

# **Safeguarding the Corporate Portal: A Review of Portal Security**

by Colin J. White

DataBase Associates

Version 1, January 2001

Sponsored by Viador Inc.



---

# TABLE OF CONTENTS

<b>PORTAL SECURITY: UNDERSTANDING THE OPTIONS:</b>	<b>1</b>
Portal Architecture and Operations	1
Portal Security Requirements	3
Web User Device Security	3
Portal Network Security	5
Web Server Security	5
Portal Server Security and Auditing	5
Business Content Security	6
Security Technologies	6
User Authentication	6
Web Device Security	8
Web Network Security	9
Web Server Security	10
Portal and Business Content Security	11
Auditing and Logging	11
Putting It All Together	11
<b>PRODUCT EXAMPLE: VIADOR E-PORTAL</b>	<b>12</b>
E-Portal Architecture and Operations	12
E-Portal Security Features	13
Web User Device Security	13
Portal Network Security	15
Web Server Security	15
Portal Server Security and Auditing	16
Business Content Security	16
Summary	17

*Brand and product names mentioned in this paper may be the trademarks or registered trademarks of their respective owners.*

---

# PORTAL SECURITY: UNDERSTANDING THE OPTIONS:

**Security becomes a key issue as portals evolve to support a widening user base and ever-increasing back-end business content**

The use of portals for providing business users with a secure and personalized Web-based interface to corporate business content is growing rapidly and portal products and technologies are evolving quickly to support this growth, and to meet the demand for improved product functionality. One area that is receiving significant attention is portal security. Initially, most portals addressed the needs of a specific department, and were focused toward the internal users of an organization. Recently, however, companies have begun to develop enterprise-wide portals that support a wide range of corporate users and back-end business content. Many of these organizations are also expanding their portals to allow access by external users such as trading partners and key clients. Security becomes a key issue as portals evolve to support a widening user base and ever-increasing back-end business content. In this paper we explore the operation of a portal and review requirements for providing secure access to the business content viewed through a portal. We also look at current security technologies, and discuss how portals should work seamlessly with other IT subsystems to exploit these technologies and provide a secure portal operating environment. Finally, we review Viador's E-Portal to explain how one leading portal product supports secure access to business content for both internal and external business users.

Before exploring portal security in detail we will first review the architecture and operation of a portal so that we have an underlying framework for discussing portal security requirements.

## PORTAL ARCHITECTURE AND OPERATIONS

**An information directory records metadata about the business content that can be viewed through a portal**

The main requirement of a portal is to provide business users with an integrated, secure and personalized view of business content (information, applications, expertise, etc.). The architecture and services of a portal that supports this requirement is shown in Figure 1.<sup>1</sup>

At the heart of a portal is an *information directory* that is used to record metadata about the business content that can be viewed through the portal. This metadata includes information such as the name and location of the content and details about its business meaning and usage. The information directory metadata can be maintained manually by business users using *publishing services*, or automatically by *portal adapter tools* that regularly scan and analyze business content on servers identified by the portal administrator. Metadata is extracted from the business

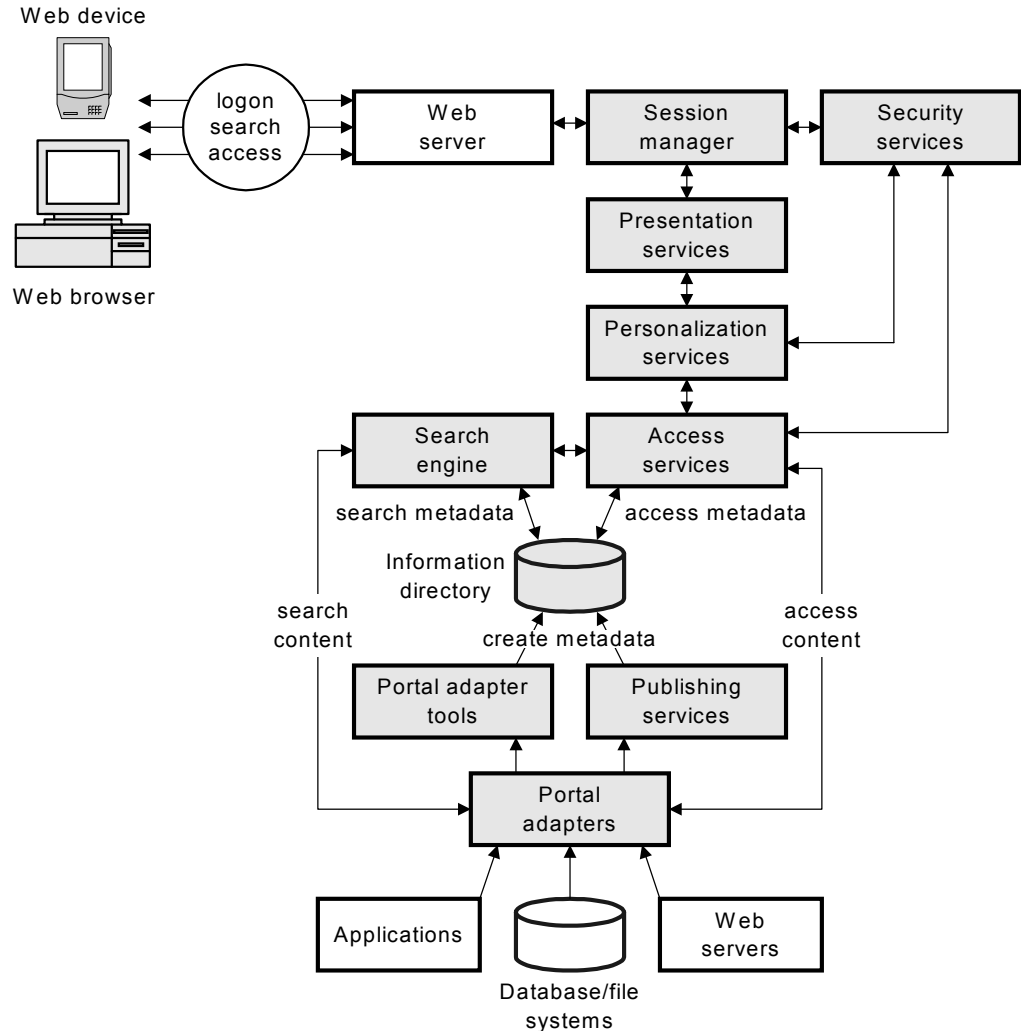
---

<sup>1</sup> Figure 1 shows only basic portal services for searching and accessing business content. Many portals provide additional capabilities such as subscription, delivery, collaborative and workflow services. For simplicity these services are not shown in the figure. The security requirements and facilities outlined in this paper, however, would also be used to manage these additional services. It should also be noted that the discussion of portal operation is generic – actual implementation will vary by product.

content using off-the-shelf or custom *portal adapters* that have been tailored to support the many different types of content that may exist in an organization.

**User access to information directory metadata must be controlled**

Metadata in the information directory is usually organized in a hierarchical set of subject-based folders. Information about sales data could, for example, be stored in a folder named *sales*. Folders in the information directory may be accessible to any portal user or may be restricted to specific users or user groups. Control over which information directory folders can be accessed by the user is managed by *personalization services*.



**Figure 1. Portal architecture**

**Information directory URLs point to business content**

Business users discover relevant corporate business content by navigating folder entries in the information directory using *access services* or by employing a *search engine*<sup>2</sup> to find folder entries that match user-supplied search criteria. Web URLs

<sup>2</sup> Some search engines can also be used to search the business content.

associated with the folder entries are then used to access the actual business content. This latter processing is handled by *access services*, which invokes the appropriate *portal adapter* to connect to the business content.

**The direction of Industry is toward providing mobile users with access to portals**

Information directory entries and business content are presented to the business user via a Web-based interface. The formatting of information presented through this interface is handled by *presentation services*, which may employ a variety of different technologies including HTML, XML/XSL, Java or ActiveX. In most situations, the user's Web interface will be a Web browser running on a desktop computer. Current growth in the use of mobile and wireless computing, however, will lead to a wide variety of different Web devices being used to access portals over the next few years.

**The portal session manager handles user authentication**

A *session manager* handles all interactions between the portal user's Web device and the portal server. One of the main objectives of the *session manager* is to work in conjunction with *security services* to handle user authentication and to provide a secure portal operating environment.

## PORTAL SECURITY REQUIREMENTS

**The portal security service should work in conjunction with other system security mechanisms**

Before beginning a detailed discussion of portal security requirements it is important to first review the objectives of a portal security service. In a portal environment two things are at risk: the *business content* that can be accessed by the portal, and the *computer resources* that are employed by the portal. The objective of a portal security service is to work in conjunction with other system security mechanisms to protect the security, integrity and availability of a portal's business content and underlying computer resources.

We now move on to discuss the facilities that are required to ensure a secure portal operating environment. These requirements are summarized in Figure 2. The next section of the paper then reviews potential security technologies for implementing those requirements. We assume throughout the discussion that a portal could be accessed by users on a secure corporate intranet and by users who connect via the public Internet.

### Web User Device Security

**The portal user's Web device is a major security risk**

The Web device employed by the business user to access a portal is one of the more vulnerable components of a portal system from a security perspective. One key requirement here is to prevent unauthorized users from using a Web device to access a portal. This is achieved by the following:

1. The portal user should be required to sign on and be authenticated when initiating a new portal session. Electronic and biometric devices such as a smart card or fingerprint reader could be used to add an additional layer of sign-on security protection.
2. The portal user's ID and password should be protected in a secure directory server.

3. The portal server’s session manager should employ a session ID to authenticate all interactions with the Web device during the life of a portal user’s session.

**User authentication may be required to access both the Web device and the portal**

Sign-on access to the Web device itself may also be required to protect information that is retrieved from the portal and stored on the device. An alternative option is to place the Web device in a secure area, but this will be increasingly more difficult to achieve as business users become more mobile and begin to use wireless Web devices.

Portal System Component	Security Requirement
Web user device	Sign-on and authentication of portal users Secure directory server to administer portal user IDs and passwords Session manager to manage portal user sessions Sensitive data (password, URLs, etc.) scrambled on portal user’s display Secure use of Web device software
Portal network	Secure message traffic
Web server	Secure use of Web server software
Portal server	Portal server security is integrated with the system software Access to information directory metadata is controlled at a user or user group level Access to portal services controlled by user or user group Logging of security violations and session data for auditing purposes
Business content	Security used to access business content should be transparent to the portal user Portal business content authentication scheme should support the customer’s infrastructure for managing security

**Figure 2. Portal security requirements**

**Active portal sessions are a security exposure**

User authentication does not solve the security exposure caused by the user leaving a Web device unattended in the middle of a portal session. This exposure can be reduced by the session manager canceling the session after a period of inactivity and requiring the portal user to sign-on again. Although the security exposure is reduced, it still remains until the user’s session is canceled.

A more difficult security issue is the protection of confidential information displayed on the Web device's screen. Although it is impossible to completely protect all the information displayed, it is important that a portal hide or scramble key information such as user passwords and the Web URLs that point to confidential material.

**Software running on a Web device can cause a security exposure**

Another potential risk is a breach of security caused by the software running on the Web device. Web browsers, for example, are notorious for security loopholes. They may expose user IDs and passwords, save important information in *cookies* that may be passed back across the network, or allow rogue viewers and plug-ins to be installed that violate security. Setting the appropriate options in the Web device software can eliminate many of these problems, but portal code running on the Web device should also protect against such security violations by judicious use of the underlying capabilities of Web device software.

### **Portal Network Security**

**Messages flowing across the network need to be protected**

A variety of different protocols are used for handling the interchange of messages across a network between Web devices and the Web server running the portal software. One common protocol that is used, for example, is the text-based protocol HTTP. Messages flowing across a network may contain information such as the network address of the Web device, the user's ID and password, the ID of the portal session and the business content retrieved using the services of the portal. Exposure of this information to non-authorized parties could lead to a breach in security, and it is essential, therefore, that these messages are protected from unauthorized access.

### **Web Server Security**

**Software running on a Web server can cause a security exposure**

Most portal products run on and exploit an underlying Web software and hardware infrastructure provided by third-party vendors. Like other Web applications, portals can be affected by security loopholes in this infrastructure. We have seen already how important it is to protect both the user's Web device and the network that connects this device to the portal server. The portal server itself also needs to be protected. The architecture of both the Web server and the portal software running on this server should be designed to prevent security breaches from other non-portal users and applications that may coexist on the same server. This will become especially important for companies that outsource portal operations to third parties such as application service providers (ASPs).

### **Portal Server Security and Auditing**

Our discussion of security requirements has focused so far on preventing breaches of security that are primarily external to a portal. We now move on to look at security requirements that are specific to the portal itself. A key requirement here is that the security facilities provided by a portal product can be integrated seamlessly into the security infrastructure of the underlying operating system and Web server software.

**Access to a portal's services and information directory must be controlled**

The two main resources of a portal server that need to be protected are the individual services provided by the portal (publishing, searching, administration, etc.), and the metadata stored in the information directory. Authorization to access a specific resource should be managed at an individual user-ID level, so that administrators can control which portal services a user can employ and the folders in the information directory that can be accessed or searched by a user.

**The authorization scheme should be capable of isolating users groups from each other**

For ease of administration, portal services should be capable of being grouped together based on the needs of different types of users. The authorization scheme should then allow the administrator to grant users access to either a specific portal service or a group of portal services. Similarly, the portal should also support user groups and have the ability to manage authorization at a user group level. The ability to isolate a user group and the resources it can access is particularly important in an ASP environment where it is likely that a portal server may manage users and business content from different independent organizations.

Metadata in an information directory folder may contain sensitive information, and authorization to access an information directory folder must be coordinated with the authorization to access the related business content. Users should not be permitted to view metadata that is related to business content they are not authorized to access.

**Security violations and user sessions should be logged**

No security service is perfect and it is important that a portal provide a logging service that tracks user sessions and security violations. The recorded information should be sufficient to enable auditors to validate the effectiveness of the security service, and security administrators to assess the impact of security violations.

### **Business Content Security**

**Single sign-on to the portal and back-end business content should be provided**

A portal may provide access to a wide range of different back-end business content including intranet sites, business intelligence tools, office systems, database servers, and operational and e-business applications. The software used to manage this content typically provides its own security mechanisms to protect resources (information stored in a database or application transactions, for example) and to control user and application access to those resources. A portal, therefore, will need to be authorized to access a back-end system to retrieve metadata or business content or to invoke an application service. The portal user must be isolated from the processing involved in authenticating portal access to a back-end system. The user should not be asked to supply a user ID and password each time a back-end system is accessed – this should be managed by security profiles defined to the portal. To prevent duplication of security definitions and to enable an organization to move to a single user sign-on portal security architecture the portal's authentication scheme should support the customer's infrastructure for managing user IDs and passwords.

## **SECURITY TECHNOLOGIES**

In this section we review security technologies that can be used to implement the portal security requirements outlined in Figure 2 above. The objective here is not to do an exhaustive study of every possible security option, but rather to provide a brief overview of some key security technologies that are directly applicable to a portal operating environment.

### **User Authentication**

**User IDs and passwords are usually managed by a directory service**

User authentication is the task of determining and validating the identity of a user who wishes to access a computer resource – in this case a portal server. The most common form of authentication is a password. When a portal user types in a user ID and password the authentication service checks that the user has entered a valid password for the user ID. The user ID and password are normally managed by a

*directory service*, which is accessed by the authentication service to validate the password. More sophisticated authentication methods based on cryptography, one-time passwords, or hardware token cards that generate a time-varying password can also be used. Some of these latter approaches validate every user request to ensure that the request is coming from the user who originally signed on. Kerberos, for example, uses cryptography and a session identifier known as a *ticket* to manage this process.

**It is important to avoid having to define user IDs and passwords to multiple systems**

Before signing on to a portal server, the user may have to sign on first to a local Web device and/or the Web server running the portal software. Although a portal may also need to authenticate the user, it should nevertheless coordinate its authentication and directory services with that of the underlying system software. This eliminates the need to maintain duplicate user IDs and passwords in both system software and the portal software. The need for users to sign on to multiple systems (and also possibly to have to remember multiple user IDs and passwords) is a growing problem and many IT organizations are beginning to implement single sign-on architectures and products to resolve this issue. In addition, a portal also needs to protect the integrity of password usage by enforcing usage policies such as minimum password length and a password expiration date.

**Digital certificates are used to authenticate a Web server to the user's Web device**

When discussing authentication, the focus is usually on ensuring that a user is authorized to access the portal server. In the Web environment, however, security needs may also require the server handling the authentication service to authenticate itself to the Web device. Server authentication ensures that the user's Web device cannot be fooled into connecting to an unauthorized Web server that could violate security by, for example, downloading a rogue application to the Web device. This type of authentication is often done by using digital certificates and the services of a trusted third-party *certificate authority* that verifies the authenticity of the certificate.

**Operating systems employ a variety of different security technologies**

Portal software developers face the problem that each operating system employs different technologies for authentication and directory services. Unix systems, for example, employ NIS and Kerberos, while Windows NT uses NTLM. Microsoft's latest operating system, Windows 2000, includes a Microsoft feature known as Active Directory, which employs LDAP for directory services, and Kerberos for authentication.

**LDAP provides cross- platform directory services**

In the area of directory services, the industry is beginning to adopt LDAP as a cross-platform solution. LDAP was initially developed as a low-cost desktop interface to X.500 directories. When the ISO X.500 standard became too complex to be universally accepted, LDAP expanded its role add a directory service. Current LDAP implementations fall into three categories: *Those that locate network users and resources, those that manage them, and those that authenticate and secure them.*<sup>3</sup> In addition to being used by the Microsoft Active Directory, LDAP products can be found in Unix implementations developed by Compaq, Hewlett-Packard, Sun, and Silicon Graphics. It is important therefore that portal products provide an interface to an LDAP directory server.

---

<sup>3</sup> Tim Howes. *LDAP: Use as Directed*. Network Magazine, February 1, 1999.

**A portal's session manager provides authentication and session controls**

For authentication a portal can rely on the services of the underlying operating system, or it can provide its own session manager to authenticate each user request. One benefit of a session manager is that it can timeout a user session after a period of inactivity, and/or request the user to sign-on at periodic intervals.

**Web Device Security**

The two objectives for securing the user's Web device are to stop unauthorized network access to the device's resources, and to prevent software running on the device from violating security. The most common approach to preventing unauthorized network access to a Web device is to use a *firewall* to control network traffic to and from the device. We discuss firewalls in the section *Web Network Security*.

**Web browsers can create a variety of security problems**

Portal products use a variety of software techniques to manage the Web device and the user's interface to the portal. These range from simple HTML/XML screen formatting using the services of a Web browser, to sophisticated user interfaces that employ downloaded applications written using Java or Microsoft ActiveX programming languages. When using HTML/XML, the biggest security exposure is the Web browser itself. There are a variety of ways a browser can create security problems. The default authentication for Web pages, for example, is *basic authentication* where the Web browser uses HTTP to send the user ID and password to the server in clear text. Using HTTPS, instead of HTTP, can solve this problem, since all HTTPS communication is encrypted.

**Web browser cookies are often used to manage user sessions**

Another Web browser security issue is related to how the portal software manages session and state information. HTTP is a stateless protocol, i.e., each HTTP request is an independent event, and this makes it difficult for an application running on a Web server to manage user sessions that involve multiple related interactions. It is not practical to ask the user to supply a user ID and password with each request. There are a variety of methods used to overcome this problem, but commonly, a *cookie* is used.

**Cookies can expose passwords and session IDs**

A cookie is an object on the user's Web device that can be used to store information about the user or the user's session. It has an identifying string, an expiration date, and a URL pattern that indicates when the cookie should be sent with an HTTP request. When the user connects to a Web site, the browser checks to see if any unexpired cookies match the URL pattern, and if so, the browser sends the cookies along with the request. A portal's session manager can use a cookie to store a session ID so that it can identify the user when he or she sends a new request to the portal server. It is important, therefore, that cookies are protected to prevent unauthorized network applications from:

- Accessing cookies on the Web user's device – this can be achieved by placing the Web device behind a firewall (this of course only protects the Web device from external applications, not from internal ones).
- Accessing cookies during transmission – one approach would be to use HTTPS to encrypt the cookies.

- Fooling the Web browser into sending a cookie to an unauthorized server – this can be prevented by the portal Web server authenticating itself using a digital signature.

**Portals servers that download software to a Web device may need to be authenticated**

More sophisticated portal user interfaces can be created by downloading a Web browser plug-in, Java applet, or ActiveX control to the portal user's Web device. The main concern here is to prevent a rogue application being downloaded that can compromise security. Again, digital certificates can solve this problem, since they can be used to authenticate the server that wishes to download an object to the Web device. The use of Java also has the benefit that a Java application cannot access Web device resources (a file, for example) that are external to its application boundary, or *sand box*.

### **Web Network Security**

**HTTPS is used to encrypt network messages**

Portal servers that employ pure HTML or XML to interact with a Web device usually do so using HTTP over a TCP/IP network. Two common ways of protecting the TCP/IP network, and the messages flowing over it, are by using a secure networking protocol, and by the use of a corporate firewall. We have already discussed how HTTPS provides a more secure version of HTTP by encrypting message traffic, and we will, therefore, move on to review briefly the facilities provided by a firewall.<sup>4</sup>

A firewall is usually installed at the point where a secure corporate intranet is connected to the public Internet. All network traffic coming from, or going to, the Internet passes through the firewall and can, therefore, be checked for acceptability. Some installations add an additional network between the intranet and the Internet to provide an additional layer of security. This network is known as a *perimeter network* or *de-militarized zone (DMZ)*.

**Firewalls do not prevent internal attacks**

Whereas a firewall can protect against external attack, it cannot protect against malicious insiders. Internal threats must be prevented by the corporate intranet itself. Another issue with a firewall is that it can interfere with access to the Internet, which may be an annoyance to internal users.

**Firewalls offer many different security features**

It is not possible in this paper to discuss the many security facilities that are available in firewall products, and we will instead simply review some of the key ones below.

- *Packet filtering* is where the firewall applies a set of rules to each packet of data coming across the network, and either allows the packet to pass or blocks it from reaching its destination. One use of this technique is to control which network devices (usually network IP addresses) can communicate across the Internet.
- *Proxy services* involve an application or server that takes internal user requests for Internet services and forwards them to that service on behalf of the user, i.e.,

---

<sup>4</sup> A good reference for more detailed information about firewalls is *Building Internet Firewalls* by Elizabeth Zwicky, Simon Cooper, and D. Brent Chapman.. Published by O'Reilly & Associates, Inc., 2000. ISBN 1-56592-871-7.

the user's Web device is hidden (and thus protected) from devices on the public Internet.

- A *virtual private network (VPN)* employs encryption and other techniques to route packets of data flowing on a private internal network across the public Internet without this being visible to the internal applications and devices that are handling those packets. VPNs reduce the need for corporations to employ costly private networks involving leased lines.

**Portal servers may use protocols other than HTTP**

So far we have looked at protecting portals that employ HTTP to communicate between the portal server and Web device. Not all portals, however, use HTTP. Portals that employ programming languages such as Java for creating a more powerful portal user interface sometimes employ a low-level protocol like TCP/IP, or a high-level protocol such as CORBA/IIOP. Software developers employ these protocols because they typically provide more flexibility, and better session control, than HTTP. Note also that wireless devices such as cellular phones employ protocols like the Wireless Application Protocol (WAP) and Wireless Transport Layer Security (WTLS) to communicate with Web and portal servers.

**HTTP tunneling is used to route non-HTTP messages through a firewall**

The security used with protocols like TCP/IP and CORBA/IIOP will vary based on how the portal software implements the protocol. The Secure Sockets Layer (SSL) is sometimes used for encrypting data and handling authentication. (Note that HTTPS is an implementation of SSL on HTTP.) Another consideration here is that if a firewall is installed, the firewall may not be able to handle protocols other than HTTP. One solution to this problem is for the portal software to employ *HTTP tunneling*, which allows a non-HTTP protocol to pass through a firewall. There are several different ways of implementing HTTP tunneling – some are secure, and some are not. Each portal product will need to be evaluated to ascertain its security capabilities in this area.

### **Web Server Security**

**Web servers can be attacked from both within and outside an organization**

Web servers can vary from rudimentary system software that handles HTTP messages to complex servers with capabilities like load balancing and fail-over that allow developers to add sophisticated software applications such as a corporate portal. The more enhancements and additions a Web server permits, the higher the risk of a security violation. A Web server can be attacked from either within an organization or from outside. As we have already discussed, outside attacks can be prevented using firewalls and inside attacks must be prevented using the security facilities provided by the corporate intranet.

**Portal server software written in Java can improve security**

One of the main security exposures in a Web server can come from the software running on that server. These exposures can be caused by a malicious user tricking the server applications into doing something they shouldn't or by a malicious user running unauthorized external programs. The first problem is caused by software developers building insecure applications or by using insecure Web server features. The use of a secure programming language such as Java reduces the likelihood of a security violation occurring.

**Server application access to file and database systems must be controlled**

Preventing a malicious user from running unauthorized external programs requires the Web server's security administrator to control services like FTP that can upload programs onto the server. It is also important to control the file and database systems each Web application can access. Given the many different types of product on the market, Web server security has to be evaluated on product-by-product basis.

**Portal and Business Content Security**

**User IDs and passwords for accessing portals and business content must be coordinated**

Portal products and the applications that maintain the business content that can be accessed by a portal all implement their own security services. In general, this involves authenticating users based on a password, and authorizing users and user groups to access specific named resources maintained by the portal or application. The requirements here are to avoid having to define user IDs and passwords both to the portal and each application and to avoid asking users to sign on to each application that the portal accesses. The proliferation of passwords leads to secure exposures such as users reusing passwords, selecting obvious passwords or writing down user IDs and passwords. These issues can be avoided by the use of directory servers and a single sign-on architecture.

Other technology issues associated with portals have already been discussed. To summarize, these involve session management, the use of secure networking protocols, and the use of secure application software running on both the Web client and the Web server.

**Auditing and Logging**

**A portal's logging and auditing services should be integrated with the underlying system software**

We have already identified the reasons why a portal product should provide a logging facility to track security violations and record user sessions. Ideally such a logging service should be integrated into the logging facilities provided by the Web server software on which it runs, or should use the logging capabilities of a central administration service, if one exists in the installation.

**PUTTING IT ALL TOGETHER**

**Security is a key element in the selection and implementation of a portal product**

Security is a key element in the selection and implementation of a portal product. Organizations need to identify the resources (business content, portal services, etc.) that need to be protected in a portal system, and the level of protection required. They also need to determine which users require access to those resources, and the type of access required. The results of this analysis are used to create a security requirements specification that is provided to the team selecting and implementing the portal software.

**A portal must be able to integrate seamlessly into an organization's security framework**

Portal security must be coordinated with, and integrated into the system components that provide the portal operating environment. Choosing the right portal product requires selecting a product that not only meets the security requirements demanded by the portal project, but one that can also be seamlessly integrated into an organization's overall security framework.

# PRODUCT EXAMPLE: VIADOR E-PORTAL

Viador Inc., headquartered in Mountain View, California, is a software company that develops and markets a technology platform for building Internet and intranet portal solutions. Its *E-Portal* software provides a scalable and open architecture for presenting a single, personalized view of business content to portal users. In this section we first provide a brief overview of Viador E-Portal architecture and operation, and then discuss in detail how the product supports the security requirements outlined in the first part of this paper.

## E-PORTAL ARCHITECTURE AND OPERATIONS

**Viador E-Portal runs on Windows and UNIX systems and provides both HTML and Java interfaces**

Portal users employ the HTML-based *E-Portal Express* Web browser interface to access a Viador E-Portal server operating under Windows NT/2000 or UNIX. Additional functionality, such as Web-based business intelligence and user administration, is provided by Java applets running under the control of the Web browser. The appearance and contents of the E-Portal user interface can be customized to satisfy user needs and corporate standards. An example of a Viador E-Portal user interface is illustrated in Figure 3.

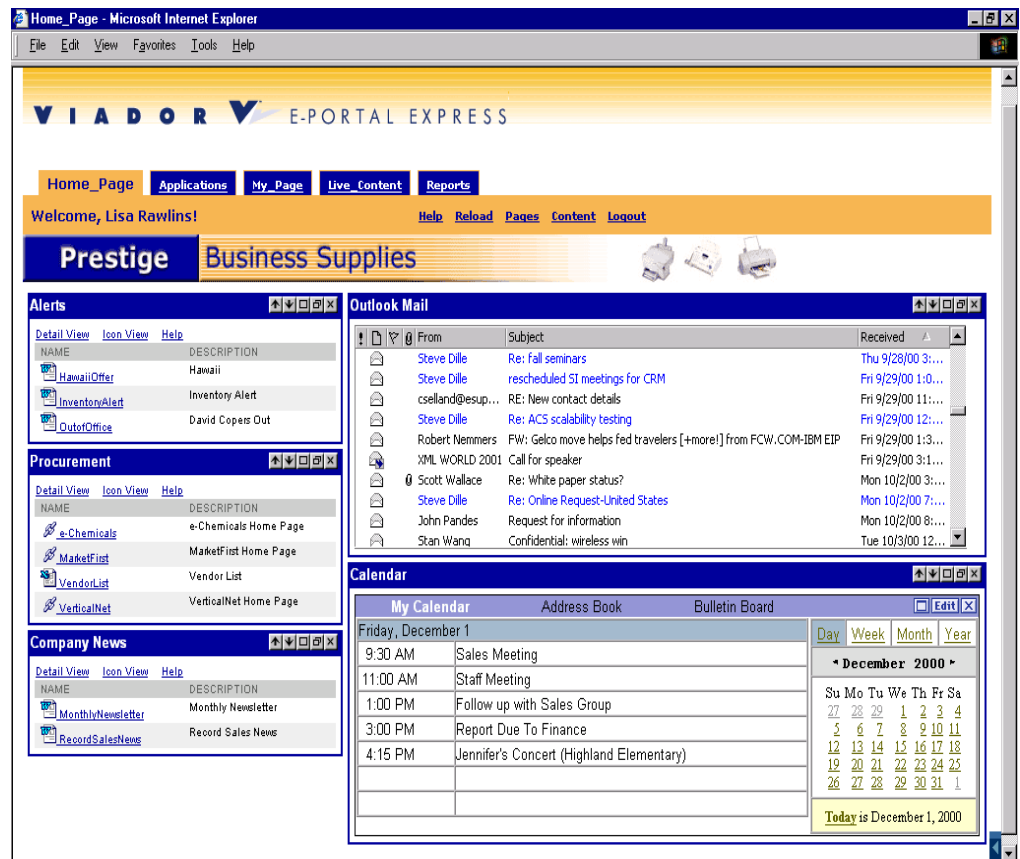


Figure 3. E-Portal user interface example

**Business content is documented in the repository in private and shared folders**

Business content viewed through a Viador portal is documented in a set of hierarchically arranged folders that are stored as metadata in the E-Portal *repository* (called an information directory in the first part of this paper). These folders are either private folders or shared *channel* folders. A private folder can be seen only by the business user who creates the folder, whereas a channel folder can be viewed by anyone. Content that is restricted to a specific user (or user group) is published directly to the user's personal folders. Content that is shared is published to a channel. A business user can select the channel folders that are to be displayed in his/her own portal interface. A search facility is provided for searching both E-Portal folders and business content.

**Business content is accessed using Viador and third-party Java portlets**

Access to back-end business source content is done using a program known as a *portlet*. These programs may be supplied by Viador, by a third-party software vendor or by the customer. Portlets are coded in languages such as Java and JavaScript and can utilize the services (data access services, repository services, etc.) of the E-Portal environment using documented APIs. The Viador E-Portal comes with portlets for a range of different business sources including relational and multidimensional DBMSs, and document and office management systems. The product also has built-in reporting and OLAP analysis capabilities.

## **E-PORTAL SECURITY FEATURES**

**Viador Sentinel extends the security features of E-Portal**

Having gained an overview of the E-Portal architecture, we are now in a position to discuss its security features. A summary of these features can be found in Figure 4. This figure, and the text that follows, document the security features of Viador E-Portal and also an optional security component of E-Portal known as *Viador Sentinel*. This latter component provides additional features for building a highly secure portal operating environment and is especially useful for e-business and ASP operations across the public Internet and corporate extranets.

### **Web User Device Security**

**An LDAP server can be used to authenticate users**

Portal users require an E-Portal account before they can employ the services of the portal. These accounts (user ID, initial password, and authorizations) are set up by the E-Portal administrator and stored in the E-Portal repository. To simplify administration, user IDs can be replicated into the repository from an LDAP directory server. When the user signs into E-Portal using his/her password, either E-Portal, or an external LDAP directory server, can authenticate the user's password. The user's account is suspended after a set number of failed sign-on attempts.

**User sessions that remain idle are terminated**

If user sign on and authentication is successful, a new portal session is initiated. Session information for validating each user interaction with the portal is passed to the user's Web device in an encrypted form in the HTML page rather than in a Web browser cookie. Viador Sentinel also encrypts the transmission of the portal pages. A session is terminated when the user exits the portal or the session remains idle for an administrator-defined period of time.

Security Requirement	Viador E-Portal Security Feature
<p><b>Web User Device Security</b></p> <ol style="list-style-type: none"> <li>1. Sign-on and authentication of portal users.</li> <li>2. Secure directory server to administer portal user IDs and passwords.</li> <li>3. Session manager to manage portal user sessions.</li> <li>4. Sensitive data (password, URLs, etc.) is scrambled on portal user's display.</li> <li>5. Secure use of Web device software.</li> </ol>	<p>The portal user must sign on and enter a password to create a new portal session. The user account is suspended after a set number of failed sign-on attempts.</p> <p>User IDs can be replicated from an external LDAP server into the E-Portal repository. The LDAP server can also be used to authenticate the portal user's password.</p> <p>Sessions will time-out after a specific period of inactivity. Session data is stored in an encrypted HTML page, rather than a Web browser cookie.</p> <p>User passwords are scrambled. Viador Sentinel encrypts the transmission between the Viador server and user web browser.</p> <p>The user interface employs Java applets. A digital certificate can be used to authenticate the portal server downloading the applets.</p>
<p><b>Portal Network Security</b></p> <ol style="list-style-type: none"> <li>6. Secure message traffic.</li> </ol>	<p>HTTP is used for HTML Pages, and TCP/IP for Java communication. Viador Sentinel uses SSL to encrypt all message traffic, and HTTP tunneling to allow TCP/IP traffic to flow through a firewall. Packet filtering can be used to restrict access to specific IP addresses.</p>
<p><b>Web Server Security</b></p> <ol style="list-style-type: none"> <li>7. Secure use of Web server software.</li> </ol>	<p>The portal software on the server is written in Java. Viador Sentinel provides a DMZ server for use with a corporate firewall.</p>
<p><b>Portal Server Security</b></p> <ol style="list-style-type: none"> <li>8. Portal server security is integrated with the system software.</li> <li>9. Access to the information directory (repository) metadata and portal services is controlled at a user or user group level.</li> <li>10. Logging of security violations and session data for auditing purposes.</li> </ol>	<p>E-portal security can be coordinated with an LDAP server or an external security system, e.g., Netegrity SiteMinder.</p> <p>Business content is published to user/user group-controlled private folders, or shared public folders. The administrator can control who can publish business content and create new public folders. The administrator can control who can view, run, and design reports using the built-in reporting facility. Report filtering can be done using the ID of the portal user.</p> <p>User session data (user ID, session start/end time, Web pages visited, etc.) is recorded in a database log.</p>
<p><b>Business Content Security</b></p> <ol style="list-style-type: none"> <li>11. Security used to access business content should be transparent to the portal user.</li> <li>12. Portal business content authentication scheme should support the customer's infrastructure for managing security.</li> </ol>	<p>Single sign-on capabilities mean that users do not need to enter IDs or passwords to access business content. Administrators instead control the business content that can be accessed by a user. The portal employs IDs and passwords recorded in the repository to access business content.</p> <p>Custom adapters added to the product can employ E-Portal security services, or an external security service. Supplied adapters do not require external security services.</p>

Figure 4. Viador E-Portal security features

**Business content URLs are encrypted by Viador Sentinel**

Portal users employ industry standard Web browsers to interoperate with the E-Portal server (known as the *Viador Information Center*). Users have the choice of an HTML or *Java applet*-driven browser interface. Both interfaces hide user passwords and encrypt URLs when using Viador Sentinel. The Java interface provides additional functionality and offers a more secure operating environment due to Java's *sandbox* security architecture. A digital certificate can be used to authenticate the identity of the portal server downloading the Java applets.

**Portal Network Security**

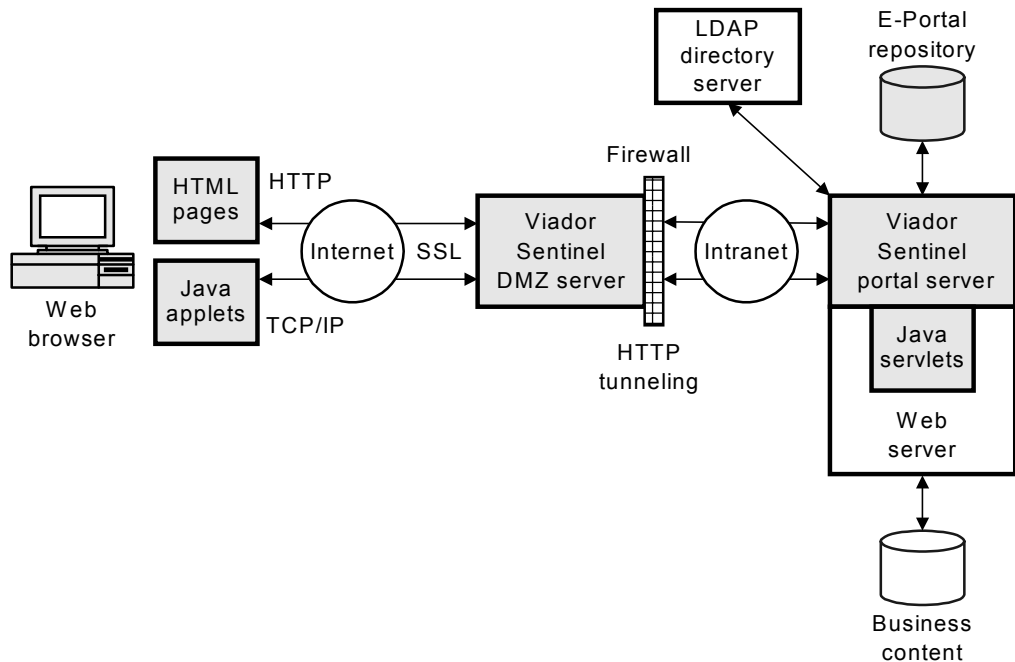
**Viador Sentinel uses SSL to encrypt network messages**

The E-Portal server employs HTTP for sending HTML pages to the user's Web browser, and TCP/IP for communications with the Java applets running on the user's Web device. All HTTP and TCP/IP network messages are encrypted by SSL when using Viador Sentinel. To allow TCP/IP messages to flow through a firewall, Viador Sentinel employs HTTP tunneling to provide secure communication. Packet filtering can be used in the firewall to restrict E-Portal server access to specific network IP addresses.

**Web Server Security**

**Viador Sentinel provides a DMZ server**

The E-Portal server supports leading industry Web servers running in a variety of operating environments, including Windows NT/2000 and UNIX. Figure 5 illustrates a typical server configuration when using Viador Sentinel in conjunction with an organization's corporate firewall. Note that there is often another firewall between the Internet and Viador Sentinel DMZ server.



**Figure 5. Viador Sentinel Web server architecture**

**E-Portal server software is coded in Java**

The E-Portal server makes extensive use of Java. All server facilities that interoperate across the network with the portal user's Web browser are implemented as *Java servlets*. The use of Java provides a secure environment that protects the E-

Portal server from corruption by other programs and users running on the same Web server.

### **Portal Server Security and Auditing**

**Authorization can be managed at a system, user group, or user level**

Users must be registered in the Viador E-Portal repository before they can use the services of the portal. The Viador E-Portal offers a variety of different user authorization schemes depending on the level of security required. Where security is not a key issue, a single user ID can be defined that can access all the services of the portal and all the business content that can be viewed via the portal. At the other extreme, for a very secure environment, each user is given his/her own user ID and authorization to access portal services and business content is managed at an individual user ID level. A middle of the road approach is to give each user an individual ID, but to assign each user to one or more user groups and then manage authorization at a user group level.

**E-Portal can be used with Netegrity SiteMinder**

E-Portal user administration can be integrated with external directory services. We have already discussed how an LDAP directory server can be used to create user IDs in the repository and to authenticate user passwords. Other third-party security services can be supported by coding a custom authentication module. Viador also has a relationship with Netegrity to integrate E-Portal user accounts and authentication checking into Netegrity's SiteMinder product. SiteMinder provides facilities such as single sign-on and centralized authentication management.

**Administrators can control who can publish content and create shared folders**

Business content that can be accessed via the Viador E-Portal is documented in a set of hierarchical folders in the E-Portal repository (see Figure 3). As discussed earlier, these folders are either private folders or shared channel folders. The portal administrator can control which users or groups can see each channel, which groups can publish business content to each channel, and which users can define new channels. Each user can personalize the content displayed by the E-Portal Web browser interface and can search channel content using a supplied search engine.

**E-Portal integrated reporting facility has record-level filtering**

In addition to providing access to back-end business content, the Viador E-Portal also has a business intelligence reporting and analysis facility. The portal administrator can control which users can view, run and design reports and analyses using this facility. Report and analysis results can be filtered based on the user sign-on ID. Authority for a user to access back-end data stores using the E-Portal reporting and analysis facility is managed in the same way as access to any other type of business content – see *Business Content Security* below for more details.

**User session data is written to an audit log**

The E-Portal server can optionally log session information so that administrators and auditors can monitor and analyze portal operations. Information recorded in the log includes the ID of users accessing the portal, session start and end times, Web pages visited by each user and so forth.

### **Business Content Security**

**Single sign-on is supported for accessing business content**

To avoid the need for portal users to have to sign on to each business content store the portal administrator can implement a single sign-on system that records the authorization ID and password for accessing each content store in the E-Portal repository. The administrator then uses the E-Portal security mechanism to authorize which users (and user groups) can access any given store. When a user attempts to

view business content (by clicking on a channel entry, for example) the portal uses the authorization ID and password stored in the repository for that user or user group to gain access to the content.

**Portlets can use the E-Portal security services API**

If the E-Portal software does not support a specific type of business content, then developers can add this support by coding and installing *portlets* in the portal server. Documented programmatic interfaces provide these programs with full access to Viador E-Portal facilities such as the repository and security services, allowing them to be fully integrated into the E-Portal environment.

## **SUMMARY**

A portal must ensure the integrity and security of corporate information assets if it is to be successfully employed by business users. This is especially true when deploying a portal for use across the public Internet, or a portal that will be accessed by external trading partners and clients.

**Viador Sentinel is well suited for corporate extranets**

Viador markets two versions of its E-Portal product. The base version provides a secure portal operating environment that is ideal for use on a corporate intranet. This product provides single sign-on and personalized access to a variety of back-end information stores and applications. Viador Sentinel extends the base product with several security enhancements that makes it particularly well suited for secure access by portal users who are accessing the portal from outside the corporate firewall.

---

### **About DataBase Associates**

Database Associates is an international consulting company specializing in leading-edge technologies in the fields of data warehousing, business intelligence, analytic applications, corporate portals, and database.



#### **DataBase Associates**

Post Office Box 398

Ashland, OR 97520

Telephone: (541)-552-9126

Internet URL: [www.databaseassociates.com](http://www.databaseassociates.com)

E-mail: [info@databaseassociates.com](mailto:info@databaseassociates.com)

*Safeguarding the Corporate Portal: A Review of Portal Security*

*Version 1, January 2001*

Copyright © 2001 by DataBase Associates

All rights reserved.