(A preliminary white paper)

SOME SECURITY CONSIDERATIONS FOR SERVICE GRIDS

Martin Milani
Chief Technology Officer
Ibuilding Inc.
martin@martinmilani.com

John Seely Brown
Chief Scientist
Xerox Corporation
Jsb@parc.com

April 2002

In his Harvard Business Review letter written in response to the Hagel-Brown article published in the October 2001 issue of the Harvard Business Review, Mr. Strassman argues a number of points challenging the viability of Web services and cooperative distributed computing across enterprises. His concerns were as follows:

1. Denial of Service (DoS) attacks;
2. Flawed software;
3. XML as a carrier for infectious viruses;
4. Trustworthiness of providers of Web services; and,
5. Wireless security concerns over wireless enabled and mobile software.

In the first section of this paper will first address Mr. Strassman's concerns on security. In the latter half, at a very high level we will discuss how the service grid, as described in the HBR article is, in fact, an important step to addressing many of these issues.

Let us agree, as a preliminary matter, that security is not binary; "you don't either have it or not". Rather than being black or white, security is more like a gray scale – something you might have but can always be improved. It is also a process that constantly needs to be maintained, managed, monitored and enhanced. Security is a multi-tiered concept by which different layers of security are applied to different types of communication. Both coarse grain and fine grain security frameworks adhere to a whole range of security policies, from the most lax to the most stringent. Yes, the Internet is insecure; the question is not whether the Internet is secure, but how will the next wave of software technologies make the Internet far more secure and make such security more reliable, accountable, scalable, extensible and interoperable. Will these improvements be sufficient to allow mission critical business processes to utilize web services as described in the Hagel-Brown article? We expect so but our thinking at this stage is still highly conceptual.

1. Security issues and common attacks

Let us now address the various issues that Mr. Strassman raises.

1.1 Denial of service attacks

There are several types of Denial of Service attacks (DoS), most notably SYN flood attacks, Smurf attacks and Distributed DoS (DDoS). Such attacks either penetrate a system and disable some of its services, such as httpd (http listener), or exhaust the resources of the system by schemes such as bombarding it with traffic and/or a large number of malformed packets, causing the network to be flooded and/or machine resources to be overwhelmed. In most cases, security hackers use the computers of an unsuspecting company to stage such attacks and sometimes spoof or hijack IP addresses of these systems. There is no danger of loss of data or exposure of data behind the firewall to these types of attacks. The hackers do not gain access to data that resides on the systems. Instead, they only cause interruption of service on public systems, including web servers, email servers/SMTP hubs, FTP servers and other services available to anyone from anywhere on the Internet; hence, the term “Denial of Service”.

The communication between the service grid as defined in the Hagel-Brown article, and the enterprise is of a different nature. Crucially, the Hagel-Brown service grid does not offer services that are available to an anonymous user on the Internet. The systems within the service grid, as well as the systems within the organizations that are using the service grid, are protected and behind firewalls. More importantly, the service grid recognizes the partner enterprises and service providers. Thus, only packets from these sources make it through the choke routers as the first line of defense.

Systems can be moderately protected against DoS and different mutations thereof by employing a variety of technologies and strategies, ranging from communicating over private Internets to the use of firewalls and VPNs over the public Internet, some of which are surveyed below as follows.

1.1.1 Private Networks, such as Frame Relay /ATM

Organizations may connect to the service grid via private Frame Relay or ATM service providers. Traffic will not go over the public Internet and is secure from DoS.

1.1.2 Point-to-Point connections such, as T1

Organizations may also connect to the service grid over a point-2-point connection. Using this method, the user's enterprise would be connected to the service grid via a T1 or T3 connection. Traffic would not be traveling over the public Internet and would thus be immune. Private IP services of a service provider could also be employed.

1.1.3 Virtual Private Networks

Another alternative is the use of Virtual Private Networks (VPN) over the Internet in the Site-2-Site configuration. All traffic between the enterprise and the service grid would be encrypted and transported over the Internet. Other services, such as data authentication, guarantee of origin and data integrity, are also provided. Admittedly, VPN connections may get overwhelmed by being bombarded with bogus traffic, however, they can be moderately protected by using choke routers, firewalls and intrusion detection systems.

1.1.4 Firewalls and Access Control Lists (ACLs)

DoS, DDoS and SYN-ACK attacks may also be prevented by implementing a combination of firewalls and ACLS on routers. This method would include the use of choker routers for packer filtering, firewalls and stateful firewalls (firewalls that can actually examine the data content of the packets as well as determine the state of the protocol within which the packet is moving). Any system that sits behind a firewall, as most systems do in an enterprise or the service grid, is fairly but not completely immune to these types of attacks because of the methods described above. A well designed firewall system, through the use of advanced IOS features, multiple choke routers, firewalls nodes and Intrusion Detection Systems with 7x24 monitoring as a managed service, would improve the security substantially and offer a higher degree of protection against these type of attacks. In sections 1.1.5 through 1.1.7 we briefly touch on some of these methods.

There are other types of DoS attacks that are designed to attack networks by flooding them with bogus traffic, which degrades network performance by overloading them. The two main types are Smurf and Fraggle attacks. Generally, in both cases, attackers bombard ICMP Echo Requests to a broadcast address on a network. The source address is sometimes forged and is of a target machine (ultimate target). In return, all the hosts on that network try to respond to the request and bombard the "ultimate target" back with Echo response, overloading the ‘ultimate target” and the network on which it resides. There are several other scenarios, all of which are a mutation of this, using TCP, UDP or both. Again, there are very effective means to combat this type of attack, especially when the service grid is aware of the IP addresses used by the enterprises and vice versa. By using ACLs and many different utilities and schemes available on most IOS running on the choker/border and edge routers, these types of attacks are easily repelled on both sides of the connection, i.e., the enterprise and the service grid.

1.1.5 IPSec

IPSec provides a framework for security for the IP layer, which natively is available in IPv6 and as an add-on to IPv4. It is an open standard developed by IETF. The main services of IPSec are:

Confidentiality:

> The packet header information and the data payload information are protected through encryption such as DES, Triple DES, etc.

Authentication:

> Through the use of PKI technologies to authenticate users and systems identities, integrity and origin may be established.

Replay protection:

> Due to the connectionless nature on IP under certain circumstances, and attacker could continue to re-send a packet to a host that has already received that same packet once and thereby tie up system resources. By providing a packet counter mechanism IPSec protects against this type of attack.

Security Association:

> In order to encrypt and later decrypt IPSec packets between conversations of two systems, an association called Security Association (SA) is maintained between the caller and the called system. SA defines the security services by which the traffic is protected. Its contents include things such as authentication and encryption keys, key lifetimes, etc.

1.1.6 IPv6

IPv6 provides the same security framework defined by IPSec. IPv6 natively includes security headers with relevant key management protocols. The Authentication Header (AH) is used to provide strong integrity service for IP datagrams (content verification data for the IP datagram), that is, strong authentication of IP datagrams, non-repudiation for IP datagrams and protection against replay attacks.

The Encapsulating Security Payload (ESP) allows the encryption of the payload (data fields) carried by the packets. The main services that are provided here are authentication of origin, confidentiality and anti-replay services.

1.1.7 Intrusion Detection

There are many different definitions for intrusion detection. Security, in general, can be broken up in to three parts: prevention, detection and response. Intrusion detection involves actions such as:

-identifying the intruder,
-how and when the intruder entered,
-where did they go,
-what did they do at each stop,
-did they take anything,
-what they leave anything behind,
-where did they come from,
-notification of law enforcement agencies and share of information.

In addition, there are different layers of intrusion detection: network based intrusion detection; intermediate systems such as message queues and email system intrusion detection; and end system/host based intrusion detection. Intrusion detection involves a process that is based on event correlation among all three layers. By offering intrusion detection as a managed service, and as part of the service grid, security experts and intelligent agents can monitor all connections and systems on both ends of the service grid on a 7x24x365 basis. Through proactive event management and event correlations, attacks and suspicious activities can be quickly identified and appropriate and evasive measures can be taken while coordinating with law enforcement authorities. As the service grid is the only connection to the enterprise for specific services, the enterprise has the ability to significantly simplify intrusion detection and filtering, as only certain protocols and packets from the service grid can be allowed. The service grid allows security policies to be focused on a single connection using security technologies specified by the enterprise, of which the enterprise has deep operational knowledge. Contrast this to many point-to-point relationships involving a wide variety of security technologies and standards.

Again, it is important to emphasize that the service grid knows whom the clients are (partner enterprises) and only packets from those addresses make it through. The rest are discarded by the choker routers, before even reaching the internal parts of the service grid. On the enterprise side the routers connecting the enterprise to the service grid also are aware of the IP addresses employed by the service grid and would only accept packets from those known IP addresses.

1.3. XML and viruses

XML may be used to carry viruses, but this has much more to do with the application architecture and communication technology used in conjunction with XML than the XML content itself. By way of example, a simple listener receives an XML document. By using a sandbox scheme the listener quarantines and then examines it's content. Operational tags could be disallowed or authenticated and authorized for accepted operations. The clients (initiators) and services servers (responders) could be strongly authenticated at the application layer and the identity of the caller programs verified and validated through directory services, such as LDAP or X.500 and certificate of authority such as X.509 along with their authority and access levels.

There are numerous design schemes, architecture and frameworks for distributed systems that offer a very high level of security and would protect applications and systems from malicious code embedded within an XML message or document. We are not arguing that viruses can be avoided completely but that systems can be designed to offer a high degree of protection from common virus schemes. We will touch on some of the general security services that should be employed at the application layer without going into detail of security aware distributed systems architecture.

1.3.1 Identification and authentication

The users and calling objects/calling programs are authenticated through open systems standards such as Kerberos, X.500 and X.509 (PKIX).

1.3.2 Authorization and ACL

ACLs may be used to Check a user or an object or a program acting on behalf of a user requesting some service and cross-checking it with user attributes, access controls and security profile policies so as to grant or deny the request.

1.3.3 Auditing

Auditing service logs all interactions and communications with the security service/subsystem.

1.3.4 Confidentiality

Communication between programs and objects can be made secure through encryption.

1.3.5 Delegation

Managing Access controls and authorization levels when a client program requests some service from an object/program which does not process the request to the end itself, but instead calls another object/program. Security polices need to move down the chain and be enforced.

1.3.6 Non-repudiation

Non-repudiation service provides evidence, proof of creation of a message, and the proof of receipt of that same message between two systems.

1.3.7 Digital signatures

Code, files and XML files may be digitally signed to ensure authenticity and integrity.

1.3.8 Reliable Message Transport

Messaging transport protocols may be used that offer reliable and guaranteed message delivery by using technologies such as persistent message queues. We will describe this scheme further in the service grid architecture section of this paper.

1.4. Trustworthiness of providers of Web services

There are different degrees of trust (i.e., trustworthiness) that are associated with different entities and the different transactions among those entities. Further, trustworthiness is built over time rather than negotiated or dictated. The service grid is an excellent mechanism to act as the trust broker for all the participating partners. Trust brokers operate by aggregating all relevant information about all participating parties, both as the parties initially register with the service grid, but also, through time, as the 'trust broker' can oversee and help to warrant the accuracy of the stated capabilities of services being offered by the parties. We stress that managing trustworthiness is not a one-time issue, but rather is something that evolves over time. The service grid is an excellent nexus for monitoring and evolving the reputation of various application services of the service grid. The service grid is also in a privileged position to validate and improve/enhance the information in the UDDI /WSDL/WSCL registries, since all exception conditions can be detected, examined and refined by the providers of the service grid if the parties so desire.

1.5.Wireless security concerns

Wireless devices, acting as mobile clients will go through the same security frameworks that other clients go through. Through a combination of user and device authentication the same level of security can be achieved as non-mobile clients. Connections could close very quickly due to very sensitive inactivity timers on these types of connections, prompting the user to re-log in if the inactivity timers have expired.

2. The service grid security considerations

We will now discuss how utilizing the service grid simplifies the complexity involved in connecting partner enterprises together. Suffice it to say in this paper that managing and maintaining the connectivity to partners and service providers is far simpler and cost-effective by connecting to a service grid rather than by directly connecting point-to-point to all these partners and service providers.

Today, to integrate two different security architectures, they must be integrated at all levels, the authentication information must be shared, the mechanism must be integrated, the data encryption policy must be shared and there must be shared ACL. Due to this complexity, most point-to-point integration efforts run into trouble either because they are very costly and unmanageble, or because they simplify the integration in a way that reduces security. The barrier to secure integration is very high in a multi partner scenario, as the overhead of managing multiple connections and multiple security policies are expensive, cumbersome and can quickly become overwhelming.

Another important point is that mediation services are provided by the service grid. Service grids, through their ability to natively integrate with each partner and provide mediation between partners at a business level, can significantly improve security and perhaps, more significantly, improve visibility of security. The decoupling of security through the service grid provides the ability of individual enterprises to constantly improve and upgrade their security without being constrained by their connections to the other partners. This also allows each organization to be moderately autonomous in terms of the use of preferred security technologies (eg. MIT Kerberos, Microsoft Kerberos, X.509, etc.) and products as well as enforcing and maintaining local security policies, procedures and policy hierarchies, thereby allowing a federated security ecosystem among service providers and consumers.

In the case stated above, each enterprise manages and maintains only one connection to a service grid instead of having to deal with all the overhead of managing multiple connections and security policies. Further, maintaining and enforcing consistent security policies are far simpler and more flexible through the use of the service grid.

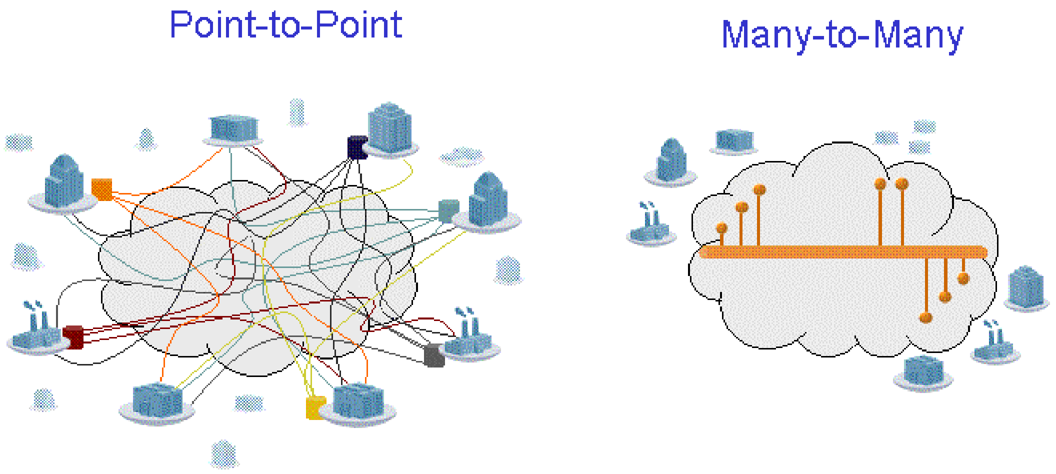

Figure 2.0

Let us now turn to some architectural aspects of security and security management within the service grid. In doing so, we will not try to define the architecture of the security service grid in depth, but rather at a very high level give a glimpse of mature and proven techniques and technologies which could be used to construct the security frameworks within the service grid.

As a preliminary matter, our position is that the security core of the service grid – which we will call the security grid – (and the service grid, itself) do not need to be based on webservices architecture as defined by webservices specifications today. Webservices specifications are often moving targets, work in progress, and sometime there are multiple competing submissions to the standards body by different groups and alliances (i.e. WSFL, XLANG, and WSCI). Such fragmentation will slow the adoption of certain standards quite a bit or even derail some of these efforts. But with respect to security, the security services in traditional distributed and services-based architectures are proven, mature, well defined and understood. The security grid would expose the aforementioned security services to the node-enabled enterprise and service consumers. This approach preserves existing security infrastructures and architectures already employed by the enterprise. It also speeds up the adoption of webservices by organizations, as they do not need to wait for standards and the expensive and cumbersome task of replacing already existing technologies and infrastructures.

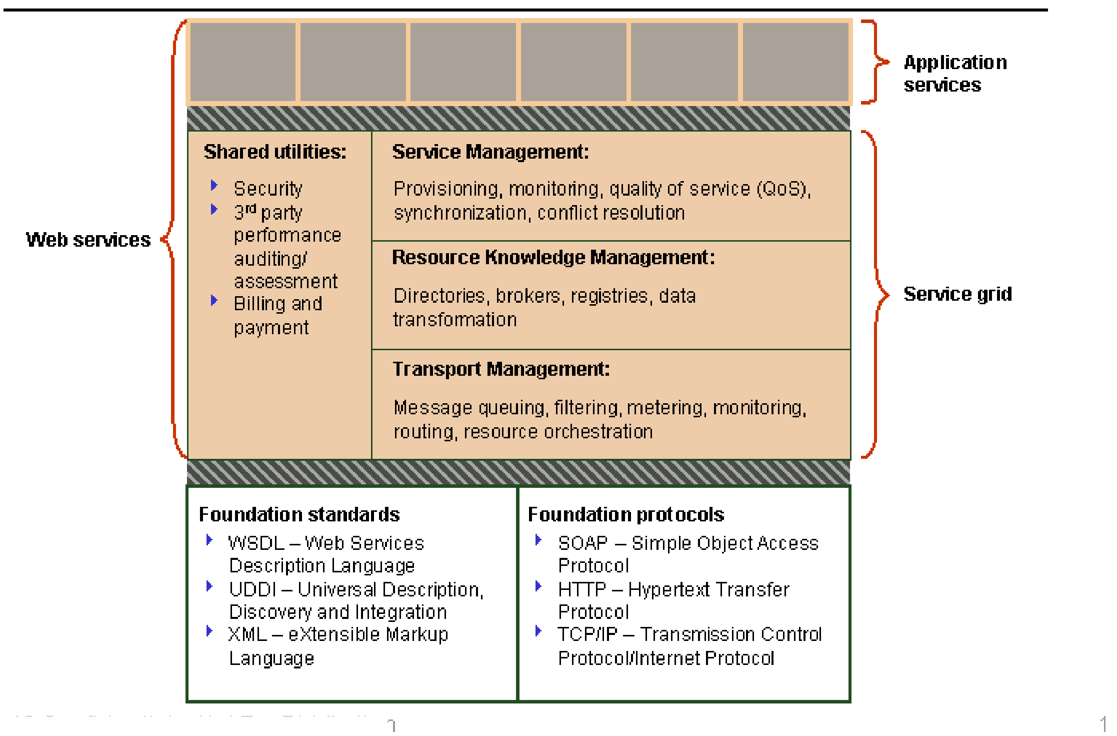
Application services
Web services
Service grid
Shared utilities:
Security
3rd party performance auditing/ assessment
Billing and payment
Service Management:
Provisioning, monitoring, quality of service (QoS), synchronization, conflict resolution
Resource Knowledge Management:
Directories, brokers, registries, data transformation
Transport Management:
Message queuing, filtering, metering, monitoring, routing, resource orchestration
Foundation standards
WSDL – Web Services Description Language
UDDI – Universal Description, Discovery and Integration
XML – eXtensible Markup Language
Foundation protocols
SOAP – Simple Object Access Protocol
HTTP – Hypertext Transfer Protocol
TCP/IP – Transmission Control Protocol/Internet Protocol
1

Figure 2.1

Although it has been argued by some of the players in the web service space that web services will conform to peer-to-peer architecture, our belief is that web services will not evolve to be primarily peer-to-peer, at least not for mission critical business processes. New software frameworks and technologies are required to provide loosely coupled but more secure and predictable connections among mission critical applications. In fact, web services technologies require significant enabling services to provide effective support for mission-critical business processes. These enabling services cover a broad range of needs and include asynchronous message queues, long and short lived transaction management, metering, accounting, billing, quality of service, traders, brokers, directory services, xml transformations, security and security management and many more.

Advanced and comprehensive security frameworks must be interjected in all aspects of distributed architecture. One main issue with security, in general, is that some organizations are very secure and some are not., that is, there are many different security schemes and different levels of security employed by the organizations that are providing application services to each other. Another point is that different security technologies are used by different organizations. In fact, one of the most pressing issues around cooperative distributed computing among enterprises is the lack of multi-layered standard security practices, policies and common security frameworks. However, by introducing a comprehensive multi-layered security framework within the service grid, standard security practices and policies can be governed, managed and maintained across all parties using different security technologies and products. Creating a standard set of practices and policies is especially crucial when considering that orchestration of services is involved. Sure, the service one requests has these policies, but what about the other services being called in the background by that service. And that is just the tip of the iceberg, all of which must be monitored and checked by the provider of the service grid.

The security framework for the service grid adheres to the service grid schema defined by the Hagel-Brown article, as illustrated below.

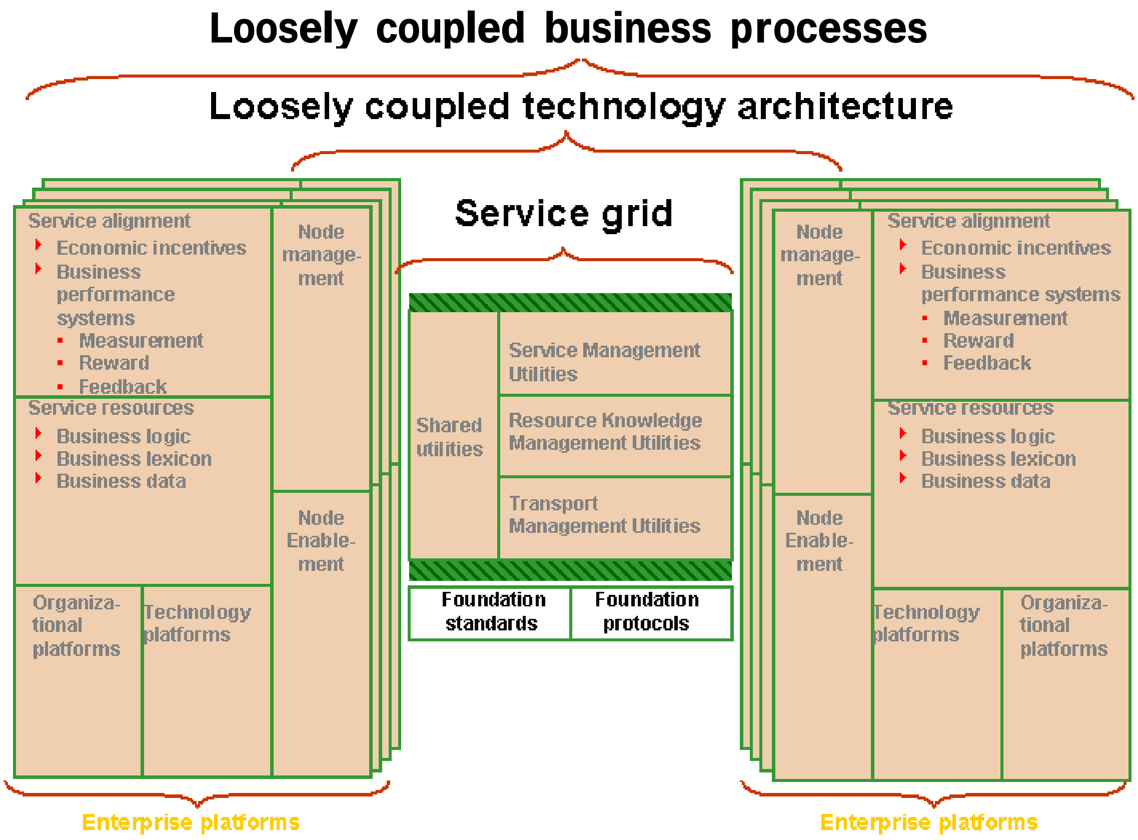

Figure 2.2

As we suggested before, one can look at the security framework within the service grid as a security grid within the service grid. In the distributed software world, we can no longer afford to have security policies and logic in different forms and degrees to be embedded within the application logic of each application or each layer. Security must be enforced, managed and audited end-to-end by employing a multi-layered and multi-tiered approach. The security services within a security grid offer a set of core services which we will call Core Security Services (CSS) that provide confidentiality, integrity, accountability and availability. The CSS provides:

-Identification and authentication (user /object);

-Authorization and access control (user/object);

-Security policy management and profiling (user/object);

-Auditing;

-Secure communication (user/object);

-Non repudiation;

-Delegation; and

-Administration,

The users and services are large in numbers and dynamic. Users, attributes, authorization levels, services, functionality and security policies are changing all the time. Security association is maintained and managed by the service grid between initiators (users and client programs) and responders. Security brokers negotiate on behalf of these systems and employ the appropriate security mechanisms and products based on security policies, profiles and degrees of trust that are mostly not constant and are often changed and updated. Logically, and we emphasize, NOT physically, the service grid could be imagined as a directory consisting of all the users, systems, resources and services. This directory would also include all of the security constraints, policies, procedures and possible relationships among them, where all interactions from the security perspective are mediated, orchestrated and managed by the security service grid.

The anatomy of the service grid itself and the node-enabled enterprise with respect to distributed application layer security would logically be broken into three security tiers as follows (we of course are assuming that distributed applications are distributed across multiple systems and networks):

a. Perimeter security;

b. Middle-tier security; and

c. Backend systems security.

The security grid would offer a set of services that provide the functionality needed to protect each of the above mentioned tiers pertaining to the service grid itself. The service grid also offers orchestration and mediation across these tiers through the use of security brokers which mange security policies and profiles end-to-end. These services use the core services that are available to them through the Core Security Services.

The network on which the service grid resides would have the three layers described below (the same scheme could be utilized by the node-enabled enterprise):

a. the first level network consists of systems such as Web servers, or systems that directly communicate to client programs;

b. the second level, or middle-layer network, consists of systems such as application servers that maintain application components , business objects and application logic; and

c. the third level, or backend network, consists of systems on which persistent data resides, such as database servers.

Within the service grid, the perimeter security policies would apply to first level networks and systems. The middle-tier security policies would apply to second level network and systems. And, the backend system security policies would apply to third level networks and systems.

In the diagram below, we have set forth a simple example of a multi-tiered security framework that could be employed by the service grid and/or partner enterprises. This diagram does not address the communication between the service grid and its partners and service providers, but highlights instead the multi-tiered approach to security to be employed within the service grid and/or the enterprise. The clients are in fact client programs that request some service, which in this particular example are using https as the protocol, but these types of communications are not and should not be limited to http.

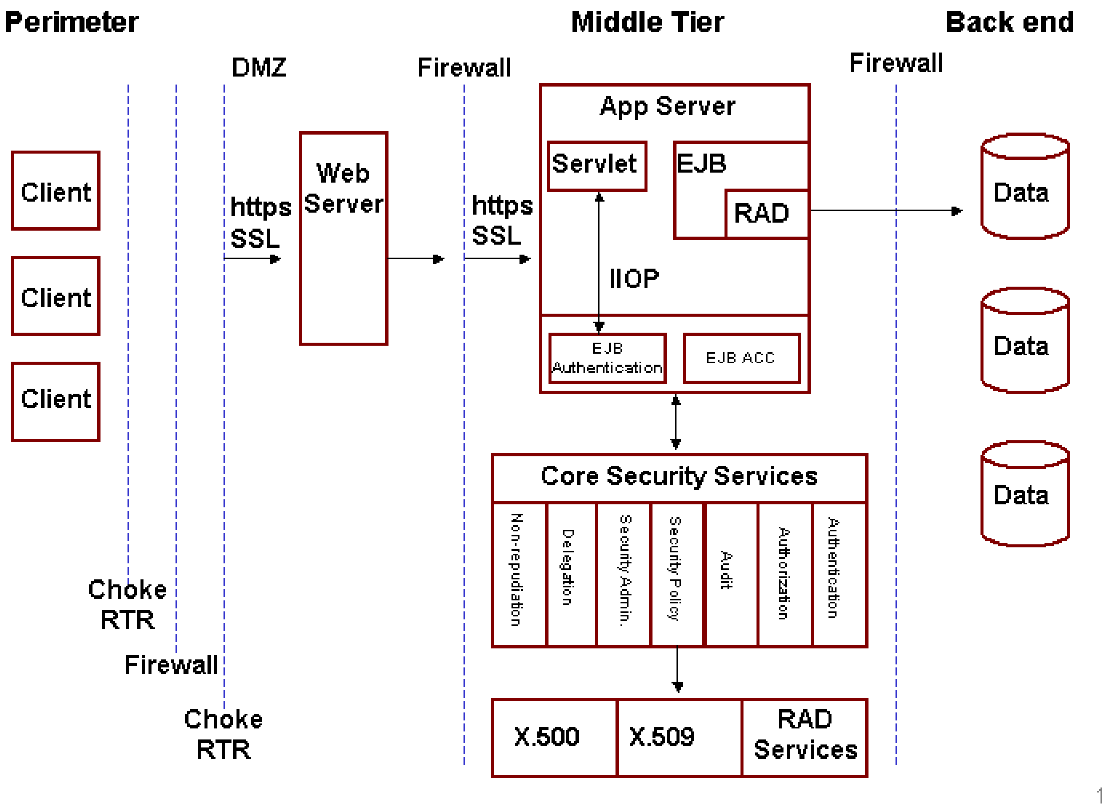

Figure 2.3

At the edge of the service grid lies the node-enabled enterprise. Let us digress briefly and discuss “node enabling the enterprise”. There are internal and external factors which drive enterprises to communicate in a much richer way with others such as business partners, third party service providers, customers and so on. In order to proceed with this manner of communication, these enterprises must expose some of their applications, systems and processes to others as services. This situation could also get compounded with software vendors selling/renting some of the software to be used by these enterprises over the Internet, traditionally known as “webservices”. Enterprises and their respective IT departments would have to go though a metamorphosis, as discussed in the Hagel-Brown article, which we call “node-enablement”.

In order for the enterprise to expose itself to the world of partners, customers and third party service providers in an efficient and meaningful manner, two major steps must be taken:

1. Create an object based abstraction layer that sits on top of internal systems. The abstraction layer will provide connectivity between various applications that are employed within the enterprise. These applications will communicate with each other only through this abstraction layer and will no

longer communicate directly through dedicated point–to-point connections. It is intended that the abstraction layer will manage the aspects of security across all the applications.

2. Once the internal abstraction layer is defined, constructed and operational, some services and systems can be exposed to the external world. We argue that these interfaces should not be exposed directly. These services will become exposed to an external object based abstraction layer only, which is also maintained by the enterprise. The external abstraction layer will sit on a DMZ on the edge of the enterprise. It is the external abstraction layer that will communicate to the service grid. This would be very similar to how internal and external DNS is managed by organizations today.

In the diagram below, we have set forth a logical view of how we believe this vision might evolve over time. The external hub, residing on a DMZ of an enterprise, acts as a proxy and a message broker that communicates with components that are exposed to it by the internal hub residing inside the firewall of the enterprise. The internal hub communicates with the middle-tier of the enterprise that communicates with enterprise applications APIs directly In this scenario the enterprise systems are 3 steps removed from the actual service consumers in the outside world. The service brokers acting as traders, broker the communications and access between service consumers, the service grid (and the security service grid) and the service providers based on identity, authorization, context, security clearance, degree of trust, quality of service, type of service, price, bandwidth, protocol desired, etc.

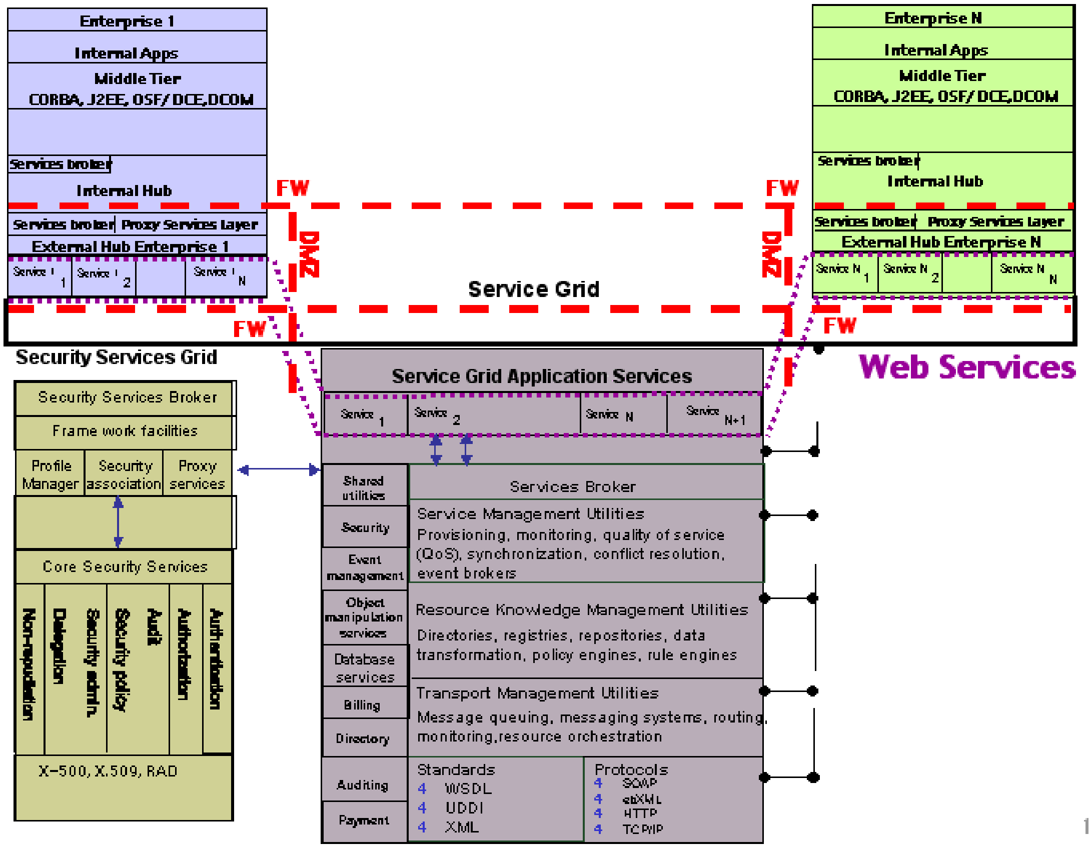

Figure 2.4

At a very high level through the use of choke and edge routers running IPSec/IPV6 (both sides), the traffic flow between the service grid and the enterprise is protected. By using an PKI infrastructure, authenticated traffic will be routed across the service grid between separate VLANs, which contain different tiers of systems (first level, second level and backend) based on policies and use of policy-based routing (PBR). State-full firewalls check all packets and their contents flowing between different VLANS.

Through the use of the service grid, communication between the service grid and clients or requestors can be of a very simple nature. UDDI, WSDL and SOAP can be used to advertise and expose these services to the enterprise. A security proxy service will enforce the security policy based on type of service requested and existing security policies and invoke the appropriate security services required for authentication, authorization and encryption. SSL/TSL and/or IPSec based schemes will protect the communication. After this juncture, based on the identity, type of request and authorization access policies, the caller is granted access to appropriate ORBs or objects/methods/APIs that offer those services. All communications between objects and ORBs/programs within the service grid itself are authenticated, authorized and encrypted. PKI and directory services such as X.509, X.500 (LDAP) and SSL are used to achieve this.

Transport services, such as asynchronous message queues with persistent queues can be used to provide high availability as well as guaranteed delivery of messages between clients and the service grid when needed. In case of network or systems failure, messages are guaranteed to reach their destination in order and will not be lost. Digitally signed code, files and XML working in conjunction with PKI technologies are used to further protect against malicious and unauthorized code.

Of course, this is a hybrid approach, in which multiple protocols would be supported and managed for communications between the service grid and the enterprise. Depending on the security policy, quality of service, trustworthiness and nature of the communication, the appropriate protocol would be used. These protocols would include SOAP, SOAP extensions that would facilitate certificate/token exchanges, IIOP, and through JMS interfaces, asynchronous message queues.

By interjecting security at different layers such as IP, TCP, Data and application and providing the framework in which each layer is conscious of all other layers, it is possible to achieve a very high degree of security. One can even imagine using XML encryption and digital signature along side with SSL/TLS, and IPSec/IPV6 in transport or tunnel mode at the same time. Of course this might have a high toll at the expense of performance but for a very high degree of security.

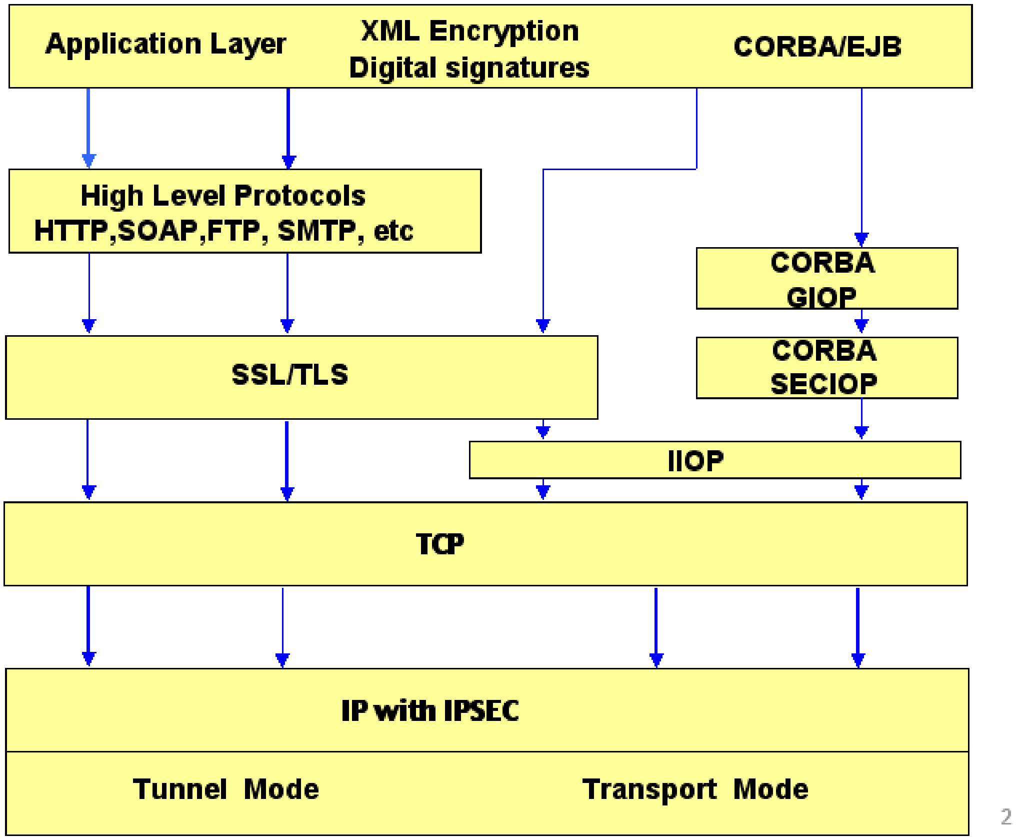

Figure 2.5

The information would be protected not only while in transport across different organizations but in fact protected end to end against intrusion and eavesdropping inside the organization’s firewall (inside hacking). The service grid can use any or any combination of these methods based on the security policies that are negotiated between the service providers and consumers and the service grid combined with the quality of service (security level), degree of trust and the sensitivity of data and communication channels on a case by case basis dynamically.

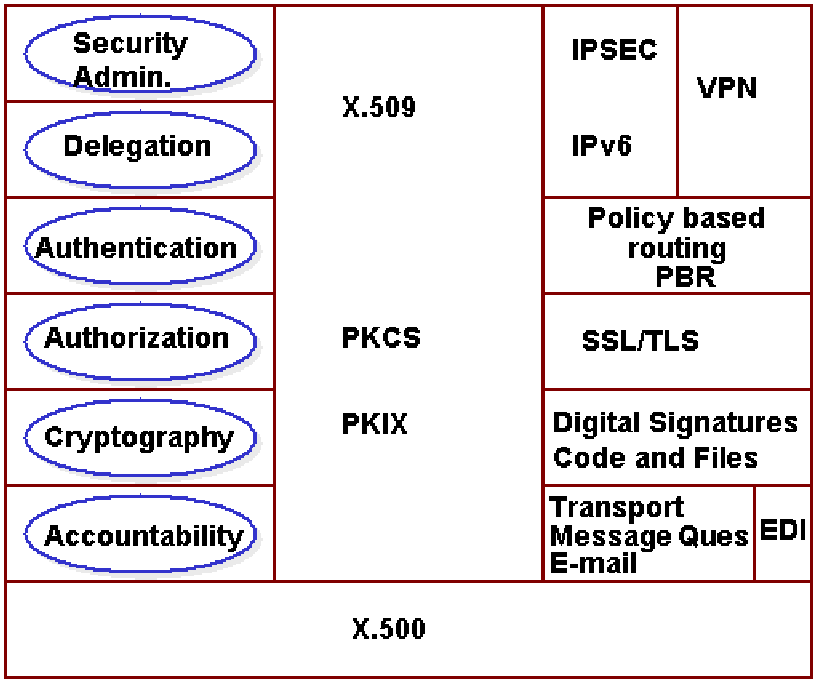

Figure 2.6

This level of completeness, complexity and cohesion is an absolute necessity for the future of a meaningful web services world. However, through the use of the service grid, this complexity is transparent to the end users and enterprises. This highly complex and complete security framework is moved to the center from the edge, thereby keeping the edge simple! It also allows much innovation on the edge without compromising security. The communication from the edge to the service grid, in fact, could be using very simple technologies such as SOAP/XML/HTML without the need for supporting and understanding the complex security models and techniques that reside within the service grid. There are many new initiatives by organizations such as OMG under way to define the next generation of SOAP and XML technologies as well as to better integrate them with already established and mature distributed computing technologies such as OSI, CORBA, JAVA/EJB and IIOP. SOAP and XML will evolve as time passes and become much more complete and security conscious. In fact, as we were writing this paper, a new security specification, WS-Security was submitted for comments by IBM, Microsoft and Verisign. This specification proposes a standard set of SOAP extensions that can be used when building secure Web services to implement integrity and confidentiality. It is designed to be used as the basis for the construction of a wide variety of security models including PKI, Kerberos, and SSL. It provides support for multiple security tokens, multiple token domains, multiple signature formats, and multiple encryption technologies. We believe that this is a very important first step, but as we have discussed, message authentication, integrity and confidentiality at the application layer is only a subset of the enabling technologies that are needed.

With rapid growth in bandwidth, computing power and memory bandwidth on new servers and systems combined with the decrease in the cost of bandwidth, one can imagine that in a few short years, stateless types of protocols such as HTTP and SOAP, will be replaced by much more complete and rich set of stateful and synchronous protocols which will reshape the nature of distributed computing and mission critical web services over the Internet.

In closing, we admit that the above is but a sketch of a vision of providing a level of security for a distributed service architecture that is mediated by a managed service grid construct. Crucial to our thinking is to find a framework that enables consumers (enterprises) to keep their own node enablements for web services simple yet to have access to whatever level of security is deemed appropriate for however they are using a shared web service within their own business processes. The service grid provides the mediation and visibility of security technology allowing secure integration between enterprises to be a policy decision. The decoupling of security technology by the service grid allows the rapid evolution of secure communication between enterprises and will result in a higher level of security rather than the opposite. We also do not claim that the current protocols in use today for invoking web services (e.g., SOAP) have all the appropriate security layers, but rather that these protocols can be progressively refined along with the emerging capabilities of service grids to provide whatever level of security is needed.

Acknowledgements

We thank John Hagel, Avisheh Avini, Ken West and Stan Raatz for their valuable input.

Glossary

ACL	Access Control List
ATM	Asynchronous Transfer Mode
CA	Certificate of Authority
CORBA	Common Object Request Broker
DES	Data Encryption Standard
DoS	Denial of Service
DDoS	Distributed Denial of Service
EDI	Electronic Data Interchange
EJB	Enterprise Java Beans

ICMP	Internet Control Message Protocol
IIOP	Internet Inter-ORB Protocol
IP	Internet Protocol
LDAP	Light weight Directory Access Protocol
OMG	Open Management Group
ORB	Object Request Broker
OSI	Open Systems Interconnect
PKCS	Public Key Cryptography Standards
PKI	Public Key Infrastructure
PKIX	Public Key Infrastructure X.509
RAD	Resource Access Decision
SOAP	Simple Object Access Protocol
SSL	Secure Socket Layer
TCP	Transmission Control Protocol
TLS	Transport Layer Security
UDP	User Datagram Protocol
UDDI	Universal Description, Discovery and Integration
VLAN	Virtual Local Area Network
WSCI	Web Services Choreography Interface
WSCL	Work Spaces Coordination Language
WSDL	Web Services Description Language
WSFL	Web Services Flow Language
XML	Extensible Markup Language
X.500	ISO 9594-1, CCITT X.500 Directory Services Standard
X.509	ISO 9594.8 CCITT X.509 Authentication Framework Standard

John Seely Brown was the director of Xerox PARC until 2000. He continues his personal research into digital culture, learning and Web services. His most recent book (co-authored with Paul Duguid) is The Social Life of Information. He can be reached by e-mail at jsb@parc.com.