Enterprise
Grid Alliance

Enterprise Grid Security Requirements

Developed by:

Enterprise Grid Alliance
Security Working Group

*** Version 1.0 ***

As approved by EGA Board of Directors

8-July-2005

This Page Intentionally Blank

Copyright Notice

Author Information

Contributor	***Organization***
Mike Beckerle	Ascential Software Corporation
Glenn Brunette	Sun Microsystems, Inc.
Lee Cooper	Oracle Corporation
Wan-yen Hsu	Hewlett-Packard Company
Adam Jacobs	Oracle Corporation
Thomas Keefe	Oracle Corporation
Richard Nicholson	Paremus, Ltd.
Parviz Peiravi	Intel Corporation
Collin Sampson	Sun Microsystems, Inc.
Bob Thome	Oracle Corporation

Revision History

Version	*State*	*Date*	*Comment*
1.0		8 July 2005	Initial v1.0 document for public release

Contents

Figures

Preface

The purpose of this document is to identify the unique security requirements of an *Enterprise Grid Architecture* for end-users, related standards organizations, and vendors alike. It builds on top of the reference model, use cases, and life cycles defined in the Enterprise Grid Alliance ("EGA") Reference Model v1.0 document. As such, the reader is highly encouraged to first read the Enterprise Grid Alliance Reference Model v1.0 document before reading this document.

This document consists of a number of sections:

Introduction

Enterprise Security Foundation

EGA Reference Model Security

Grid Security Use Cases

Grid Security Threats and Risks

Grid Security Requirements

Comments regarding content and format are welcome, and may be directed to the EGA Grid Security working group (ega_gridsecuritywg@mail.gridalliance.org).

This Page Intentionally Blank

1 Introduction

1.1 The Enterprise Grid Alliance

The purpose of the *Enterprise Grid Alliance (EGA)* consortium is to drive the adoption of grid computing and the technologies that enable its deployment and use within enterprise data centers. The EGA is pragmatic and is focused on short and medium term goals that accelerate adoption, as well as the long-term objectives in enterprise grid computing.

Grid computing is typified by a focus on sharing and managing pools of network-distributed resources to deliver applications and services. Grid computing environments may also be typified by -

- The use of network distributed, shareable pools of discrete resources to achieve greater strategic agility, architectural flexibility, performance, scaling, resilience and utilization
- A focus on managing services, rather than on managing individual, discrete components particularly as enterprise grids turn networks into arbitrarily rich and complex fabrics of resources
- Flexibility or mutability, as service components may be regularly composed, re-purposed or re-provisioned in response to changing goals, regulations, and business objectives or simply in response to a service's needs
- Application or service architectures that are disaggregated or distributed in nature and which leverage the properties of the fabric of resources. Examples include traditional multi-tiered applications such as ERP or CRM, Service Oriented Architectures (SOAs) or decomposable compute intensive workloads
- The consolidation of computing components into [typically] a smaller number of larger resource pools that promote easier provisioning, increased service availability, greater resource utilization and simplified management
- The standardization of components and their interfaces, configurations, processes and applications which all serve to promote highly automated and resilient architectures that can efficiently respond to changing business and service requirements

Enterprise Grid Computing is specifically the use of grid computing within the context of a business or enterprise, rather than perhaps for academic or research purposes. While there may be some overlap of requirements between enterprise grid architectures and other variants, it is clear that there are unique requirements and challenges associated with the adoption of grid architectures within an enterprise, especially in the operational sense.

Enterprise Grid Architectures are typically managed by a single enterprise, entity or business. A single organization is responsible for creating and managing a shareable networked pool of resources, composing higher-order components and services from individual resources, and delivering services that not only are capable of meeting a set of defined goals and requirements but also help drive value for the business. Resources can take the form of compute, network, storage and even service capabilities. The resources and services may or may not all be owned by a business. Subject to policy and business objectives, an organization may choose to leverage resources and services from another entity such as a service provider or an outsourcing or managed services firm. What defines the boundaries of the enterprise grid is its sphere of management responsibility and control. An enterprise grid may be confined to a single data center or it may extend across several. There are typically no geographic limitations to the size and scope of an enterprise grid architecture. The resources and services managed within enterprise grid architectures are however typically under the management responsibility and control of a single organization regardless of their actual physical location.

The EGA and the scope of each of the EGA working groups is described in greater detail in the publications: “Accelerating the Adoption of Grid Solutions in the Enterprise” and “Enterprise Grid Alliance Reference Model v1.0”, available on the EGA web site – http://www.gridalliance.org.

1.2 The Goals Of The Grid Security Working Group

The goal of the EGA’s Grid Security Working Group (“EGA-GSWG”) is to identify the unique security threats, issues and requirements associated with enterprise grid architectures and computing and to describe how these requirements can be satisfied (where possible) using existing and new technologies, processes and recommended practices. Like a traditional data center where compute systems and storage are connected to the network and administered by humans, managing information security in an enterprise grid environment is a risk management exercise where the benefits should outweigh the risks. The EGA-GSWG formally looks at the risks and information security of an enterprise grid so users can derive value while limiting their risk exposure.

The initial focus of all of the EGA’s working groups is on *commercial, enterprise applications* within a single data center. Commercial, enterprise applications are the lifeblood of most organizations. They are involved with the delivery of content and services to customers, partners, employees and shareholders as well as organizing and managing supply chain and business operations. Often, these services tend to be multi-tiered however this architectural artifact is not a requirement. Commercial, enterprise applications also may have both batch and interactive components, may be geographically distributed, may be based on commercial or open source software packages, and are often highly customized for a given organization’s needs. Examples of such applications include but are not limited to CRM, ERP, BI, etc. Such applications may or may not be enterprise grid enabled, by default. Applications that are not enterprise grid enabled may still be able to participate in an enterprise grid architecture through the use of a connector, proxy, or other mechanism.

The EGA expects to extend the scope of its working groups into multi-data center models as well as *technical enterprise applications* at a future date. Technical enterprise applications tend to be more compute intensive and less interactive,

The scope of the EGA-GSWG group covers the unique security issues in an enterprise grid environment where components are centrally managed and may be shared or rapidly repurposed. The goal of this working group is not to reinvent or relive discussion related to (non-grid or traditional) enterprise security controls and best practices. By identifying the *unique* enterprise grid security issues, organizations will be better armed with information in order to make appropriate risk management decisions as they adopt enterprise grid computing within their environments. Further, vendors can leverage this information to enhance their products and technologies to not only make them more competitive but also more readily able to support their customers' security needs.

Version 1 of this document focuses on enterprise grid security requirements. Later versions of this document will address how these requirements can be satisfied using existing technology. Furthermore, as this and other EGA working groups extend their scope beyond the single data center use case, security topics such as federation, cross-organizational trust models, and cooperative management and monitoring techniques will necessarily be addressed.

2 Enterprise Security Foundation

Given that enterprise grid environments are an evolutionary adaptation of enterprise computing, it must be recognized that the security policies, requirements and regulations that impact enterprise computing will certainly apply to these new environments. Similarly, since enterprise grid computing represents a logical step beyond enterprise computing, one can see how many of the requirements, architectural patterns and recommended practices will still apply to enterprise grid environments.

Enterprise security topics are generally well covered by academia, industry organizations, consortia, and standards bodies as well as by government entities. Those groups have typically focused on security controls related to areas such as:

- Platform and application security minimization and hardening
- Platform and application authentication, access control and auditing
- Network security configuration, filtering, monitoring and encryption

Much of the work in these areas is widely published and freely available on the Internet at the web sites of vendors, consortia such as the Center for Internet Security, as well as at government web sites such as those sponsored by NIST and the NSA. There exists a wide array of information covering everything from theory and general practices to specific product configuration and operational recommendations. Given that these topics are not new and are generally well understood and covered, this working group has chosen to leverage the work done by these organizations as a foundation on which enterprise grid specific security recommendations can be developed.

3 EGA Reference Model Security

The EGA reference model defines an enterprise grid as follows:

> An ***enterprise grid*** is a collection of interconnected (networked) ***grid components*** under the control of a ***grid management entity***.

Enterprise grids are also typically differentiated from more traditional data centers by management practices and technology which:

> Enable service or application-centric management rather than component-centric management
>
> Enable pooling and sharing of networked resources

The following sections build on the EGA reference model by providing more details on the various security specific aspects of enterprise grid architectures. For more on the EGA reference model, see "Enterprise Grid Alliance Reference Model v1.0."

3.1 Grid Component Security

A ***grid component*** is defined as a super class of object from which all of the components that are managed within an enterprise grid are descended. This includes everything from servers, network components, and disk arrays to applications and services such as databases, an ERP service, an online bookstore, etc. The nature of grid components is that they can generally be combined in a variety of ways to form more sophisticated elements (that are themselves grid components).

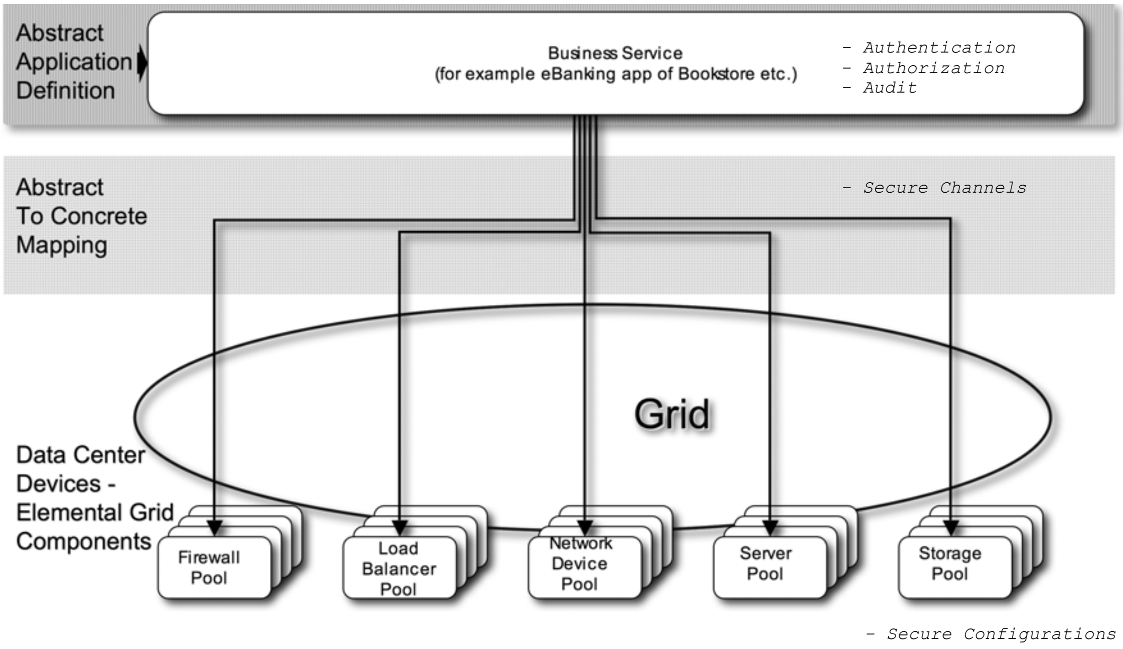

Figure i - Abstract To Concrete Component Mapping 1

Figure i illustrates a simple example of a high level mapping between a service and its underlying physical and logical grid components. It should be understood that each physical and logical grid component has its own security properties and controls (i.e. configurations). This is, in fact, no different than how individual products and components are deployed today. Each must be secured or be combined with other components to achieve a desired level of security and assurance. Similarly, aggregates and combinations of grid components may also have unique security requirements or properties that are in addition to the logical union of the properties of the elements themselves. Again, this is no different from how these components would be deployed in traditional data centers today. For example, the notion of an authorized transaction for an e-Banking application is preserved through the mapping to the various physical and logical components whether it is deployed in an enterprise grid environment or not. Enterprise grid computing does not negate the need for traditional security principles and controls in the areas of identification, authentication, authorization, confidentiality, integrity, availability, non-repudiation and audit – just to name a few. If they were needed in a traditional enterprise architecture, they will likely be needed in an enterprise grid environment.

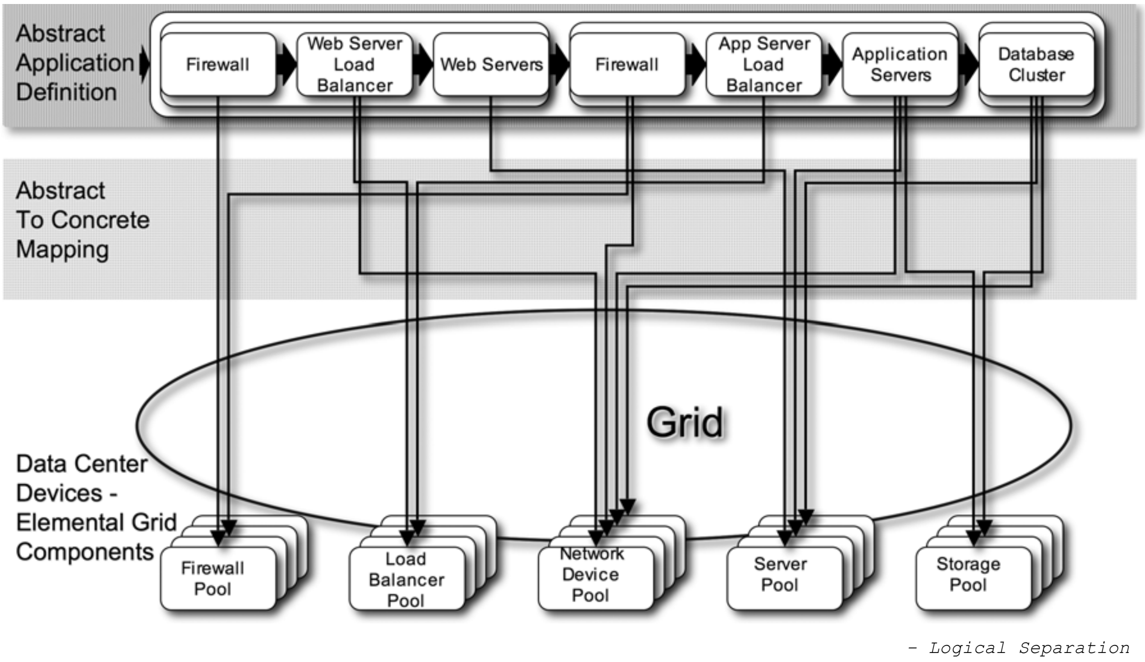

Figure ii - Abstract To Concrete Component Mapping 2

Figure ii illustrates an example that highlights how discrete components that may have once been physically separated can now be mapped to a shared pool of physical grid components. In this case, there are security issues that, while not new, are more prominent in a grid environment. When moving from physically silo-ed environments to ones where components are pooled and shared by several, often unrelated applications and services, the pooled resources need to provide for logical separation and security between the different applications. For example, a storage resource may contain sensitive information that should only be accessed by and through a single application, even though the actual physical storage pool is also used by other applications.

This sort of logical separation is not entirely unique to enterprise grid environments but it is worth pointing out. Similarly, network traffic needs to be segregated as appropriate between different applications and services even if they are running on shared resources. For instance, network traffic for an HR department should only be seen and received by the HR applications even though other applications and user communities may share the same physical system, network or storage resource. In fact, depending on the requirements defined for a service and level of trust and assurance required, physical or electrical separation may still be required for certain components. The enterprise grid architecture must be flexible enough to still provide for dedicated resources through its policy definition and enforcement mechanisms.

Note that this is not simply a physical resource issue. The security concerns heightened by resource pooling apply equally to logical resources such as shared services (naming, directory, time, logging, and other services) as well as controls such as software-based clustering mechanisms, packet filters, intrusion detection systems, etc. For example, rather than pooling all of your directories into a shared directory server pool, it may be useful or required that some of the directory trees or entries be isolated from the rest.

This example also highlights the need for a flexible, extensible security policy definition and enforcement framework that facilitates the types of compartmentalization, least privilege and defense in depth noted above. This too is not specific to enterprise grid architectures although the centralized management framework provided by the grid can help simplify the deployment and enforcement of centralized policy.

Grid components, from composible services down to each of the logical and physical grid components have security properties or attributes associated with them. These security attributes may be internal attributes of the grid component (e.g. file permissions in a file system), or they may be attributes explicitly associated with a managed grid component within the Grid Management Entity (see next section). The latter type of attribute is more typical of a grid environment versus a more traditional data center environment.

In addition, components may define specific dependencies. Such dependencies can be placed on attributes specific to the component or on external elements (services, service attributes, etc.) These dependencies can be used to help enforce security policy and to ensure that exposures are minimized. For example, in Figure ii above, it is likely that the service represented by the abstract application definition would declare a dependency such that a web server is not started until the resource hosting the web server has been adequately secured (and validated). Similarly, another dependency could insist that a firewall be provisioned and readied before the web server could come online in order to ensure that the network security policy is properly enforced.

This concept of enterprise grid wide dependencies and constraints would allow entire services or business functions to be securely provisioned, configured and enabled. Every step of the way, the security attributes, dependencies and constraints would be enforced to help minimize risk and limit exposure. This capability is not without risk itself however. Poorly configured attributes or misaligned dependencies could open holes in the environment or prevent services from properly being enabled and used. It is critical therefore that sanity and validation checks be implemented to help detect such cases where possible. Where automation is not possible, additional guidance must be provided in the areas of awareness, training and process.

3.1.1 Grid Component Life Cycle

The EGA reference model defines the following life cycle states of a grid component:

- Provisioning
- Ongoing Management
- Decommissioning and Re-purposing

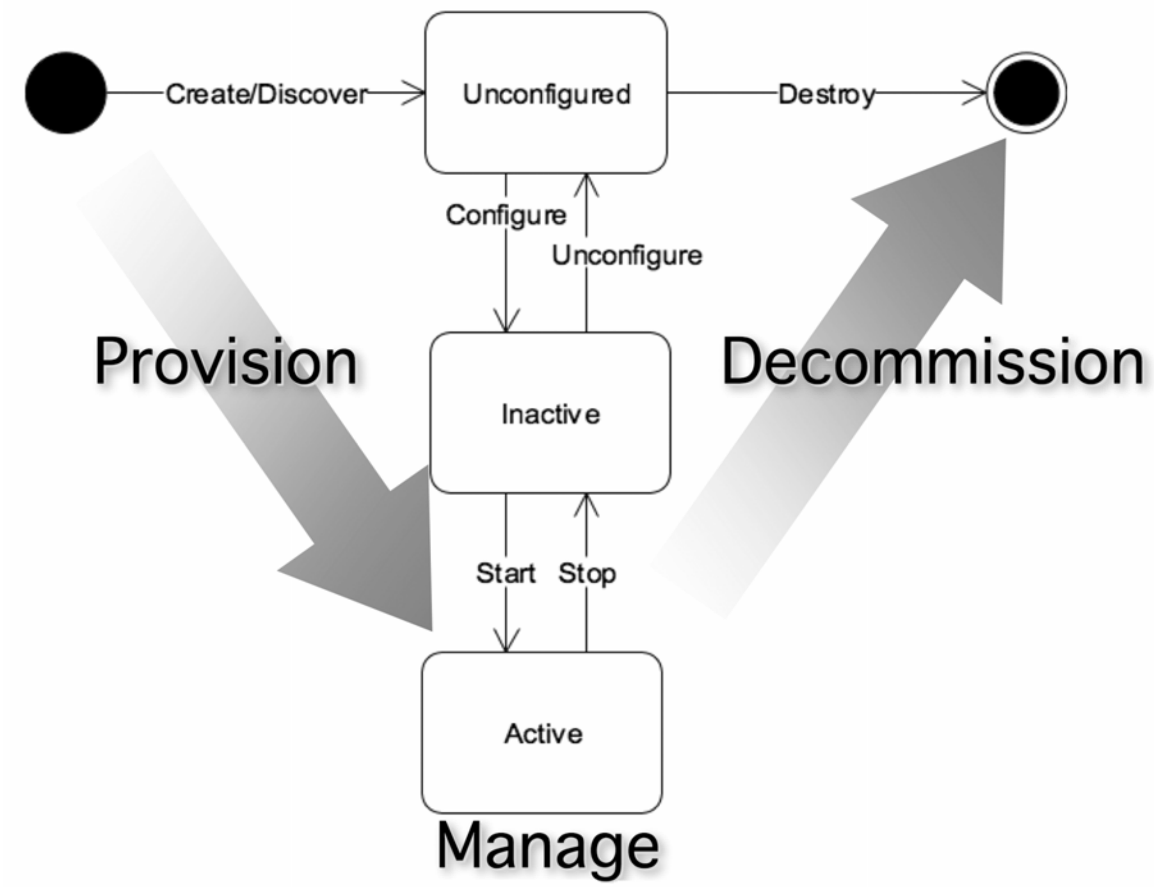

Figure iii – The General Grid Component Life Cycle

Figure iii illustrates these states with the transitions between them. Each state has security attributes and properties that must be considered as described in the following sections.

3.1.1.1 Provisioning

Provisioning a grid component involves adding or creating the component, configuring it, and starting it up to put it in an active state. Among the security issues that must be addressed are:

- Who (user or application/service) can provision (create/discover) a given grid component?
- Who attempted to provision a given grid component?
- Who did provision a given grid component?
- When was the grid component provisioned (for accounting purposes)?
- What is the grid component's provisioning history (for audit records)?
- What was the state of the grid component before provisioning (fresh installation, repurpose, unknown, etc.)?
- Was the grid component verified after provisioning to ensure that it had entered the expected state (configuration and runtime)?
- Is the software image to be provisioned trustworthy? Is it coming from a trusted source? Was the image's integrity verified? Does it contain any malicious code?
- Have all required dependencies been satisfied prior to provisioning?

- Are there any constraints that would preclude using a grid component for a given purpose (placed in a non-secured location, lacks sufficient resources or security capabilities, is currently a part of an investigation, etc.)?
- Has the security posture and integrity of the provisioned grid component been validated prior to it being enabled for use?
- Provisioning priorities (so a high-priority requester can preempt a low-priority requester)
- If the grid component was previously used, has the grid component been scrubbed so that no sensitive data remains for the next user, application or service to see? This may include Flash PROM or BIOS settings, operating systems, applications, user configuration and data objects, and any other information that is associated with the original purpose of the grid component prior to decommissioning and (re)provisioning.

3.1.1.2 Ongoing management

The ongoing management of a grid component involves any management related activities while the component is in an active state. When a component is not provisioned (and therefore not active), no management functions except provisioning can be performed on the component. Among the security issues that must be addressed are:

- Who has the authority to create, modify or remove administrative roles?
- From where can administrators perform grid management functions? Are there restrictions as to what actions can be performed based on where an administrator connectors or what how the administrator authenticates to the enterprise grid?
- Who has the authority to create, modify or remove grid components?
- What administrative roles have been created, modified or removed? When? By whom?
- Who (user or application/service) can manage the grid components and their security attributes?
- Who can define relationships between grid components?
- Who can define dependencies between or constraints on grid components?
- What security attributes can be managed for a grid component?
- What security attributes, relationships, dependencies and constraints were changed? When? By whom?
- How can security attributes and policies be securely distributed or updated to grid components in a heterogeneous environment? Also, how are security policies consistently interpreted by grid components in a heterogeneous environment?
- Is there a common taxonomy for describing security attributes across a range of products? Are there common attributes that can be defined on a per product type basis (e.g., web server, directory server, etc.)? How are product specific attributes added and managed within the context of a security policy?
- Can the grid management entity use both common attributes and vendor/product specific attributes when defining requirements or making decisions?
- How is the security configuration of each grid component validated once it is deployed? Is there a standard form that these validation checks should use? How will validation checks across a wide range of products be evaluated for compliance both individually and as part of a composed resource?
- What are the repercussions for failure? Is there an automated or manual process for security failure detection and recovery? If automation is possible, to what level of specificity should this be declared? How will administrators be notified of a security failure (and any subsequent recovery actions)?
- Grid components should be changed only through the grid management entity. How are external changes detected and corrected?

- How are changes to the grid management framework itself prevented and detected? What should happen if the grid framework detects a security breach or violation of systemic integrity?
- What capabilities should exist for allowing the grid management framework to normalize, correlate and report on security events happening throughout the enterprise grid architecture?
- Can the grid management framework itself be optionally segmented into multiple tiers or compartments in response to an organization's risk management decisions?
- Who has the responsibility for reviewing and responding to security events?
- How will users authenticate to grid components?
- What mechanisms will users utilize to access grid components?
- What type of network communications will be allowed to enter or leave grid components?
- Who or what decides the type of network communications that will be allowed to traverse the grid?
- How will security breaches to grid components be contained?

3.1.1.3 Decommissioning and Re-purposing

Decommissioning a grid component may be done for a variety of reasons. The grid component may be fully decommissioned in order to retire a service and/or remove an obsolete grid component. Decommissioning may also be initiated in order to re-purpose a grid component so that it can be provisioned yet again. Among the security issues that must be addressed are:

- Who (user or application/service) can decommission/re-purpose (create/discover) the security attributes of any given grid component?
- Who attempted to decommission/re-purpose a resource?
- What resource was decommissioned/re-purposed? When? By whom?
- When was the resource decommissioned/re-purposed (for accounting purposes)?
- What is the resource's provisioning/re-purpose/decommission history (for audit records)?
- Has any required state been captured prior to the resource being decommissioned/re-purposed (e.g., audit and system logs, cryptographic key material, forensic data, etc?)
- Under what conditions can a resource be decommissioned/re-purposed? Are there global constraints or constraints specific to locations, services, users, etc.?
- Have all required dependencies been satisfied prior to decommissioning/re-purposing?
- Has the resource been sanitized prior to decommissioning/re-purposing? Has the desired level of assurance been satisfied with respect to the sanitization procedures used? The grid component must be scrubbed so that no sensitive data remains for the next user, application or service to see. This may include Flash PROM or BIOS settings, operating systems, applications, user configuration and data objects, and any other information that is associated with the original purpose of the grid component prior to decommissioning. The level of scrubbing may depend on whether the grid component is being re-purposed or being removed entirely from the grid.

3.2 Grid Management Entity Security

The EGA reference model defines the ***Grid Management Entity (GME)*** as the ***logical*** entity that manages:

- The grid components

- The relationships between grid components
- The entire lifecycle of the grid components (provisioning through decommissioning)

The GME may be realized as any combination of people, process, and technology. While the GME is logically distinct from the managed grid components, the realization of GME functionality may not be as clearly separate from the realization of the grid components themselves. Figure iv shows the GME as a logically separate entity from the grid components being managed, but in actuality a given grid component could include management (GME) functionality. The GME is therefore essentially hierarchical and distributed in nature, and may be decomposed in the same way as applications and services, i.e. into grid components. Some degree of caution should be exercised however if a grid component itself performs GME functionality as you will lack compartmentalization of functionality and the security failure of the component could have repercussions to the validity of its management functions as well as to the entire GME (if the security failure could result in unauthorized access to other GME functions, for example).

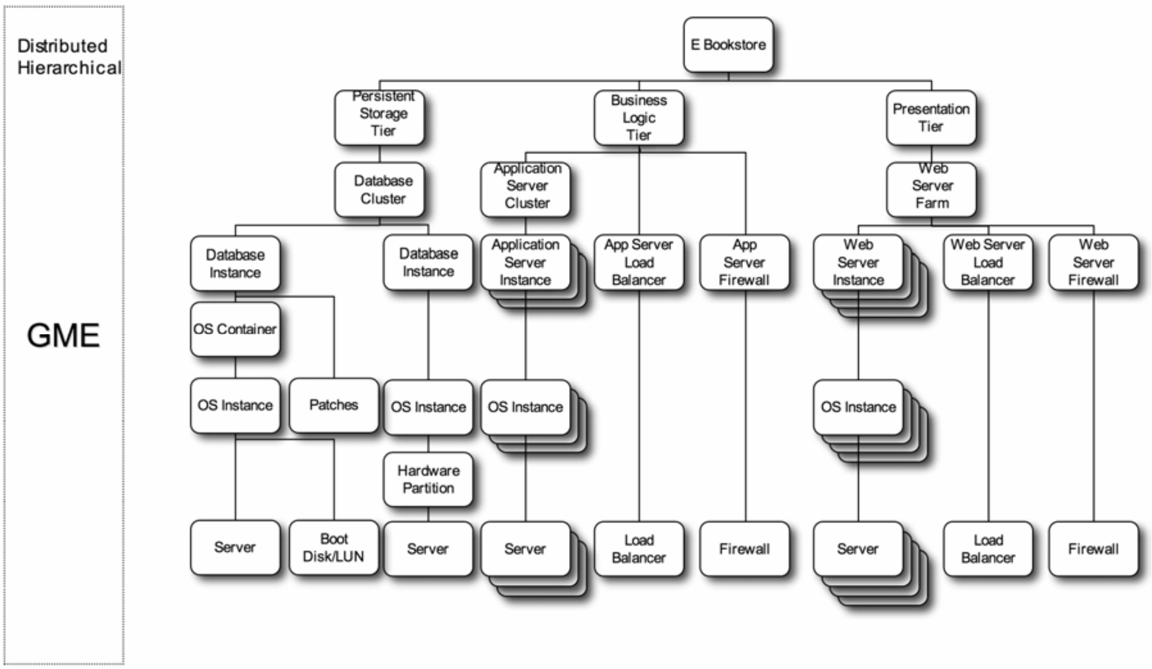

Figure iv – The GME As Logically Separate From The Grid Components

One of the key areas of management for the GME is in support of the definition, enforcement, and validation of the enterprise grid's security policy. In addition to managing the grid components, the relationships between components, and their life cycles, the GME manages the following security functions:

- Management of all user identities, administrative roles
- Management of all core enterprise grid wide shared services
- Authentication of identities (users, applications and services)
- Authorization of actions taken by both authenticated and unauthenticated principals
- Restricting access to the GME and all other enterprise grid components
- Capture, storage, analysis and reporting of all security and audit-related events
- Management (creation, installation, rotation, validation, storage, destruction, etc.) of all

cryptographic keys, shared secrets, etc. used by elements in the enterprise grid architecture

- Enforcement of secure communication across the grid environment (including administrative access – local or remote)
- Enforce secure isolation of shared grid components as well as services constructed from physical grid components
- Ensure local and remote management and troubleshooting operations of the entire grid architecture are secured in accordance with an organization's security policy.
- Validation of individual or groups of grid components to determine if they are in their expected security states with respect to object integrity, attributes, dependencies, constraints, etc.

While enterprise grid specific requirements associated with individual components have been discussed, it should be noted that the majority of those requirements center on the Grid Management Entity. This is in part a result of the role the GME plays with respect to policy definition, enforcement and validation. Grid resources (or simply pools of networked resources) alone are not unique to a grid environment. What is unique is the way in which they are aggregated and managed. By introducing the GME with the ability to provision, manage and decommission pools of grid resources, we get to the heart of the unique threats and security requirements in a grid environment. By understanding and addressing these issues, one can take advantage of the benefits of (logical) centralization of security management through the GME.

4 Grid Security Use Cases

The following use cases build on those developed by the EGA Reference Model Working Group by providing a greater focus on the security risks, issues and actions expected in real world situations.

The following cases are intended to highlight the specific security risks associated with enterprise grid computing environments for which requirements or guidance should be elaborated. These use cases assume that the underlying compute, network and storage infrastructure has been adequately secured for the environment into which the grid will be deployed. Recommendations to secure these infrastructure elements are widely available through vendor, industry and government sources.

4.1 Generic Use Case – Provision A Grid Component

The first phase any grid component will enter is provisioning. The provisioning process is further broken down into three phases, each of which could be considered a sub-use case:

4.1.1 Add/Create Grid Component

This phase captures the creation of a new grid component. This activity can include the following activities:

- A grid component joins the grid, is discovered, or is manually added so that the GME is able to identify and manage it. The grid component must be a legitimate component and not a rogue component masquerading as something legitimate. Likewise, the potential grid component must be able to verify that it is communicating with a legitimate GME.

  This joining or discovery process must therefore include a form of mutual authentication where a trust relationship is established between GME and grid component. The requirement of mutual authentication should be configurable to accommodate those environments that have mitigating controls in place to prevent grid component masquerading.
- A new instance of a grid component is automatically (through the use of dependency relationships) or manually created from an existing logically manageable grid component. The existing component must previously have been authenticated or otherwise trusted by the GME. The new instance must also be authenticated, as it is a grid component in its own right.

Whether the grid component is discovered or newly created, this sub-use case also results in the GME tracking the grid component, its security policy and properties as well as its capabilities. In cases where its security properties, dependencies or constraints may not be known and cannot be discovered, the GME will prevent the component from being actively used until its security requirements and constraints have been defined.

4.1.2 Configure Grid Component

This phase captures the act of configuring a given component so that it may be activated. Attributes and other elements used to configure a grid component may be manually assigned or they may be discovered through the use of interfaces provided by the GME. If a given security configuration element is not applicable for a component or the component can not implement the dictated security policy, then the GME will not permit the component to be actively used. Administrative intervention will be required to either permit the use of the component or to adjust its security policy or configuration. Configuration activities may include:

- Defining or discovering the baseline security requirements for the component
- Defining or discovering security dependencies and constraints
- Applying access control policy for users and administrators of the component
- Defining access paths for system, network and storage communication
- Binding security parameters to the security parameters of other grid components
- Validating that the security policy for the component has been correctly applied and all required dependencies are satisfied
- Specifying security settings that define the SLOs. This may include security "hardening" or "lockdown" requirements (e.g. ports/protocols available in a web server service, available o/s services, etc.)

4.1.3 Start Grid Component

This phase describes the act of enabling the component and placing it into the enterprise grid in an active state so that it can be used. This can only be performed by a properly authenticated and authorized user or service working through the GME. Further, this operation can only be applied to a grid component that has been provisioned and whose security policy has been validated (per the configuration step above). Similarly, before a grid component can be enabled and placed into use, all of its required dependencies and constraints must be satisfied.

Note that users, administrators and services may not be permitted to directly start or stop grid components. Depending on an organization's policy, the GME may be used as an architectural hypervisor that must be accessed to perform those functions. In this case, an administrator would interface with the GME to perform various component operations. The GME would then complete the actual component operations on behalf of the administrator (assuming the administrator had proper authorization to perform the actions). This model has the benefit of never allowing users or services to have direct management access to other grid components. Further, the use of the GME in this capacity will ensure that all management operations are validated and audited, providing a higher degree of operational assurance. Lastly, this model has the added benefit of permitting the GME to automatically restart failed components (subject to policy).

4.2 Generic Use Case – Monitor and Manage A Grid Component

The monitor and manage use case involves the ongoing security management of each grid component. This is predicated on a successful provisioning of the grid component where a trust relationship is established between the GME, the grid component, and its constituent parts. With this trust relationship, the GME monitors and therefore has authorized access to information regarding the grid component. This information may be sensitive and is most likely not public information or even information that can be shared with other grid components (without a need to know). This use case also includes management commands from the GME to the grid component. These commands must be secured so that the grid component trusts their origin. Security related monitoring and management activities might include:

- Automatic discovery and manual definition of attributes that can be monitored and managed from the GME.
- Monitoring security related metrics and events (e.g. grid component utilization and availability, user/service logon and logoff attempts to an application, service or GME running on the grid, etc.). Intrusion detection and prevention functionality falls into this category.
- Logging, storage, analysis and correlation of all grid component audit and security events. The GME may also be required to measure and record usage for the purposes of chargeback, compliance, audit requirements, etc. The audit logs must be protected from unauthorized disclosure, modification and destruction.
- Reporting security patch status and applying security patches (through the GME)
- Identification of components installed into, modified or removed from the enterprise grid without authorization or in violation of grid policy
- Any other system or identity management activity performed by the GME

4.3 Generic Use Case – Decommission A Grid Component

The decommissioning process may be broken down into three phases, each of which could be considered a sub-use case:

4.3.1 Stop Grid Component

This action is performed by the GME on behalf of an authorized user or service. This action will not only disable the designated grid component (and its constituent parts) but may also disable any components that rely on this component if appropriate. Disabling a grid component will take it out of service. Existing users or transactions operations on that component will be transferred if possible to other resources (depending on site policy). Once the component has been quiesced, it will be stopped. Alternatively, the GME should also support the notion of immediate stop whereby a grid component is halted with no attempt to save existing service state, user data or transactions. After the grid component is stopped, it is still in a state where it can be managed by the GME.

4.3.2 Unconfigure Grid Component

Before a grid component can be fully decommissioned, it must first be unconfigured. The activities associated with this step include the saving of any critical state and the sanitization of the component. Critical state such as security or transaction logs, cryptographic key material and perhaps event key component configuration and data files may need to be preserved for future use or to support audit or forensic investigations. Consequently, before a component is unconfigured, its critical state must first be captured.

Once complete, the component can then be sanitized. From a security perspective, the grid component must have all sensitive information scrubbed, and related security policies and properties removed from the GME. Also any security policies and properties that protect or lock it from future usage must also be reset.

Once a grid component is unconfigured, it may be removed from the grid (as described next) or it may be repurposed and reprovisioned to serve in another capacity. For the latter case, the degree to which the grid component is unconfigured may depend on how it will be repurposed. For example, a compute component running a web server may simply be reprovisioned to run as a web server for different application or service. Or, the same compute component may be reprovisioned as a database server for a different application over a different network for different users, and therefore require greater levels of sanitization.

4.3.3 Remove Grid Component

Once the grid component has been sanitized, it should be removed from the grid to complete the decommissioning process. This will prevent the component from being repurposed for use by another grid component. When a grid component is removed, it is completely separated from the grid. It is not permitted to participate in any enterprise grid activities. In order to rejoin the grid, it must be added once again and go through the normal grid component creation/addition process. Note that when a grid component is removed, it is no longer being controlled or monitored by the GME. As a result, it may be used by non-grid environment applications. This must be taken into account if that component is ever added back into the grid.

4.4 Operational Use Case

This use case focuses on ongoing security related events that are orthogonal to the three grid component life cycle use cases described above. This use case may be broken down into the following sub-use cases:

4.4.1 User/Service Security Events

This phase includes basic user and service security actions including:

- User/Service Grid Logon - A user or service attempts to authenticate to the GME (successfully or not).
- User/Service Grid Logoff - An authenticated user or service attempts to logoff from the Grid.
- User/Service Role Assumption – An authenticated user or service attempts to assume another identity or role and must therefore re-authenticate to the GME in order to obtain this additional level of privilege. The method of authentication may be required to differ from that originally used by the user or service to initially authenticate to the GME.
- User/Service Logon Status – An authenticated user, role or service (with the proper authority) attempts to determine if a specified user or service has successfully authenticated to the GME.

4.4.2 Workload Security Events

This sub-use case takes a workload or “unit of work” perspective and assumes that grid components are not dedicated to and owned solely by a single application or service. This phase includes workload security actions including:

- Workload Submission - An authenticated user or service attempts to submit a workload for execution. This action should address whether the user may submit a workload and for what workload classes (if defined). This action may further restrict where a given workload may run, whether it can be suspended and resumed, whether it can be moved, when it must be started or stopped and for how long it is permitted to run, as well as other relevant attributes.

- Workload Termination - An authenticated user or service attempts to terminate an existing workload. This action should address whether the user may terminate a workload or class of workloads. It may also introduce restrictions as to if or when certain types of workloads can be terminated (in order to ensure consistency and integrity).
- Workload Monitoring - An authenticated user or service attempts to monitor the progress/status of an existing workload. This action should address whether the user or service has the necessary authority to monitor an existing workload. It may also introduce restrictions as what information may be queried based on the credentials of the requestor.
- Workload Scheduling - An authenticated user or service attempts to schedule a workload or set of workloads for execution. Similar to the Workload Submission action, this case determines what workloads may run, where they may run, and in what order. This action may be responsible for workload suspension and resumption based on scheduling priorities as well as resource availability. Further, this action may explicitly declare which workloads (or workload classes) may not run concurrently (on the same resource or perhaps across the entire Grid). This may be necessary in order to prevent operational conflicts or promote user, service or “customer” isolation.
- Workload Batch Processing – Responsibility of the GME. This action is responsible for controlling workload behavior in the Grid. In particular, this action is responsible for the transfer of a workload to a resource, the collection of results (if any) from a resource as well as all workload control and monitoring functions such as start, stop, suspend, resume, query state, etc.

5 Grid Threats and Risks

Enterprise grid computing offers a new way to visualize, aggregate, virtualize, provision and manage enterprise IT environments. This new model empowers organizations to standardize, automate and optimize their IT assets in ways not ordinarily achieved by the design of today's data centers and corporate networks. These benefits must be tempered with the risks of deploying such architectures. To manage these new risks, we must first develop a better understanding of the threats that are specific to enterprise grid computing architectures.

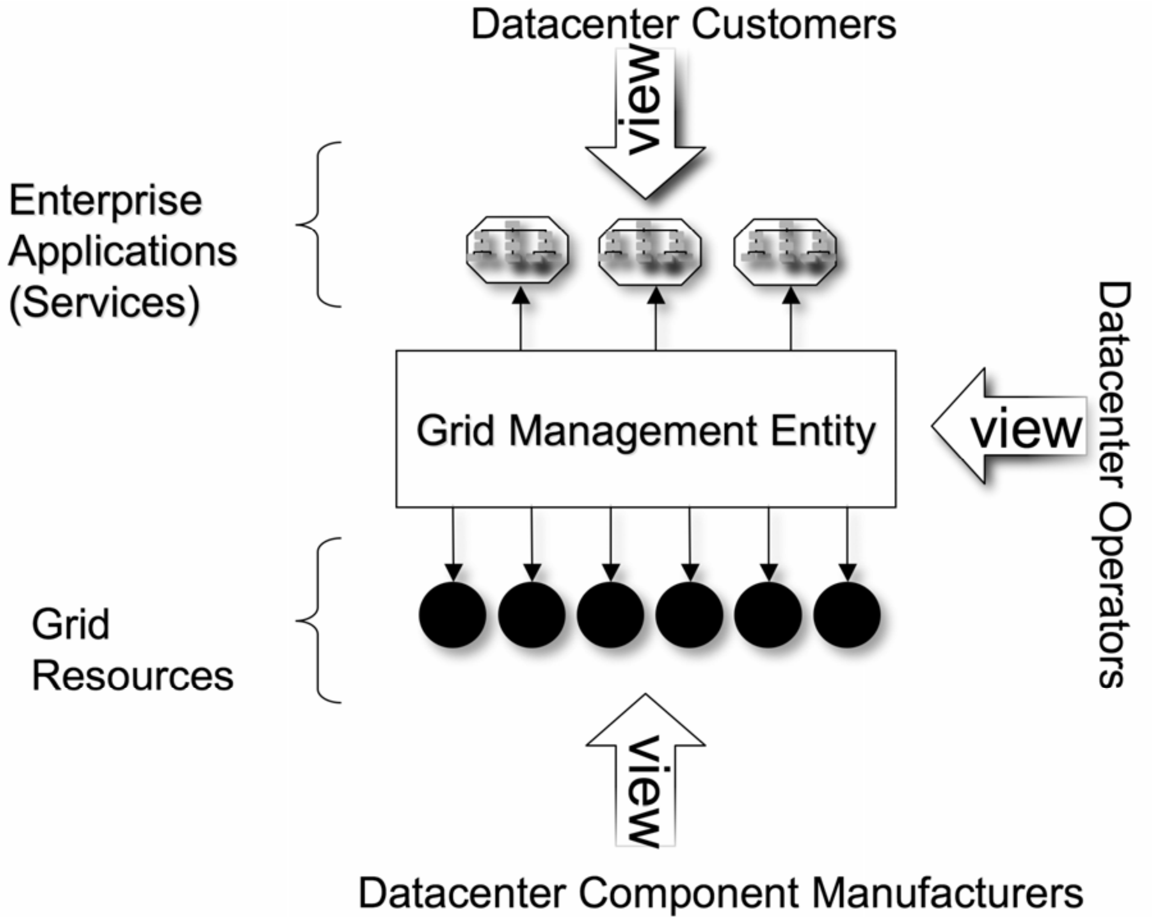

Figure v – Basic View Of A Data Center

The novel security concerns raised by enterprise grid computing arise from how the GME communicates with individual and groups of grid resources to make them work together as one, as illustrated in Figure v. The GME is a powerful entity in that it knows all of the grid resources and can tell each of them what to do and when to do it. It also ensures fairness—when there are multiple workloads presented to the grid, it is the GME that makes sure everyone's workload gets its fair chance to run on the grid. Further, it is also the GME that enforces, audits and validates compliance of grid components from a security policy standpoint.

An enterprise grid is also characterized by the use of flexible, shared pools of grid resources that may be regularly repurposed and reprovisioned. The enterprise grid component life cycle must therefore also be considered when looking at security threats and risks.

The following categories of threats and risks are based on the unique characteristics of an enterprise grid. This list is not intended to be all-inclusive. It is provided as a representative sample of the types of risks and threats inherent in this kind of architecture.

- Access control attacks. This describes risks associated with unauthorized entities and authorized entities bypassing or defeating access control policy.
    - Unauthorized grid component or user joins the grid
    - Unauthorized user or service submits/attempts to submit, terminates, control or monitor grid applications and services
    - Authorized user or service bypasses or attempts to bypass access controls for applications and services running on the grid
    - A user, application or service running on the grid attempts to exceed its permitted privileges or resources
- Defeating grid auditing and accounting systems. This describes any threats to the integrity of auditing and accounting systems unique to an enterprise grid environment. This includes false event injection, overflow, event modification, and a variety of other common attacks against auditing systems.
- Denial of Service (DoS). This describes any sort of attack on service or resource availability. While an enterprise grid generally offers better availability compared to a non-grid environment, the following DoS threats must be considered as part of a risk assessment.
    - DoS attack against the GME
    - DoS against the grid component join protocol to prevent new authorized grid component or users from successfully joining the grid
    - Authorized grid component or user is "forced" to leave the grid
    - User or service attempts to flood the grid with workloads such that compute, network and/or storage components become exhausted, or the latency to access those resources significantly impacts other grid users
    - Defeating or modifying scheduling messages from the GME to grid components to unfairly prioritize one application/service over another
    - Other DoS attacks affecting the entire grid and grid-wide QoS
- Malicious code, malware. This describes any code that attempts to gain unauthorized access to the grid environment, elevate its privileges, hide its existence, disguise itself as a valid component, propagate itself, etc. in clear violation of the enterprise grid architecture's security policy.
- Object reuse. This describes how sensitive data may become available to an unauthorized user. In the enterprise grid context, this is a risk if a grid component is not properly decommissioned (and sanitized).
- Masquerading Attacks, This describes as class of attacks whereby a valid grid component can be fooled into communication or working with another entity masquerading as valid grid component. Such a failure could permit the disclosure or modification of information, the execution of unauthorized transactions, etc. The impact of such a breach will depend on the trust relationship between the target and attacking components and the security policies currently in place.
- Sniffers. Watching packets as they travel over the network is referred to as sniffing or snooping. An enterprise grid potentially introduces additional network traffic between applications/services, the GME and grid components that should be protected. Failure to address this threat may result in other types of attacks including data manipulation and replay attacks.

In addition to the threats and risks listed above, there are other general categories that are not unique to an enterprise grid environment but warrant some mention here.

- Physical security is an important component of an information systems overall security posture. An enterprise grid, like any information system, needs to protect against physical threats from humans (either malicious or accidental) as well as man-made and natural disasters. Having one enterprise grid system instead of multiple silo-ed systems may lead to some efficiency in addressing physical security threats.
- Social engineering describes how an unauthorized user cons an authorized user into providing information needed to access a system. An enterprise grid does not necessarily introduce new forms of social engineering threats, and may even reduce these threats if an enterprise grid is rolled out with a strong set of security controls (e.g. a clear admin model, processes, and policy).
- Regulatory Compliance ensures that the functions being performed by the grid and its users are not violating any government or industry laws. The security requirements that are within the regulations should be incorporated to the grid policies, procedures and processes.

As the GME evolves in any given environment from a combination of people, process, and technology to something that is more automated, many of these risks become clearer. Although centralization (even if it's just logical) of enterprise grid management in the GME may make an attractive target for malicious users, more resources can be brought to bear to secure the grid in a focused manner through the GME. The more the GME is automated, the easier it is to address many of these risks.

All the risks listed above should be included as part of an overall risk assessment. A business decision can then be made to mitigate these risks, transfer them, or to accept them. The next section delves further into the grid security requirements that can be used to mitigate these risks.

6 Grid Security Requirements

Security requirements for a given organization can and should factor in any unique aspects of the organization. These requirements should be based on an organization's security policy. The security policy should be derived from some sort of risk assessment exercise. For an enterprise grid environment, the threats listed above can form the basis for a risk assessment. An organization then needs to decide whether they want to mitigate a given risk, transfer the risk, or accept the risk based on a cost/benefit/impact analysis. The following security requirements are based on an approach to mitigate unique threats and risks in an enterprise grid environment.

6.1 Confidentiality, Integrity, Availability (CIA)

These three basic requirements of any information security system apply in an enterprise grid computing environment. For the most part, these requirements are likely met by each grid component's existing security functionality (and not necessarily unique to the grid environment). There are, however, aspects of these requirements that are unique to the grid environment (particularly with the presence of the GME).

- Communication must be secure between the GME and grid components, or between collections of grid components themselves. This includes providing confidentiality through something like channel encryption, as well as integrity checks to guard against tampering across the wire. This may also include satisfying a non-repudiation requirement where that is needed.
- Confidentiality of sensitive data must be preserved through the life cycle of grid components (from decommissioning to repurposing and reprovisioning)
- Images used to provision grid components and settings used during the configuration process (attributes, dependencies, constraints) must be validated for integrity. Changes to these settings must be controlled by the GME. Similarly, violations must be detected by the GME. Items whose integrity cannot be validated must be isolated from the enterprise grid to prevent their use and/or propagation.
- Grid components themselves must be validated for security and integrity in accordance with an organization's enterprise grid security policy. This includes individual components as well as composed collections of components.
- The integrity of information preserved from provisioned resources must also be validated. This may include log files, cryptographic key material, or other material that is collected from a resource prior to it being decommissioned.
- Availability is not only a requirement but also a design center and touted benefit for enterprise grids. Preserving availability of (and protecting against DoS attacks on) the enterprise grid, the GME, and grid components is therefore a requirement.

6.2 Identification

One basic building block of security systems is the ability to uniquely identify everything. In the enterprise grid, this applies to all grid components and user communities. In particular, grid components must preserve their unique identity through the life cycle of repeatedly being provisioned and decommissioned. Alternatively, if a new identity is created every time a grid component is provisioned or reprovisioned, past identities must be recorded for audit and forensics purposes (most likely by the GME). The GME must also be uniquely identified so that grid components and applications/services know who they are communicating with.

6.3 Authentication, Authorization, and Auditing (AAA)

The access management related requirements of authentication, authorization, and auditing also apply to the grid environment. Aspects of these requirements that are unique to the grid environment include:

- In order to ensure secure communication between the GME, grid components and application/services, each communicating entity must be able to authenticate to one another.
- Grid components may communicate with other grid components subject to a defined security policy. Grid components must be authorized to communicate with other grid components. Authorization can take either strict or loose forms depending on the organization and environment into which the enterprise grid is being deployed.
- The auditing capabilities of an enterprise grid environment must track and be able to resolve the dynamic binding of grid components and their potentially short life cycles. Audit data must still be meaningful even after audited grid components are reprovisioned or decommissioned.
- The GME must provide to an enterprise grid a set of functionality equivalent to an AAA server in a non-grid environment. This includes support for policy-based, extensible, and “strong” authentication mechanisms (e.g. SAML, X.509, etc.) and support for role based access control to grid resources.

6.4 Separation of Duties, Least Privilege

The two standards of access control policy, separation of duties and least privilege, also apply to an enterprise grid. Focusing on the unique aspects of an enterprise grid, these standards are applicable to the GME and the associated administrators. In some cases, it may make sense to define new administrator roles (e.g. “grid admin”) to support this separation of duties. Even with new roles, each and every role should be designed using the least privilege (or “need-to-know” and “need-to-have”) principle as part of security best practices.

6.5 Defense in Depth

Another general security principle is defense in depth. One example of this in a traditional networked system is configuring physical separation of network segments into demilitarized zones (DMZs). In an enterprise grid, traditional defense in depth measures such as DMZs should be preserved, even if they are accomplished using logical mechanisms within a pool of grid components. Additional defense in depth measures can be taken by looking at the directed acyclic graphs (DAGs) of the EGA reference model and utilizing security measures where possible to reinforce systemic security at every layer of the graph.

6.6 Fail Secure

Like any well designed information security system, each individual grid component, the GME, and the enterprise grid as a whole must be designed to fail securely. Since an enterprise grid does place a lot of emphasis on grid component life cycles, the status of desired state changes may resulting a success or failure. To fail securely means that the grid component or any other aspect of the enterprise grid are not placed in a vulnerable state that may be susceptible to any kind of security threat.

6.7 Grid Lifecycle Security Requirements

There are also security requirements that are unique to the enterprise grid environment that are related to the life cycle of grid components. Since the enterprise grid environment is one where grid components are frequently reused, it is important that there is “secure packaging”, “secure update”, “secure archival”, and “secure reuse.”

- “Secure packaging’ refers to the ability to logically package grid components so that they can be easily provisioned to and deprovisioned from resources in the grid. This allows each grid component to be logically isolated from other components. Further, this allows each package to be subject to change control, revision control and integrity management. Similarly, each package could be digitally signed and/or encrypted depending on site security policy and requirements.
- “Secure update” refers to the ability to safely update deployed grid components using a newer version of the object. This includes the ability to securely communicate with the components for the purpose of querying existing state, updating as well as check pointing changes so that the grid component can be rolled back to an earlier version should a problem be detected. This may be necessary in order to correct a known flaw or weakness, remediate a security vulnerability, or even incorporate a new feature or capability. Further, this capability may be used as part of a compliance effort to determine which grid components may be in need of an update. This process will rely on the “Secure packaging” concept in order to preserve proper change and revision control.
- “Secure archival” refers to the ability to extract from a provisioned resource information that may be needed at a later time. This includes packages such as those described above as well as audit logs, cryptographic key material, or other sensitive configuration and data that must be preserved and survive repurposing and potentially decommission of assets.
- “Secure reuse” refers to something as simple as scrubbing EEPROM, memory or disk space, or it could be something more complex. This requirement is not new to the grid. For example, operating systems do this all the time with memory and disk usage. For the grid environment, the same requirement applies to each and every grid component that is being pooled and reused. Grid components must be re-initialized (or wiped clean) before the next user has access to it. In some cases, part of grid component may need to meet this requirement if there is more than one user using it.

6.8 Interoperable Security

In an enterprise grid environment, we cannot assume a homogeneous environment. In fact, in some cases “legacy systems” will comprise part of the enterprise grid. Therefore, support for interoperable security across heterogeneous grid components is a requirement. This may apply to a pool of grid resources that are being managed by the GME, where the GME must be able to uniformly manage the security attributes of all those resources. Or, the various security models being used by the heterogeneous grid components must be able to map to each other in order to support an authentication and authorization framework that applies and works across the entire enterprise grid. While interoperability is a constraint, it should not preclude the use of vendor or product specific features if they can be leveraged within the grid framework. The goal is not to force grid environments to the lowest common denominator, but rather to ensure that products and technologies can be easily integrated into enterprise grid environments.

6.9 Secure Isolation

Since the enterprise grid often aims to satisfy the resource needs of multiple applications and services using shareable pools of grid components, it is important that the same secure isolation requirements can be met that are traditionally satisfied in physically, electrically or logically isolated silo-ed environments. For each grid resource, a company's security policy may state what type of isolation is required (physical, electrical, logical, none, etc.). The enterprise grid must therefore be flexible enough to meet any company's unique policy. The requirement for isolation applies not only to individual grid components but also to collections of grid components and higher order services built upon them.

6.10 Trust Relationships

Like any security infrastructure system, the enterprise grid must have certain trust relationships in place in order to be secure. This includes the relationship between users, administrators, applications, and services to the GME and each grid component. These trust relationships are supported by underlying security mechanisms (such as authentication and having a secure communication channel). They must also be tracked by the GME and the grid components. Examples of this include:

- Which compute, network and storage resources may work with one another?
- How is trust established, maintained and terminated within the grid?
- What is the trust relationship between related or dependent grid components?
- How are trust violations detected? What is done when such a violation is found?

7 Summary

An enterprise grid environment offers many compelling business benefits. By understanding the unique threats and security requirements associated with such an environment, one can minimize risk exposure while taking advantage of those benefits. Furthermore, some of those benefits include enhanced security, particularly in the area of availability. As the Grid Management Entity moves from a combination of people, process, and technology to something more automated, one can also realize the many benefits of centralized security management.

8 Related Work

8.1 Standards Related Work

- OASIS Security. http://www.oasis-open.org/
- OASIS Web Services Security TC. http://www.oasis-open.org/
- OASIS eXtensible Access Control Markup Language (XACML) TC. http://www.oasis-open.org/
- OASIS Security Services (Security Assertion Markup Language [SAML]) TC. http://www.oasis-open.org/
- Liberty Alliance Project. http://www.projectliberty.org/
- Global Grid Forum (GGF) Grid Security WG. http://www.ggf.org/
- OGSA WG Security Sub-Team. http://www.globus.org/ogsa/
- OGSA Security (Open Grid Service Architecture Security) WG. http://www.globus.org/ogsa/
- OGSA Authorization (Open Grid Service Architecture Authorization) WG. http://www.globus.org/ogsa/
- OGSA AuthZ WG. http://www.globus.org/ogsa/
- DMTF User and Security WG. http://www.dmtf.org/
- SNIA Management Protocol and Security TWG. http://www.snia.org/
- Trusted Computing Group (TCG). https://www.trustedcomputinggroup.org/

9 References

- “Enterprise Grid Alliance Reference Model v1.0”, 23rd March 2005, EGA
- “Accelerating the Adoption of Grid Solutions in the Enterprise”, 2004, EGA

Enterprise Grid Alliance
2400 Camino Ramon, Suite 375
San Ramon, CA   94583
Tel: +1.925.275.6644
Fax: +1.925.275.6691
http://www.gridalliance.org