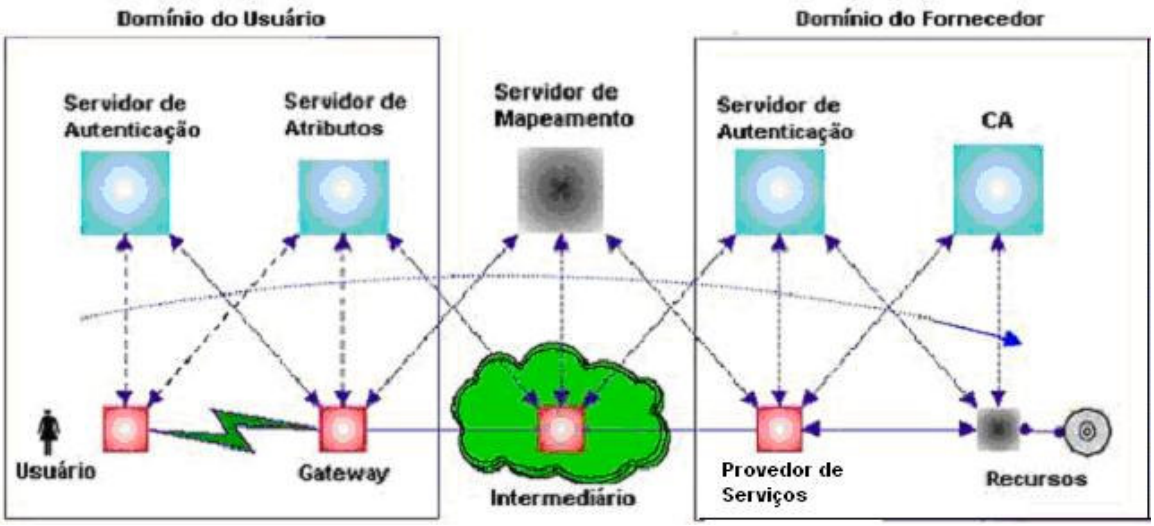


Figura 4.6: Prestação de serviços com o emprego de intermediários. Inspirado em [OGSA, 2002].

O pedido é distribuído através de algum intermediário, que pode converter as solicitações em um formulário compreendido pelo domínio do recurso objetivado (por exemplo, baseado em *WS-Federation*). O intermediário pode converter a credencial de autenticação (por exemplo, bilhete do Kerberos) em um formulário credenciado que alveje o domínio podendo trabalhar com, por exemplo, certificado X509. Adicionalmente, o intermediário pode honrar um conjunto de políticas no encaminhamento do pedido, incluindo as regras mapeadas e as políticas de delegação.

Quando o pedido é recebido pelo domínio alvo este pode validar o certificado. Tendo a validação bem sucedida, pode derivar uma identidade baseada no certificado e fazer decisões de autorização usando as políticas locais definidas na autorização. Este exemplo ilustra o valor de examinar o problema a partir de uma credencial de autenticação em um domínio, de delegar através da passagem do intermediário mapeando um usuário no domínio alvo, através da decisão de autorização feita baseada no mapeamento da credencial executada por um intermediário.

4.7 Resumo

Grades computacionais apresentam uma grande complexidade na integração de domínios e criação de organizações virtuais para o estabelecimento de prestação de serviços.

Neste capítulo podemos ver que para se alcançar essa integração de forma segura existem inúmeros desafios e exigências para a implantação de serviços de grades.

De forma geral, os desafios estão em manter e utilizar padrões abertos de arquitetura respeitando os modelos de segurança para os serviços de grade.

Seguindo com o trabalho veremos como ocorre o emprego de infra-estruturas de chaves públicas em grades computacionais. Esse capítulo faz uma complementação aos capítulos 3 e 4 mostrando o funcionamento de algumas entidades de segurança citadas até aqui.

Capítulo 5

Emprego de Infra-estruturas de Chaves Públicas (ICP) em Grades

O emprego de infra-estrutura de chaves públicas ocorre na maioria dos relacionamentos entre domínios diferentes. Neste capítulo iremos abordar questões como certificação digital em grades computacionais, depois serão explicados aspectos de confiança e delegação que também utilizam infra-estrutura de chaves públicas para garantir ambientes seguros e para finalizar demonstraremos como os sistemas de suporte Globus e Legion tratam de forma concreta estes requisitos de segurança.

5.1 Grades e Certificação Digital

Em grades computacionais muitos mecanismos de segurança são empregados com o propósito de dar segurança ao ambiente. Alguns desses mecanismos foram citados em diversas partes deste trabalho e serão explicados em detalhes mostrando como são utilizados no ambiente de grades [IBM_REDBOOK, 2003].

5.1.1 Autoridade Certificadora (AC)

A Autoridade Certificadora é uma instituição que pode ser pública ou privada, com estrutura hierárquica responsável pela emissão de certificados digitais visando identificar pessoas, usuários, sistemas, redes, equipamentos, etc, para proporcionar um relacionamento de confiança entre sessões de comunicação na rede. O papel principal da AC é determinar políticas e procedimentos que orientam o uso dos certificados por todo o sistema. A Autoridade Certificadora tem outras atribuições conforme lista abaixo:

- Manter a mais rígida segurança possível para a chave privada da AC.
- Assegurar que seu próprio certificado seja amplamente distribuído.

- Emissão de Certificados.
- Revogação de Certificados.
- Emissão de Lista de Certificados Revogados.
- Publicação de Lista de Certificados Revogados.
- Disponibilizar a situação do certificado quando requerida.
- Gerencia de chaves criptográficas.
- Publicação de suas regras operacionais
- Fiscalização do cumprimento desta política pelos usuários.

Antes que uma AC possa assinar certificados para outros, tem que se fazer a mesma coisa de modo que sua identidade possa ser representada por seu próprio certificado. Isso significa que uma AC tem que fazer o seguinte:

- A AC gera aleatoriamente seu próprio par de chaves.
- A AC protege sua chave privada.
- A AC cria seu próprio certificado.
- A AC assina seu certificado com sua chave privada.

Se um recurso da grade necessitar se comunicar com um outro recurso da grade, necessita de um certificado assinado por uma AC. Assim, uma função básica de uma AC é criar e emitir certificados para recursos da grade computacional.

5.1.2 A chave privada das ACs

A chave privada das ACs é uma das partes mais importantes na infra-estrutura de chaves pública (ICP). É usada, por exemplo, para a AC assinar cada certificado digital emitido dentro da rede da grade computacional. Por isso, é altamente suscetível aos ataques dos hackers. Se qualquer hacker conseguir acessar à chave privada das ACs, este pode personalizar qualquer um dentro do ambiente de grade. Conseqüentemente, é muito importante proteger esta chave. Sabemos como a chave privada é sensível para o ambiente de grades, sendo assim é muito importante fornecer para a AC todas as medidas de segurança disponível. Isto inclui a restrição do acesso físico e remoto e da monitoração aos serviços de uma AC.

5.1.3 Certificação Cruzada da AC

Geralmente dentro de um único ambiente de grade a AC fornecerá certificado a um grupo fixo de usuários. Se duas companhias ou organizações virtuais necessitarem se comunicar e confiar um no outro, é necessária a confiança das ACs de ambos os lados ou a participação na certificação cruzada (transversal).

Por exemplo, Alice, uma funcionária que pertence a uma organização com sua própria AC, pode precisar executar um trabalho no computador de Mike que pertença a uma outra AC ou que está fora da organização. Para possibilitar a realização de tal trabalho os seguintes passos devem ser realizados:

- Alice e Mike necessitam obter de alguma maneira um certificado de chave pública um do outro.
- Mike necessita estar certo de que pode confiar na Autoridade Certificadora de Alice. Alice necessita estar certa de que pode confiar na Autoridade Certificadora de Mike.

Grades computacionais de diferentes domínios de segurança ou VOs precisam confiar um no certificado do outro, para isso as regras e os relacionamentos entre ACs devem ser definidos. A finalidade de criar tais relacionamentos de confiança é conseguir uma ICP global, interoperável e ampliar eventualmente a infra-estrutura da grade. Uma vez que o relacionamento é estabelecido, ambas ACs podem ser configuradas para trabalhar com o sistema da grade.

5.1.4 Certificados Digitais

Os certificados digitais são documentos digitais que associam um recurso da grade com sua chave pública específica. Um certificado é uma estrutura de dados que contem uma chave pública e detalhes pertinentes sobre o proprietário da chave. Um certificado é considerado uma contra-prova do ID eletrônico assinado pela AC para o ambiente da grade.

Certificados Digitais fornecem meios de identificar recursos da grade. Um certificado digital pode ser (e deve) distribuído e copiado sem limitação. Os certificados não contêm normalmente nenhuma informação confidencial e sua distribuição livre não cria um risco à segurança. Um importante fato para saber e compreender sobre certificados digitais é que a AC

certifica que a chave pública incluída pertence à entidade identificada no certificado. A implementação técnica é feita de tal forma que é extremamente difícil de alterar qualquer parte de um certificado sem que seja detectado. A assinatura da AC fornece uma verificação da integridade para o certificado digital.

Quando um cliente da grade quer começar uma sessão com um receptor da grade, não é a chave pública que é anexada à mensagem, mas sim o certificado. O receptor recebe a comunicação com o certificado e verifica então a assinatura da AC dentro do certificado. Se a assinatura for assinada por uma AC em que confie, o receptor pode com segurança aceitar que a chave pública contida no certificado é realmente do remetente. Isto impede que alguém use uma chave pública fraudada. O certificado digital contém informações sobre o próprio cliente e sua chave pública. Quando o cliente se comunica com um outro recurso na grade, o receptor usará sua chave pública (contida em seu certificado digital) descrita na sessão ID do SSL, que é usada para cifrar todos os dados transferidos entre computadores da grade.

Um certificado digital é composto de um nome distinto original (DN) e das extensões do certificado que contêm a informação sobre o indivíduo ou o host que está sendo certificado. A figura 5.2 ilustra a estrutura de um certificado digital.

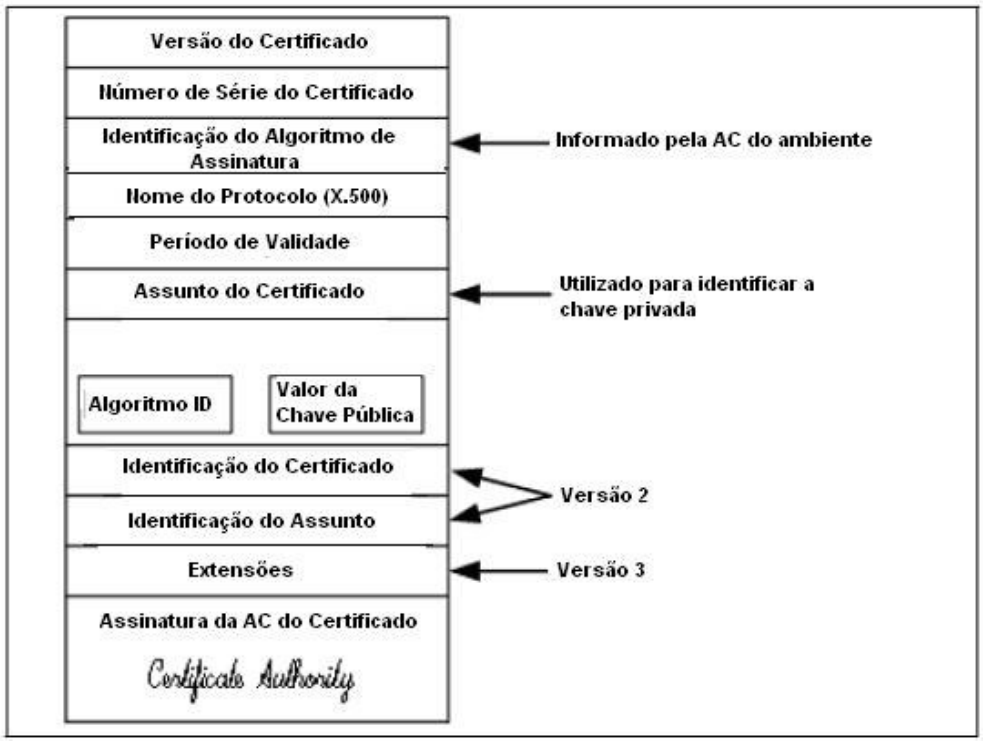


Figura 5.2: Certificado Digital. Inspirado em [IBM_REDBOOK, 2003].

Verificação do usuário

A autenticação dada pela AC através do Certificado Digital pode ser comparada ao processo em que uma autoridade do governo emite um passaporte a um indivíduo. O passaporte serve então como um mecanismo de autenticação quando este indivíduo viaja aos países estrangeiros. Como os passaportes, os certificados digitais podem subseqüentemente ser usado em operações diárias para assuntos que requerem a autenticação de outras partes na grade.

Tipos diferentes de certificados

Há dois tipos diferentes de certificados que são usados dentro de um ambiente de grade computacional. O primeiro tipo de certificado é um certificado do usuário que identifica usuários diferentes na grade. O segundo tipo de certificado é emitido aos servidores da grade.

Usuário

Como um usuário da grade, você necessitará de um certificado para identificar-se dentro dela. Este certificado identificará seu nome de usuário dentro da grade, não seu servidor ou sua estação de trabalho.

Se seu nome fosse Pedro Silva, seu certificado digital poderia ter o nome distinto da seguinte forma: "/O=Grid/O=GridTest/OU=test.domain.com/CN=Pedro Silva”

Servidor

Se você planeja permitir através de ICP rodar programas em seu servidor, você necessitará registrar um certificado de servidor. Este certificado de servidor registrará todas as qualidades de seu domínio no seu certificado. Para seu certificado funcionar, o DNS com todas as qualidades terá que combinar com seu certificado digital. Por exemplo, se o nome do seu servidor fosse "Campinas”, seu certificado de servidor leria: /CN=Service/Campinas.

5.1.5 Autenticação e autorização

Imagine um cenário onde você necessite se comunicar com uma aplicação em um outro domínio da grade. Primeiro você quer garantias de que os dados do host sejam realmente do host. Além de certificar-se de que você pode confiar no host da grade, você quer certificar-se que há confiança entre o host da grade que você quer se comunicar e o seu próprio computador na grade.

Nestes casos, você pode usar uma função de autenticação. Depois que você estiver autenticado com o recurso remoto da grade, você tem então a opção de ter ao seu alcance recursos da grade caso tenha interesse. Para isso, você pode usar uma função de autorização apenas, não sendo necessário se autenticar novamente.

Com as etapas descritas abaixo, você (o host da grade A) autentica e é autorizado pelo host B do outro domínio da grade computacional. Quase todas as etapas são para a autenticação, a não ser a última etapa que é de autorização:

- Envie seu certificado ao outro host (o host B) que você deseja ser autenticado.
- O host B obterá sua chave pública e o conteúdo de seu certificado para seu emprego junto à chave pública da AC.
- O host B cria um número aleatório e devolve a você (host A).
- Se você for utilizar o número, cifre-o com sua chave privada e envie o número cifrado ao host B.
- O host B decifrará o número e certificará que o número decifrado é realmente o que ele emitiu antes. Então o host B autentica que o certificado é realmente seu, porque somente você pode cifrar o número com sua chave privada autorizando o acesso.

Em ambientes de grades computacionais, seu host pode ser um cliente em alguns casos, e em outros casos, um servidor. Conseqüentemente, seu host pode ser requerido a autenticar um outro host e ser autenticado pelo mesmo host ao mesmo tempo. Neste caso, pode ser usada uma autenticação mútua que será detalhada na seqüência.

Brevemente falando, autenticação é o processo de compartilhar chaves públicas de forma segura, e autorização é o processo de mapear um determinado domínio para o acesso de um usuário que já esteja autenticado.

5.1.6 Autenticação Mútua

Para permitir uma comunicação segura dentro da grade, o pacote OpenSSL pode ser utilizado. OpenSSL é um pacote de software que é usado para criar um túnel cifrado usando SSL/TSL entre clientes da grade e servidores. O processo de autenticação mútua começa quando dois recursos da grade querem compartilhar seus recursos. Em vez de usar um repositório de chaves, cada recurso da grade se autentica com o outro com base em seu certificado digital. Por exemplo, um recurso da grade tentará estabelecer uma comunicação segura com um outro recurso da grade. Antes que o receptor permita o acesso do cliente a seus recursos, é necessário autenticar um ao outro. Este processo é conhecido como SSL Handshake.

5.1.7 SSL Handshake

A fim de estabelecer uma comunicação segura entre o servidor da grade e o cliente da grade, um processo *handshake* deve ser estabelecido. Este processo *handshake* é responsável por determinar os ajustes do SSL, trocando chaves públicas e fornecendo a base para processo mútuo de autenticação. O processo do *handshake* é visto a seguir:

1) Um cliente da grade computacional contata um servidor remoto da grade para iniciar uma sessão segura usando um certificado digital de X.509 ID.
2) O cliente da grade emite automaticamente o número de versão do SSL ao servidor, atribuições de criptografia, dados gerados aleatoriamente e outras informações que o servidor necessita para se comunicar com o cliente usando o SSL.
3) O servidor da grade responde, automaticamente emitindo o certificado digital local ao cliente da grade, junto com o número de versão do SSL do servidor, atribuições de criptografia, e assim por diante.
4) O cliente examina a informação contida no certificado do servidor, e verifica que:
    a) O certificado do servidor é válido e tem uma data válida.
    b) A AC que emitiu o certificado do servidor foi assinado por uma AC confiável cujo certificado é reconhecido pelo cliente.
    c) A chave pública emitida pela AC, reconhecida pelo cliente, valida a assinatura digital.

d) O Nome Domínio especificado pelo certificado do servidor combina com o Nome Domínio real também do servidor.

5) Se o servidor puder autenticar com sucesso, o cliente da grade gera uma única "chave de sessão" para cifrar todas as comunicações com o servidor da grade usando criptografia assimétrica.
6) O cliente cifra a chave da sessão própria com a chave pública do servidor de modo que somente este possa ler a chave da sessão. Assim o cliente a envia ao servidor.
7) O servidor decifra a chave da sessão usando sua própria chave privada.
8) O cliente da grade envia uma mensagem ao servidor informando que as mensagens futuras estarão cifradas com a chave da sessão. O servidor da grade envia então uma mensagem ao cliente da grade informando que as mensagens futuras do usuário estarão cifradas com a chave da sessão.
9) Uma sessão SSL segura é estabelecida. O SSL usa então a criptografia simétrica para cifrar e decifrar mensagens dentro do "túnel SSL seguro".
10) Agora que o primeiro recurso da grade foi autenticado, o segundo recurso da grade e os próximos serão autenticados usando o mesmo processo.
11) Uma vez que a sessão está completa, a chave da sessão deve ser eliminada.

Este é um bom exemplo de como a segurança da grade computacional usa criptografia simétrica e assimétrica para autenticar e assegurar a transferência de dados entre recursos da grade. Um cliente da grade usa uma vez a criptografia assimétrica para se autenticar, e então passa a utilizar a criptografia simétrica junto com uma chave secreta compartilhada para cifrar e para decifrar todos os dados transitáveis entre eles.

5.1.8 Infra-estruturas de Chaves Públicas em Grades

Na utilização de ICP dentro de um ambiente de grades é muito importante compreender as funções de uma comunicação. De modo geral, a comunicação é baseada na autenticação mútua de certificados digitais e de SSL/TLS. Os certificados digitais que foram instalados nos computadores da grade fornecem a autenticação mútua entre as duas partes. As funções de SSL/TLS que OpenSSL fornece cifrarão todos os dados transferidos entre hosts da grade. Estas

funções fornecem os serviços de segurança básicos de autenticação e confiabilidade [IBM_REDBOOK, 2003].

De acordo com Lock, et al. [LOCK, 2001], Infra-estruturas de Chaves Públicas (ICP) é um ambiente para prover aos negócios eletrônicos, condições de viabilidade a fim de que tenham os mesmos resultados daqueles conferidos aos contratos fora da rede. Tal recurso viabiliza a autenticação oficial e a integridade do documento, assim como sua elaboração e a confidencialidade nas operações e na assinatura digital, garantindo o valor jurídico e precavendo os envolvidos nas negociações da recusa do que foi firmado anteriormente. Além dos requisitos de segurança, a ICP conta com leis e decretos elaborados pelo governo federal que possibilitam a legitimidade do negócio digital, quebrando definitivamente o paradigma do uso da Internet, por exemplo, para transações financeiras entre empresas e pessoas físicas.

Construir um ambiente de ICP pode fornecer os serviços necessários para projetar uma solução segura da grade e para outras aplicações de computação distribuída. Isto, entretanto, não garante que não haja nenhum risco de segurança. Nós examinaremos algumas vulnerabilidades possíveis que mesmo com todas as atenções dispensadas ainda podem existir, possibilitando verificar cada uma delas para melhorar a segurança do ambiente.

Todas as ferramentas, processos, e políticas de segurança no mundo não garantem completamente um ambiente seguro. Há ainda riscos envolvidos, mas que com a utilização de todos os recursos apresentados podem ser reduzidos a níveis aceitáveis.

5.1.9 Vulnerabilidades da ICP

Apenas pelo fato de se ter um ambiente de ICP não significa que o ambiente esteja completamente seguro. Ainda existem muitas vulnerabilidades que devemos estar cientes. É necessário saber que apesar de toda a infra-estrutura empregada sempre existirá algum risco envolvido.

Dentro de um ambiente de ICP, é sempre necessário estar atento aos locais que armazenam as chaves privadas e estar atento para evitar possíveis roubos de certificados digitais. Segundo Lock, et al. [LOCK, 2001] os seguintes aspectos devem ser considerados em um ambiente de ICP:

- Personificação: Obter um certificado através de meios fraudulentos (usuário ou organização) para poder se passar por estes.
- Roubo de chave privada: O uso desautorizado de uma chave privada associada a um certificado válido.
- Obter chave privada e acordos da AC (Autoridade Certificadora): Usar chaves da AC para assinar certificados fraudulentos ou destruir chaves privadas.
- Decisões de confiança automáticas: as decisões de confiança automatizadas também podem automatizar a fraude.

Como visto, até mesmo uma estrutura provida para gerar segurança pode criar aberturas que desestabilizem o ambiente provocando problemas de segurança.

5.2 Confiança

Estudos realizados por Farag, et al. [FARAG, 2002] dizem que a confiança pode ser definida pela garantia de confiabilidade e respeito que as partes envolvidas terão em relação às informações alheias. Em geral a confiança existe entre entidades pertencentes ao mesmo grupo em que é demonstrado e necessário um interesse mutuo entre as partes. O gerenciamento de confiança é responsável por decidir para qual entidade e quais ações poderão ser tomadas.

Em um ambiente onde as políticas de acesso são descritas em termos de identidade de usuários e atributos requeridos, gerenciamento de confiança consiste em definir os recursos de autorização para identificar os usuários através de assinaturas e a criação de possíveis políticas de acesso.

Em um sistema em que é concedida autorização a todos os usuários através de símbolos de identificação, necessita-se que o gerenciamento de confiança seja chamado a cada ação para que esta seja autorizada. Já em um sistema onde usuários podem delegar um ou todos os seus direitos para que outros usuários realizem ações em seu nome, o controle de delegação faz parte do gerenciamento de confiança.

Em grades computacionais, o gerenciamento de recursos deve ser capaz de realizar buscas de acordo com os seguintes pontos: recursos geograficamente distribuídos, recursos heterogêneos, autonomia administrativa das grades e diferentes controles de acesso.

A busca de recursos não pode simplesmente passar sobre todos esses controles de acesso, mas sim respeitá-los e agir de forma autorizada e para isso é empregado o gerenciamento de confiança que busca estabelecer confiabilidade entre as partes.

5.2.1 Definição de Confiança e Reputação

A noção de confiança é um assunto complexo relacionado a características tais como a confiabilidade, a honestidade, e a competência da entidade confiada.

A literatura em geral não tem uma definição única de confiança para sistemas de computação, mas iremos trabalhar em cima do seguinte pensamento:

"A confiança é uma forte opinião de certeza sobre a competência das entidades que tomará ações não apenas com valores fixos pré-estabelecidos, mas com valores aleatórios que mudam de acordo com o cenário e o momento aplicado" [FARAG, 2002].

O nível de confiança é construído com base em experiências passadas e para um contexto específico. Por exemplo, a entidade Y pode confiar na entidade X para usar seus recursos de armazenamento, mas não executar programas usando estes recursos. O nível de confiança é especificado dentro de um determinado tempo porque o nível de confiança de hoje entre duas entidades não é necessariamente o mesmo nível da confiança de um ano atrás.

Ao tomar decisões baseadas em confiança as entidades podem confiar em outras suas informações específicas. Por exemplo, se a entidade X quiser fazer uma decisão de se usar uma máquina M1 que é desconhecida de X, então X pode confiar na reputação de M1. Para Farag, et al. A definição de reputação para o ambiente de grades computacionais é a seguinte:

"A reputação de uma entidade é uma expectativa de seu comportamento baseado em observações de outras entidades ou de informação sobre a entidade após o comportamento em uma estadia dada" [FARAG, 2002].

5.2.2 Modelo de Confiança para Grades Computacionais

O sistema total de grades é dividido em domínios da grade (GDs). Os GDs são entidades administrativas autônomas que consistem em um conjunto de recursos e de clientes controlados

por uma única autoridade administrativa. Organizando uma grade como uma coleção de GDs, pontos tais como a escalabilidade, a autonomia do local, e a heterogeneidade, podem ser administrados. Nesse modelo, nós associamos dois domínios virtuais com cada GD: (a) um domínio do recurso (RD) para representar os recursos dentro do GD e (b) um domínio do cliente (CD) para representar os usuários dentro do GD. Como RDs e CDs são domínios virtuais dentro dos GDs, alguns casos de RDs e de CDs podem estar dentro do mesmo GD.

Um RD tem os seguintes atributos que são relevantes ao modelo de confiança: (a) posse, (b) grupo de tipo de atividade (TA) que dá a ele sustentações, e (c) confiança em nível (CN) para cada TA. O grupo de TAs determina as funcionalidades fornecidas pelos recursos que são parte do RD. Algumas atividades, como exemplo, que podem compor uma atividade em um RD incluem imprimir, armazenar dados, e usar serviços de exposição. Associar um CN com cada TA fornece a flexibilidade de abrir seletivamente serviços aos clientes.

Similarmente, os CDs têm seus próprios atributos de confiança relevantes ao modelo de confiança. Os atributos para CDs de confiança incluem: (a) a posse, (b) TAs procurada, e (c) CNs associada aos TAs. O campo de TA indica o tipo e o número da atividade que um cliente está solicitando. Os TAs podem ser atômicos ou compostos. Um cliente com um TA atômico requer apenas uma atividade visto que um cliente com um TA composto requer atividades múltiplas.

Domínio do Cliente	**Domínio do Recurso**			
	…	RD1		
	…	CN1		
	…	TA - 1	…	TA - k
CD1	…	CN1-1j	…	CNk-1j
⋮		⋮		
CDi	…	CN1-ij	…	CNk-ij

Tabela 5.1: Exemplo de níveis de confiança [FARAG, 2002].

A tabela acima demonstra um conjunto de níveis de confiança entre o domínio do cliente e o domínio dos recursos.

As entradas na tabela do nível de confiança são quantificadas simetricamente para os relacionamentos de confiança que são assimétricos. Por exemplo, para o relacionamento de

confiança entre o domínio CDi do cliente e o domínio RDj do recurso é definido por f(i; j). Como a confiança é uma função assimétrica o relacionamento reverso entre RDj e CDi, no geral, não é dado pelo f(i; j). Entretanto, nós denotamos o valor atual das duas funções usando um único valor, isto é, ij de CNk para CDi e RDj que acoplam na atividade TA - k. O ij de CNk denota o valor de confiança para uma atividade de um cliente de CDi em um recurso em RDj. Suponha que nós temos o cliente X de CDi que quer acoplar nas atividades TAp, TAq, e TAr no recurso Y em RDj. É possível calcular o nível oferecido de confiança, o ij de CNn entre o X e o Y, isto é, ij = min(CN de CNn para o TAp; TAl para TAq; TAl para TAr). Há dois níveis requeridos da confiança, sendo um do lado do cliente e o outro do lado do recurso.

CN Requisitado	**CN Oferecido**				
	A	B	C	D	E
A	0	0	0	0	0
B	B – A	0	0	0	0
C	C – A	C – B	0	0	0
D	D – A	D – B	D – C	0	0
E	E – A	E – B	E – C	E – C	0

Tabela 5.2: Valores previstos de suporte à confiança. [FARAG, 2002]

Os valores do nível de confiança usam uma escala que vai de nível muito baixo de confiança até um nível de confiança muito elevado que vai de A até E respectivamente.

Os níveis de confiança devem ser mantidos atualizados, caso contrário pode resultar em um processo ineficiente em um sistema muito grande em escala tal como grades computacionais. Este processo é eficiente no modelo de confiança por vários motivos. Primeiramente, como mencionado previamente, nós dividimos o sistema de grades em GDs.

Os recursos e os clientes dentro de um GD herdam os parâmetros associados com os RD e o CD que são associados com o GD. Isto aumenta a escalabilidade para a busca de recursos. Em segundo, a confiança é um atributo que varia lentamente, conseqüentemente, a atualização dos níveis de confiança ocorrem de maneira pouco significativa. Um valor na tabela do nível de confiança é modificado por um valor novo do nível de confiança que seja calculado baseado em

uma quantidade significativa de dados transacionados que justifiquem a alteração dos níveis de confiança.

A figura abaixo (figura 5.1) mostra um diagrama de bloco de um sistema de gerenciamento de recursos baseado nos níveis de confiança. Os CDs e o RDs têm os agentes associados a eles que auxiliam no monitoramento das transações do nível da grade e dão forma às noções de confiança. Estes agentes têm o acesso à tabela do nível de confiança. Se a nova confiança avaliar diferentes valores existentes nas tabelas, os agentes atualizam a tabela. Neste estudo, nós mantemos uma única tabela em um sistema de gerenciamento de recursos centralmente organizado. A tabela pode, entretanto, ser replicada em domínios diferentes para finalidades de leitura.

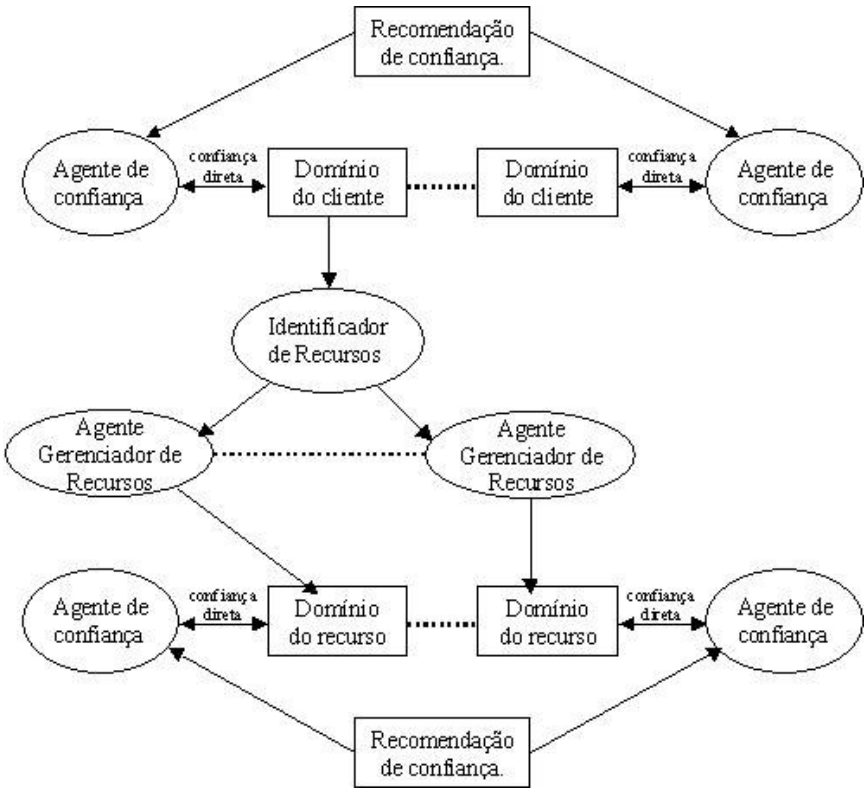


Figura 5.1: Componentes de gerenciamento de confiança da grade. Inspirado em [FARAG, 2002].

Como mostrado na figura 5.1, um agente do CD ou do RD pode estimar a confiança através de canais diretos e de recomendação. O canal direto está estimando a confiança baseada em transações diretas e o canal de recomendação está estimando a confiança baseada na reputação. A recomendação pode ser um conjunto de agentes do CD ou do RD que tiveram

interações precedentes com o domínio de interesse. O agente do CD ou do RD do alvo que recebe a recomendação deve decidir como será atribuído o valor de confiança definindo níveis de confianças para as novas entradas.

O gerenciamento de confiança em um ambiente de grade é considerado um dos principais pontos para a garantia da integridade das informações, uma vez que apenas clientes e recursos confiáveis poderão estabelecer relacionamentos e atividades dentro dos domínios da grade.

5.3 Delegação

Imagine uma situação onde você distribua tarefas para as máquinas remotas da grade e as deixe distribuir sub tarefas a outras máquinas sob sua política da segurança. Se você estiver no lado do host A, você pode criar um proxy (procurador) no lado do host B para delegar sua autoridade. Este procurador age em seu nome, e submete uma requisição em seu interesse. De acordo com documento Redbook elaborado por pesquisadores da IBM Corporation [IBM_REDBOOK, 2003] será explicado o processo de delegação.

As etapas seguintes descrevem o procedimento para criar o procurador (criação do proxy) em uma máquina remota e o procedimento para submeter um requerimento a outro host remoto em seu interesse (ação do proxy).

5.3.1 Criação do proxy

Para a criação do procurador:

- Uma comunicação confiável é criada entre o host A e o host B.
- O host A pede para o host B criar um procurador que delegue sua autoridade.
- O host B cria o certificado de procurador (proxy) conforme requisitado e envia de volta ao host A.
- O host A assina o certificado de procurador (proxy) usando sua chave privada e devolve para o host B.
- Host A envia seu certificado ao host B.

5.3.2 Ação do procurador (proxy)

Para a ação do procurador:

- O procurador envia o certificado do host A e o certificado de procurador ao host C.
- O host C obtém a chave pública do procurador através do seguinte caminho:
    - O host C obtém sua identificação e sua chave pública usando a chave pública da AC.
    - O host C obtém a identificação e a chave pública do procurador através do certificado proxy utilizando sua própria chave pública que acabou de ser obtida.
    - A identificação é um nome distinto similar a "O=Grid/O=Globus/OU=itso.Grid.com/CN=yourname". A identificação do certificado de procurador (proxy) é similar a identificação de seu proprietário "O=Grid/O=Globus/OU=itso.Grid.com/CN=yourname/CN=proxy". A fim de validar o certificado do procurador, o host C apenas tem que se certificar de que a parte principal da identificação do certificado do procurador seja a sua. Se for validado, seu procurador autenticado pelo host C é capaz de agir em interesse do host A.

- O procurador cifra uma requisição de mensagem usando sua chave privada e envia ao host C.
- O host C decifra a mensagem usando a chave pública do procurador e obtém a requisição.

O host C roda o pedido sob a autoridade de um usuário local. O usuário é especificado usando um arquivo de mapeamento, que representa o mapeamento entre os usuários da grade (identificação na grade) e usuários locais (nome local do usuário).

5.4 Globus

Uma das principais ferramentas de suporte existentes hoje em dia para a construção de grades computacionais é o Globus Toolkit. O site da aliança Globus [GLOBUS, 2005] explica as características dessa tecnologia.

O Globus Toolkit é um conjunto de serviços que facilita a computação em grade. Tendo sido a solução de maior impacto na comunidade da computação de alto desempenho, o Globus e os protocolos definidos em sua arquitetura tornaram-se um padrão como infra-estrutura para computação em grade.

A utilização dos serviços do Globus pressupõe a instalação e configuração de uma considerável infra-estrutura de suporte. Cada recurso é gerenciado por uma instância do *Globus Resource Allocation Manager* (GRAM), o responsável por instanciar, monitorar e reportar o estado das tarefas alocadas para o recurso.

As requisições do cliente são recebidas pelo *Gatekeeper* que consulta o *Globus Security Infrastructure* (GSI). Este serviço permite uma autenticação única do usuário na grade. A partir desta autenticação, o GRAM verifica se o usuário pode executar no recurso em questão. Caso o usuário tenha o acesso permitido, é criado um *Job Manager*, que é responsável por iniciar e monitorar a tarefa submetida. As informações sobre o estado da tarefa e do recurso são constantemente reportados ao serviço de informação e diretório do Globus chamado de *MetaComputing Directory Service* (MDS).

Apesar de resolver satisfatoriamente diversas questões do aspecto operacional, o Globus não ataca o problema da grade do ponto de vista gerencial. Utilizando o mecanismo de acesso hoje disponível, são necessárias negociações com os donos de recursos para obter o acesso a estes e há necessidade do mapeamento dos clientes para usuários locais, fatos que limitam a escalabilidade deste mecanismo.

O *Globus Security Infrastructure* (GSI) disponibiliza um conjunto de ferramentas e bibliotecas que possibilitam o acesso seguro de aplicações e usuários aos recursos de uma grade computacional.

5.4.1 Características do GSI

Com o crescimento das grades computacionais, surgiu a necessidade de se pensar em um esquema que permitisse uma comunicação segura entre elementos do ambiente, com suporte à autenticação, incluindo delegação de credenciais para os recursos.

Motivado por isso, a API-GSI, desenvolvida pela Globus Alliance, baseia-se em chaves criptografadas, certificados no padrão X.509, e Secure Sockets Layer (SSL) como protocolo de comunicação.

As características de segurança do GSI estão voltadas principalmente para a autenticação, comunicação segura, autorização e privacidade. Os elementos da grade (usuários, estações e recursos) utilizam certificados, contendo informações relativas à identificação e autenticação, assinados por uma AC.

O GSI oferece duas formas de autenticação, a única e a mútua. Na autenticação única (single sign-on), o usuário autentica-se apenas uma única vez para, a partir de então, utilizar usuários proxies, que podem iniciar procedimentos no sistema e alocar recursos da grade, não importando o domínio administrativo em que ele se encontra.

Tratando-se de autenticação mútua, esta utiliza SSL e envolve a troca de certificados entre os elementos da grade. Para garantir a segurança na comunicação, o GSI implementa um conceito de Assinaturas Digitais, responsáveis pela autenticidade e integridade dos dados de origem. Essa abordagem permite prevenção de bloqueios ao envio de informações (não repúdio). Por se basear em funções Hash, as assinaturas digitais proporcionam uma melhoria na Assinatura Digital Simples, recebendo a entrada de dados encriptados de tamanhos variados e produzindo um tamanho fixo, o que torna a transmissão de informações não só mais segura como também mais eficiente.

Outra característica relevante é a delegação de credenciais, que se dá através da criação de usuários proxies. As chaves privadas e os certificados X.509, criados para cada usuário proxy gerado remotamente dentro da grade, devem ser assinados pela chave original do usuário credenciado.

O certificado X.509 utiliza um protocolo baseado no TLS ou SSL. O certificado X.509 e a chave associada identificam univocamente o utilizador ou o serviço na grade. Para isso é necessário utilizar uma AC em que a organização virtual confie. O protocolo baseado no TLS é

utilizado para verificação, integridade e criptografia dos dados. Os *gateways* são usados para fazer o mapeamento entre a infra-estrutura GSI e os mecanismos locais.

Um conceito central na autenticação GSI é o certificado. Todos os usuários, serviços e hosts na grade são identificados por um certificado que contém informações vitais para a identificação e autenticação dos usuários ou dos serviços. Um certificado é um documento digital que contém a chave pública e os detalhes sobre o proprietário do certificado. Quando um certificado é assinado por uma AC, ele se torna um identificador eletrônico.

Se as duas partes tiverem certificados e confiarem nas ACs que os assinaram, será possível comprovar suas identidades. Antes que a autenticação mútua possa ocorrer, as partes envolvidas deverão confiar primeiramente nas ACs que assinaram seus certificados. Isso significa que elas possuem cópias dos certificados das ACs que contêm a chave pública das ACs às quais devem confiar que os certificados pertençam.

5.5 Grade das Américas - TAGPMA

De acordo com a TAGPMA (The Americas Grid Policy Management Authority) [TAGPMA, 2005], a grade PMA das Américas opera no continente americano orientado pela organização das ACs desta região e por partes de confiança nestas ACs. O objetivo é facilitar a criação de grades neste continente estabelecendo políticas que tragam confiança para toda entidade que fizer parte da grade.

Na Europa e Ásia também existem organizações com o objetivo de desenvolver grades continentais. Desta forma no mundo está se formando 3 grandes vertentes em termos de grade sendo TAGPMA (Américas), EUGridPMA (Europa) e APGridPMA (Ásia Pacífica).

Os escopos principais de uma grade PMA são:

1 - Fornecer certificação para todos os fornecedores de serviços da grade;

2 - A grade PMA TAGPMA funcionará apenas no território conhecido por Américas;

3 – Um dos objetivos dos PMAs é estabelecer confiança entre eles para difundir de forma segura e confiável o conceito de grade global;

4 – O TAGPMA desenvolverá critérios de confiança de acordo com os diferentes tipos de serviços e autenticação;

Em síntese os PMAs serão vistos com uma grande organização gerenciadora das ACs em nível continental que estará cordenando políticas e autorizações que permitam o estabelecimento de confiança para a operação em grades computacionais.

5.6 Legion

O Legion é um exemplo de um sistema de provimento para aplicações em grades que não utiliza ICP para gerar um ambiente seguro.

Segundo os documentos técnicos em grades computacionais do sistema de suporte Legion [LEGION, 2003], o seu modelo de segurança difere muito dos sistemas convencionais. Os programas e objetos do Legion executam em cima do sistema operacional do computador, no espaço do usuário. Dessa maneira, eles estão sujeitos a políticas e controle local do sistema operacional. Um aspecto crítico da segurança do Legion é que a segurança do sistema depende de todos os computadores serem confiáveis; existem vários domínios de confiança, mas estes podem possuir computadores não confiáveis.

O controle de acesso para objetos Legion requer primeiro que o usuário determine a política de segurança de um objeto pela definição de direitos do objeto e chamadas de métodos que ele permite. O controle é executado através de uma função denominada MayI, presente em todo objeto. MayI é como um policial do tráfego: todos chamados de um método para um objeto devem primeiros passar por MayI antes da função desejada ser invocada. Somente se o requisitante possuir os direitos para o método desejado, MayI permitirá que o método invocado prossiga.

Para tornar os direitos disponíveis a um requisitante, o dono do objeto apresenta uma lista de direitos concedidos. Quando o requisitante invoca um método de um objeto, ele apresenta a lista de direitos concedidos a MayI, que checa o escopo e a autenticidade. O dono de um objeto também pode fixar um conjunto de direitos semipermanentes a um requisitante ou grupo em particular. A responsabilidade de MayI é então confirmar a identidade do requisitante e comparar os direitos autorizados com os direitos requisitados para o método chamado. A verificação é também para endereçar o pedido e proteger a comunicação entre os objetos. Todo objeto Legion tem um par de chaves pública e uma identificação única, sendo que a chave é parte do nome do objeto. Objetos podem usar a chave pública de um outro objeto desejado para criptografar sua

comunicação com ele. Da mesma forma, uma chave privada de um objeto poder ser usada para assinar mensagens e fornecer autenticação.

Legion é um dos principais sistemas de provimento para aplicações em grades, mas que possui características muito específicas o que tem dificultado sua evolução. Ao contrário do Legion, Globus possui uma facilidade muito grande de integração com as aplicações por utilizar ICPs e projetar serviços que tornam mais fáceis o desenvolvimento de sistemas para o ambiente de grades computacionais.

5.7 Resumo

Este capítulo destacou o emprego de infra-estruturas de chaves públicas em grades computacionais. As principais operações apresentadas se referem aos controles realizados entre as fronteiras dos domínios administrativos, mostrando como é o controle de confiança entre os participantes da grade. A questão da delegação é um outro aspecto importante, pois permite que direitos sejam oferecidos a entidades, chamadas de proxy (procuradores), para que serviços sejam executados em nome de usuários não pertencentes ao domínio do recurso requerido. Também foram mostrados exemplos de sistemas de suporte para aplicações em grades computacionais (Globus e Legion) no que se refere a características de segurança e como os países e continentes estão se organizando para criar grades computacionais regionais e num futuro atingir nível mundial.

Capítulo 6

Conclusões

A proposta inicial de grades computacionais diz que inicialmente a montagem das grades será feita entre partes com interesses similares, como saúde, biologia, astronomia, física, dentre outros que estarão interligando laboratórios, empresas e universidades e, que tenham interesses semelhantes. Com sua expansão, pode-se chegar, no final, em algo como a formação de uma grade global, uma rede distribuída de colaborações e prestação de serviços entre seus participantes em nível mundial.

Muitas iniciativas de colaboração com interesses similares, principalmente nas áreas de pesquisas cientificas, já existem. Aplicações para grandes organizações também já estão sendo comercializadas, mas se formos pensar nas perspectivas apresentadas para o conceito de grade podemos ver que ainda está ocorrendo uma evolução e muito ainda tem de ser feito.

Considerando as perspectivas de evolução de grades que seria em 3 (três) fases, podemos dizer que estamos apenas na primeira, andando em paralelo com algumas iniciativas de outras fases. Na primeira fase, estão ocorrendo as implementações comerciais de grades de produção por corporações que possuem presença global ou que precisam acessar recursos fora de uma simples corporação local. Na segunda fase em que acontecerão operações de organizações com indústrias similares ou áreas de interesse colaborativo em projetos de objetivo em comum praticamente não existem iniciativas, uma vez que aspectos como concorrência ainda pesa mais do que a colaboração em um projeto único. Mas o ponto extremo que denominamos de terceira fase, é a adoção dos sistemas de grade por parte dos usuários como um modelo utilitário. Para se chegar a esta fase leva-se algum tempo, mas é impossível dizer que isso não será uma realidade se analisarmos todos os esforços que estão sendo feitos.

Muitos grupos (por exemplo, Globus e Legion) já possuem ótimas arquiteturas de grade, com soluções eficientes de segurança, mas que na maioria dos casos absorveram aplicações de pesquisas em que o controle de segurança é mais simples por geralmente envolver um ambiente educacional ou científico, em que os interesses e benefícios são comuns a todos. Além disso,

tecnologias como o *Globus Toolkit* estão sendo adotadas por grandes empresas, como é o caso da IBM, no desenvolvimento de suas aplicações para grades computacionais.

Empresas como IBM, Sun e Oracle possuem iniciativas para introduzir o conceito de grades computacional no mundo empresarial. Algumas empresas começam a operar essa infra-estrutura, porém dentro de um limite em que a própria organização possa ter o controle sobre seus recursos e informações trazendo segurança em sua utilização.

A popularização do conceito de grade computacional levará algum tempo ainda, já que é um tanto recente e até desconhecido da grande maioria de profissionais da própria área de informática.

Algumas iniciativas de utilização de grades através do conceito de grades de serviços, ou seja, as prestações de serviços através da infra-estrutura de grade, começam a acontecer, porém de maneira bastante tímida através de parcerias (por exemplo, serviço de busca de reserva de hotéis, porém em hotéis de um mesmo grupo) que irão beneficiar usuários na obtenção de serviços e trazer ganhos financeiros às empresas que oferecem estes serviços.

Mas quando tratamos de negócios que envolvem valores financeiros, apesar de vislumbrarmos um cenário magnífico em termos de prestação de serviços a diversos segmentos, a questão de segurança é primordial. A confiança nestes ambientes só será conquistada se o ambiente for comprovadamente seguro, tanto do ponto de vista dos usuários para o prestador de serviço, quanto das empresas que estão utilizando a grade para oferecer seus serviços.

A segurança está evoluindo de acordo com o avanço da própria infra-estrutura de grades computacionais. À medida que a amplitude das grades aumenta a sua complexidade, os desafios de segurança também se tornam maiores.

Nas primeiras iniciativas realizadas dentro de instituições científicas ou acadêmicas, em que o foco era a utilização de recursos ociosos para agregar poder computacional e resolver problemas de alta demanda em termos de processamento de dados, podemos dizer que a estabilidade dos mecanismos de segurança é maior e mais tranqüila de ser alcançada já que a própria infra-estrutura de segurança existente é utilizada, sendo apenas integradas as funcionalidades da grade.

Para atingir novos objetivos, principalmente aqueles considerados da fase final, em que chamamos de Grades de Serviços, a confiança depositada no ambiente será conquistada passo a

passo, peça a peça, como se fosse um quebra cabeça que se unirá formando uma grande rede de processamento de informações.

A popularidade de grades computacionais se dará apenas com a construção de aplicações que prestem serviços em nome de pessoas de forma transparente e segura. Um exemplo de aplicação foi citado no trabalho, que é a reserva de hotéis, mas outros serviços como reserva de passagens aéreas, reserva de passagens rodoviárias, são outros casos que ajudariam a tornar popular o conceito de grades.

Os sistemas de suporte a grades estão evoluindo nas questões de segurança. Vimos que técnicas para a resolução de inúmeros problemas são propostas e desenvolvidas com o objetivo de proteger, dar confiabilidade e integridade às informações e processos existentes na grade. Dentro dessas funcionalidades abordamos questões de autenticação, autorização, confiança, delegação e outras que visam tornar o ambiente seguro.

Em casos de aplicações em colaborações cientificas em que existe um conhecimento dos seus limites e que não há interesses sobrepostos, a infra-estrutura de segurança e a arquitetura da grade computacional se mostram confiáveis e eficientes vistos os inúmeros casos citados neste trabalho.

A popularização do conceito vai gerar grandes desafios que necessariamente farão com que pesquisas e estudos sejam realizados com maior objetividade e velocidade para que os problemas, principalmente os de segurança, sejam descobertos e resolvidos antes que grandes prejuízos se espalhem. Este é um dos motivos que induz um crescimento gradual do conceito de grades.

O caso do projeto *Seti@home* mostra que mesmo não envolvendo interesses financeiros, a expansão do projeto se deparou com problemas de segurança nunca previstos. Por esses problemas a quantidade de computadores que disponibilizam seus ciclos de processamento hoje, é bem menor do que na época em que o projeto atingiu seu maior nível de popularidade. Uma das questões responsáveis por isso foi a perda de confiança nos aspectos de segurança da aplicação que colocava em risco informações pessoais dos usuários.

Atingir um nível de maturidade para aplicações em grades para uma escala global é um grande desafio, principalmente para negócios que envolvem concorrência. Sempre que existir concorrência, principalmente para fins financeiros, existirão pessoas mal intencionadas para conseguir seus objetivos desonestamente. Os problemas serão percebidos quando ocorrerem e

deverão ser resolvidos pelas próprias aplicações, porém sempre integradas aos mecanismos de segurança das grades computacionais.

Iniciativas de empresas como IBM, Oracle e Sun, por exemplo, através de seus produtos ajudam que o conceito de grades seja empregado de forma confiável em grandes corporações integrando parceiros como clientes e fornecedores. Nesse cenário, todas as tecnologias de segurança empregadas nas corporações não serão substituídas, mas sim incorporadas para serem usadas dentro das grades e por isso os ganhos com os benefícios de grades computacionais são reais e mais seguros para as corporações do que para usuários comuns.

Quanto à segurança em si, mecanismos estão sendo pesquisados e aprimorados, e em paralelo à expansão do conceito de grades computacionais, as necessidades de segurança também estão evoluindo. Aos poucos empresas começarão a utilizar a infra-estrutura de grades e mais aplicações serão desenvolvidas fortalecendo o conceito até que o ideal almejado por Ian Foster, Carl Kesselman e outros, sejam alcançados ou até mesmo superados.

Como expressado por Ian Foster, daqui a algum tempo a computação em grades será considerada um dos grandes padrões de tecnologia da informação. Apesar dessa evolução estar ocorrendo de forma mais lenta do que o esperado é fato que dentro de alguns anos teremos inúmeras aplicações de serviços sendo executadas dentro de um ambiente de grades computacionais, trazendo grandes benefícios a seus usuários.

As principais contribuições deste trabalho foram:

- Apresentação dos principais conceitos de grades computacionais.
- Levantamento dos principais problemas e desafios de segurança encontrados em um ambiente de grades.
- Demonstração de uma arquitetura aberta para a integração de serviços ao ambiente de grades.
- Detalhamento do funcionamento e emprego de infra-estruturas de chaves públicas em grades computacionais.

O assunto abordado neste trabalho, que faz um levantamento sobre segurança em grades computacionais, possibilita extensões. Dentre elas podemos citar:

- Estudo aprofundado de protocolos e mecanismos de segurança utilizados em grades computacionais.
- Integração de aplicações de prestações de serviço com a infra-estrutura das grades.

De tudo o que foi visto podemos perceber que as maiores dificuldades estão no estabelecimento de confiança entre domínios diferentes, o que em muitos casos está fazendo com que as potencialidades das grades computacionais sejam utilizadas como simples clusters.

Apesar de toda tecnologia empregada é possível perceber que a evolução das grades tem barrado mais em questões culturais e sociais já que é difícil se prover que é possível depositar confiança em toda essa infra-estrutura da noite para o dia.

O TAGPMA para as Américas é um exemplo de iniciativa para começar a resolver estes problemas já que estabelece uma grande estrutura de certificação e que por consequência amplia os níveis de confiança e segurança entre as grades computacionais o que nos fazer acreditar que no futuro não muito distante estaremos utilizando serviços que envolvam vários países de diferentes continentes.

Glossário de Siglas

ASP: *Application Service Providers.*

GRAM: *Globus Resource Allocation Manager*, que submete e controla as tarefas na grade.

HTTP: *HyperText Transfer Protocol* ou Protocolo de Transferência de Hipertexto.

HTTPs: *HiperText Transfer Protocol Secure* ou Protocolo de Transferência de HiperTexto Segura.

IDS: *Intrusion Detection System* ou Sistema de Detecção de Intrusão

IIOP: Protocolo de Transporte utilizado entre CORBA.

IPSEC: *Internet Protocol Security*

J2EE: Versão JAVA para o desenvolvimento e execução de aplicações.

LAN: *Local Area Network* ou Rede Local

MPI: *Message Passing Interface*. Uma biblioteca de rotinas de passagem de mensagens voltada para a padronização e para a portabilidade de programas entre sistemas diferentes.

MPL: *Model Programming Language*.

MPPs: Massively Parallel Processor.

NET (.NET): NET é um *framework* desenvolvido pela Microsoft para rodar aplicações escritas em diferentes linguagens transparentemente, umas para as outras. Este *framework* permite que qualquer aplicação seja executada de qualquer lugar do mundo, em qualquer dispositivo.

NOWs: Um conjunto de estações de trabalho ou PCs, ligado por uma rede local.

SMPs: *Symmetric Multi-Processing* ou multiprocessamento simétrico.

SOAP: *Simple Object Access Protocol* é um protocolo para intercâmbio de mensagens entre programas de computador. SOAP é um dos protocolos usados na criação de Serviços web.

SSL: Secure Socket Layer é um padrão de comunicação, utilizado para permitir a transferência segura de informações através da Internet.

SSP: *Storage Service Providers.*

TCP/IP: *Transmission Control Protocol / Internet Protocol.*

TI: Tecnologia da Informação.

UPS: *Uninterruptible Power Supply.*

VPN: *Virtual Private Network* é uma rede de uso exclusivo dos usuários autorizados por uma empresa, para que se conectem a ela de qualquer lugar do mundo.

WANs: *Wide Area Network* é uma rede de computadores que abrange uma grande área geográfica, com freqüência um país ou continente.

WS: Abreviatura de *Web Services.*

XML: *Extensible Markup Language* é uma linguagem universal para permitir a troca de informações de forma estruturada através da Internet. Permite que os programadores transportem dados de um servidor para outro da rede de forma transparente e organizada.

Referências Bibliográficas

[ALGOS, 2002] Trezentos, Paulo – Grid Computacional: O futuro ou a reinvenção da roda ?: http://paulo.trezentos.gul.pt/artigos/AlgosGrids/GridsHandout.pdf - Acessado em 12/12/2004.

[ANATOMY, 2001] Ian Foster, Ian; Kesselman, Carl; Tuecke, Steven – *The Anatomy of the Grid* http://www.globus.org/research/papers/anatomy.pdf - Acessado em 18/01/2005.

[ASP, 2000] SANS *Institute - Application Service Provider* www.sans.org/resources/policies/Application_Service_Providers.pdf - Acessado em 25/06/2005.

[BAIRD, 2002] Baird, Ian - *Understanding Grid Computacional* – Publicado na *Platform Grid Conference* de 2004 em San Francisco http://www.platform.com/pdfs/whitepapers/understanding_Grid.pdf - Acessado em 18/01/2005.

[BONBONATO, 2004] Bonbonato, Fábio – II Workshop de TI, 2002, Brasília. GRID Computing - Uma Introdução, 2002. http://www.geleira.org/artigos/?redes+Grid_Computacional - Acessado em 12/12/2004.

[CBPF, 2005] CBPF - Projeto GRID do CBPF http://mesonpi.cat.cbpf.br/grid/ - Acessado em 20/06/2005.

[CHIVERS, 2003] Chivers, Howard - *Grid Security: Problems and Potential Solutions* http://www.cs.york.ac.uk/ftpdir/reports/YCS-2003-354.pdf - Acessado em 23/01/2005.

[CIRNE, 2002] Cirne, Walfredo – Grids Computacionais: Arquiteturas, Tecnologias e Aplicações – Apresentado no Terceiro Workshop em Sistemas Computacionais de Alto Desempenho – Vitória – ES de 2002 http://walfredo.dsc.ufcg.edu.br/papers/Grids%20Computacionais-%20WSCAD.pdf - Acessado em 23/01/2005.

[DIMAP, 2003] Galvão, Gabriel e Aquino, Hercules – Grid Computacional http://www.dimap.ufrn.br/~thais/SD20041/Grid.pdf - Acessado em 15/12/2004.

[COLE, 2001] COLE, Eric - *Hackers Beware*. *New Riders Publishing*, 2001.

[EGEE, 2005] EGEE, Projeto – Sobre a computação Grid http://www.infowester.com/cluster.php - Acessado em 25/06/2005.

[EUDTGRID, 2003] European Union Data Grid – Web Site http://www.eu-dataGrid.org/ - Acessado em 05/08/2005.

[FARAG, 2002] Azzedin, Farag e Maheswaran, Muthucumaru - *Towards Trust-Aware Resource Management in Grid Computacional Systems* – Apresentado no *First IEEE International Workshop on Security and Grid Computing* em 2002 www.princeton.edu/~rblee/ELE572Papers/trust_awareGRID.pdf - Acessado em 23/01/2005.

[FOSTER, 2001a] Ian Foster, Ian; Kesselman, Carl; Tuecke, Steven – *The Anatomy of the Grid* – Publicado no Globus Alliance em 2001 www.globus.org/alliance/publications/papers/anatomy.pdf - Acessado em 18/01/2005.

[FOSTER, 2001b] Foster, Ian; Kesselman, Carl; Tsudik, Gene e Tueckel, Gene – *A Security Architecture for*

Computational Grids – Apresentado na *ACM Conference on Computer and Communications Security* de 2003
http://www.princeton.edu/~rblee/ELE572Papers/p83-foster.pdf - Acessado em 23/01/2005.

[FOSTER, 2005] Foster, I – Web Site
http://www-fp.mcs.anl.gov/~foster/ - Acessado em 23/01/2005.

[GDTL, 2005] Guedes, Anne Margareth de Souza e Tecles, José Eduardo T. – Oracle 10g - um Banco de Dados para Computação em Grid
http://www.sqlmagazine.com.br/Mat_Capa_SQL9.asp - Acessado em 18/02/2005.

[GENTZSCH, 2002] Gentzsch, W - *Response To Ian Foster's What Is The Grid?* – Artigo Publicado no Fórum GridToday em 2004
http://www.Gridtoday.com/02/0805/100191.html - Acessado em 23/01/2005.

[GLOBUS, 2005] Globus Toolkit - *Overview of the Grid Security Infrastructure*
http://www.globus.org/security/overview.html - Acessado em 23/01/2005.

[GRIDFORUM, 2005] Grid Fórum - Web Site
http://www.Gridforum.org/ - Acessado em 23/01/2005.

[GRID_IMAGEM, 2005] Adarsh's - Web Site
http://www.adarshpatil.com/pictures/Grid-Computacional.gif - Acessado em 26/12/2005.

[GRIDPORT, 2003] NPACI GridPort Toolkit – Web Site
https://Gridport.npaci.edu/ - Acessado em 05/08/2005.

[GRIPHYN, 2003] GriPhyN, the Grid Physics Network – Web Site
http://www.griphyn.org/ - Acessado em 05/08/2005.

[GUEDES, 2005] Guedes, Anne Margareth de Souza e Tecles, José Eduardo T. – Oracle 10g - um Banco de Dados para Computação em Grid – Matéria de capa da revista SQL Magazine – edição 9 - publicada em 2004.
http://www.sqlmagazine.com.br/Mat_Capa_SQL9.asp - Acessado em 18/02/2005.

[HAGEL, 2002] Hagel, John - *Service Grids: The Missing Link in Web Services*
http://www.johnhagel.com/paper_serviceGrid.pdf - Acessado em 12/06/2005.

[HUMPHREY, 2001] Humphrey, Marty; Thompson, Mary R. - *Security Implications of Typical Grid Computacional Usage Scenarios* – Apresentado no *International Symposium on High Performance Distributed Computing (HPDC)* de 2001 - San Francisco - Califórnia
http://www.cs.virginia.edu/~humphrey/papers/humphrey_security.pdf - Acessado em 23/01/2005.

[IBM, 2004] IBM – Grid Computacional Web Site
http://www-1.ibm.com/Grid/ - Acessado em 18/02/2005.

[IBM_REDBOOK, 2003] IBM RedBook – *Introduction to Grid – Computacional with Grid*
http://www.redbooks.ibm.com/redbooks/pdfs/sg246895.pdf - Acessado em 18/02/2005.

[INTRANET, 2005] Almeida, Marcus Garcia e Rosa, Pricila Cristina
Internet, Intranet e Redes Corporativas – Publicado em 2000 – Editora Brasport

[IVDGL, 2003] International Virtual Data Grid Laboratory – Web Site
http://www.ivdgl.org/ - Acessado em 05/08/2005.

[LEGION, 2003] Legion – Worldwide Virtual Computer – Web Site
http://legion.virginia.edu/index.html - Acessado em 23/11/2005.

[LNCC, 2005] LNCC – Sistema de Processamento de Alto Desempenho – Web Site
http://www.lncc.br/sinapad/ - Acessado em 20/06/2005.

[LOCK, 2001] Lock, Russell e Sommervile, Ian - *Grid Security and its use of X.509 Certificate* – Apresentado no *DIRC Research Conference* 2001
http://www.comp.lancs.ac.uk/Computacional/research/cseg/projects/dirc/papers/Gridpaper.pdf - Acessado em 18/02/2005.

[NCSA, 2003] Welch, Von; Siebenlist, Frank, Foster, Ian; Bresnahan, John; Czajkowski, Karl; Gawor, Jarek; Kesselman, Carl; Meder, Sam; Pearlman, Laura e Tuecke, Steven - *Security for Grid Services*
http://Grid.ncsa.uiuc.edu/papers/GT3-Security-HPDC-Final.pdf - Acessado em 05/03/2005.

[NEESGRID, 2003] Network for Earthquake Engineering Simulation – Web Site
http://www.neesGrid.org/ - Acessado em 05/08/2005.

[NUG30, 2003] Nug30 Solution – Web Site
http://www-unix.mcs.anl.gov/metaneos/nug30/solution.html - Acessado em 05/08/2005.

[OGSA, 2002] Nagaratnam, Nataraj; Janson, Philippe; Dayka, John; Nadalin, Anthony; Siebenlist, Frank; Welch, Von; Foster, Ian; Tuecke, Steve - *The Security Architecture for Open Grid Services*
http://www.cs.virginia.edu/~humphrey/ ogsa-sec-wg/OGSA-SecArch-v1-07192002.pdf - Acessado em 05/03/2005.

[OURGRID, 2005] OurGrid – Web Site
http://dsc.ufcg.edu.br/ourGrid/ - Acessado em 20/06/2005.

[PHYSIOLOGY, 2002] Foster, Ian; Kesselman, Carl; Nick, Jeffrey M.; Tuecke, Steven - *The Physiology of the Grid*
http://www.globus.org/alliance/publications/papers/ogsa.pdf - Acessado em 05/03/2005.

[PPDG, 2003] Particle Physics Data Grid – Web Site
http://www.ppdg.net/ - Acessado em 05/08/2005.

[RNP, 2003] C.Ribeiro, F.Oliveira, J.Oliveira e B.Schulze – Implementação e Desenvolvimentos de Grade Computacional
http://www.rnp.br/_arquivo/wrnp2/2003/idgc01.pdf - Acessado em 12/01/2005.

[SECURITY, 2001] Foster, Ian; Kesselman, Carl; Tsudik, Gene e Tueckel, Gene – *A Security Architecture for Computational Grids*
http://www.princeton.edu/~rblee/ELE572Papers/p83-foster.pdf - Acessado em 23/01/2005.

[SERPRO, 2005] SERPRO – Grid Computacional – Elucidando os Conceitos
http://www.serpro.gov.br/publicacao/tematec/tematec/2005/ttec79 - Acessado em 20/11/2005.

[SETI, 2004] SETI – Search for Extraterrestrial Intelligence – Web Site
http://setiathome.ssl.berkeley.edu/ - Acessado em 14/03/2005.

[SOUTO, 2004] do Souto, Pedro Alexandre Guimarães Lobo Ferreira - Grid Computacional: Sumário
http://paginas.fe.up.pt/~pfs/aulas/pdp2004/at/2Grid.pdf - Acessado em 05/08/2005.

[SRSD, 2003] Sardinha, Luis – *Security for Grid Services*
http://lasige.di.fc.ul.pt/~sardinha/work/Security-for-Grid-Services.pdf - Acessado em 23/01/2005.

[SSP, 2005] SANS *Institute - Storage Service Provider* www.sans.org/resources/policies/Storage_Service_Providers.pdf - Acessado em 25/06/2005.

[TAGPMA, 2005] *TAGPMA, The Americas Grid Policy Management Authority* http://www.gridpma.org/docs/TAGPMA%20charter%20v1.pdf - Acessado em 01/06/2006.

[TAURION, 2004] Taurion, Cezar – Grid Computacional – Um novo paradigma computacional - Editora Brasport, 2004

[TERAGRID, 2003] TeraGrid Project – Web Site http://www.teraGrid.org/ - Acessado em 12/06/2005.

[TRUST, 2002] Azzedin, Farag e Maheswaran, Muthucumaru - *Towards Trust-Aware Resource Management in Grid Computacional Systems* www.princeton.edu/~rblee/ELE572Papers/trust_awareGRID.pdf - Acessado em 23/01/2005.

[WELCH, 2004] Welch, Von - *X.509 Proxy Certificates for Dynamic Delegation* http://www.globus.org/alliance/publications/papers/pki04-welch-proxy-cert-final.pdf - Acessado em 18/02/2005.

[WHITE, 2001] White, Brian; Walker, Michael; Humphrey, Marty e Grimshaw, Andrew - *LegionFS: A Secure and Scalable File System Supporting Cross - Domain High-Performance Applications* – Apresentado no Workshop *Supercomputing 2001* – Denver - Colorado. http://legion.virginia.edu/papers/SC2001.pdf - Acessado em 17/05/2006.