Whale Communications
A Microsoft Subsidiary

Application Intelligence: What Is It and Why Do You Need It?

Published: February 2007

For the latest information, please see:

http://www.microsoft.com/iag

Contents

Introduction

A growing number of organizations have come to recognize the value of Secure Socket Layer (SSL) VPN technology in terms of providing secure, browser-based access to business-critical enterprise applications. SSL VPNs deliver tangible value to organizations by broadening the availability of resources and enforcing parameters for access without the need for dedicated client software. As organizations increasingly look to Web-based access as a tool to improve employee productivity and business agility and responsiveness, the technology must deliver not only a mechanism to connect the user to the network, but also determine how the user can connect and to which resources. The organization requires a solution that serves the need for connectivity and network integrity and at the same time provides a tool to address the business issue of how an organization effectively uses information.

The next phase, therefore, in enterprise adoption is based on leveraging the pervasiveness of the browser to implement enhanced business processes, and engineer a balance between productivity and security. This requires a far more comprehensive and flexible approach to access, security and policy enforcement that encompasses multiple access modes. However, the impetus for an organization to look beyond the more immediate benefits of Web-based remote access is significant. Web-based access not only expands the availability of critical applications to internal employees and simplifies remote access from branch offices, but also allows customers to gain access to real-time data and enables partners to be tightly integrated into business processes – all from the same infrastructure and policy logic repositories. Many forward-looking enterprises have looked at SSL VPNs as a means of enforcing access to applications regardless of whether the user is on the LAN or the WAN.

Longstanding SSL VPN vendors and even newer entrants to the market can deliver some level of proxy functionality that enables secure publishing of intranet applications to the public Internet, including URL cloaking to avoid exposure of internal topologies as well as dynamic URL rewrites. This generic rewriting approach has its limitations, in large part because establishing connectivity to a broad a set of applications forms the basis of the product development philosophy. In addition, many applications use hard-coded paths that cannot be rewritten or rely on cookies that cannot be passed through a gateway.

In order to enable functionality for complex applications, a generic approach requires a component download – a methodology that raises security and endpoint compatibility issues. Tunneling an application through a client-side Java or ActiveX® component reduces the level of control over presentation the gateway can apply, affects the user experience by removing the possibility of implementing Web-based Single Sign-On, and generates support costs when users cannot download the component. With each move to enable broader application availability, organizations must decide how to appropriately secure access in a manner that is not overly intrusive, remains secure, and preserves functionality and a positive user experience. Achieving secure application availability requires a platform that can enforce fine-grained control over application access based on user location and align access policies with business needs. Consistently publishing Web applications to the Internet to enable fully functional and policy-driven access requires specific knowledge of how an application functions and how actions are implemented.

Microsoft refers to this ability to understand how an application functions and the semantic dialogue between the browser, client software and server as Application Intelligence. This capability is in turn based on the Intelligent Application Gateway (IAG) 2007's architecture, designed from the outset to classify, secure and manage application-layer flows between the client browser and downstream servers. Microsoft has leveraged this underlying ability to understand the context of application communications to create application-specific modules that encapsulate the specific rewriting rules for applications, as well as the preconfigured

settings for endpoint security, positive logic policy enforcement, and access policy management.

In the absence of Application Intelligence, SSL VPNs cannot evolve to deliver the capabilities required to address strategic business concerns:

- Improved productivity
- Deeper business-to-business collaboration
- Enhanced customer relationships.

The IAG 2007's complete control of outgoing and incoming content enables organizations to ensure the optimal end-user experience while enforcing comprehensive security, including endpoint access control, session security, and application firewalling and filtering.

What is Application Intelligence?

Application Intelligence refers to the underlying technology of the IAG 2007. This technology supports a related set of features that can either be utilized for generic Web applications or tailored to specific applications with more complex requirements. The ability to understand how the components of an application interoperate and how they are published through a browser cannot be confined to a single feature. Rather, it should be integral to the overall product, from enabling connectivity to defining intra-application access based on user identity and endpoint profile. Microsoft's Application Intelligence is the foundation for preconfigured software modules that incorporate default settings and values for specific applications. These Intelligent Application Optimizers, encapsulating predefined logic, are the result of in-depth research into application behavior, browser-server interaction and use of client components.

The Optimizers serve to conceal the complexities of configuration and policy settings to the administrator, but provide an intuitive wizard-driven policy management interface for customizing access to match particular enterprise needs.

Application Intelligence is fundamental to achieving the business aims of secure remote access and balancing the competing imperatives of increased productivity, data integrity and information security. Generic SSL VPN approaches allow for 'tiers' of access from a browser based on location, advance over IPsec VPNs in the sense that no client is required, and different access locations are not treated uniformly. Microsoft, however, looks at the problem across several dimensions – restricting access serves to limit the security risks posed by an unmanaged endpoint, but can also diminish the value to the organization of Web-based access.

In addition, defining access rights by location and user group works well for remote employee use cases, but provides less flexibility in extending access to a broader range of user categories such as business partners, vendors or customers, or in enforcing permissions for internal employee groups.

IAG 2007's approach is based on the most robust security of any SSL VPN platform in the marketplace coupled with Application Intelligence. By securing the entire session lifecycle, the IAG 2007 can ensure that increased access does not come at the cost of escalated risk. Application Intelligence further entrenches the value proposition by offering the ability to define access rights within applications, such as download or upload actions. This approach results in maximum functionality for the largest possible number of external users.

The IAG 2007 provides physical network separation, data encryption and user authentication (including strong authentication) to address the need to shield internal servers from directly interacting with unauthenticated users and exposure to the Internet. The gateway further provides built-in baseline policy checks for host characteristics to ensure that only healthy clients access the network and minimize the possibility that Web access can disrupt overall operations by introducing a worm or a virus. The gateway and Optimizer(s) then ensure that the next leg of the data transfer does not compromise corporate IT infrastructure by enforcing positive logic policy that ensures only legitimate server queries are allowed to pass through the gateway. In the course of the user session, the gateway cloaks host names and IP addresses through URL rewrites so that the application domain's IP addressing and topology are not inadvertently exposed. Once the user terminates the session, the Attachment Wiper™ will shred all user information and data stored in cache.

Application-Defined Access Control

The initial step in delivering application access is verifying the user and establishing which network resources they are authorized to use. While implementing these procedures effectively is fundamental to remote access security, there are distinct advantages to be gained from elaborating on generic approaches. In the IPSec VPN world, authentication and authorization support network-layer access rights – once the user is authenticated, there is a predefined set of IP addresses which are authorized for access. In the realm of browser-based application access, this model falls short both in meeting security requirements and in providing the flexibility to align technology with business needs.

The concept of network access control was pioneered in the SSL VPN space since, in the absence of an IPSec client, the SSL VPN gateway must establish whether the client is healthy at the point of connection. Whale Communications introduced the notion of endpoint security to the marketplace, based on the realization that an authenticated user accessing authorized resources over an SSL tunnel may still pose a threat to network integrity due to undetected malware, viruses or Trojans lurking on their client.

The ability to define a baseline of endpoint compliance for connectivity in conjunction with flexible user authentication and authorization enables organizations to extend access to a broader range of access scenarios without creating vulnerabilities. An additional layer of policy is required to serve multiple user groups, possibly with varying degrees of application authorities. Where applications have a greater degree of sensitivity and access controls must be stringently defined, organizations must have the ability to impose custom authentication schemas as well as custom endpoint integrity checks. In instances where corporate data is subject to a greater degree of control, organizations can establish parameters of acceptable application actions as well as acceptable access scenarios to allow for limited application usage, in this way satisfying security or policy restrictions while still enabling resource availability.

Endpoint integrity checks

Connecting an endpoint that is external to the corporate LAN poses a threat to the corporate network. Malicious software resident on that endpoint can disrupt network integrity or result in undetected data harvesting. Such malware can exploit the SSL VPN tunnel to try and infect or attack the corporate network – using what is sometimes referred to as a secure ‘sewage tunnel’.

A further area of sensitivity is the residue of application data on the endpoint once the session is terminated. While the network session is encrypted, data leakage may still occur if privileged or sensitive information remains in the browser cache or is saved in association with other applications on the endpoint. To secure the session, the application gateway must employ an endpoint detection component to perform a set of host checks, verifying the client’s health and the presence of up-to-date security software. The results of the endpoint integrity check should be used to set access privileges, either extending full functionality access or restricting certain application actions that may introduce malware or viruses into the corporate network.

One example restriction may be to block uploads. Where a cache cleaner cannot be downloaded, organizations should have a mechanism to deny access entirely.

The endpoint detection agent should support the configuration of a baseline check for defining access rights, as well as be able to support supplementary requirements for specific applications (such as Service Pack 2 for Microsoft SharePoint® Portal Server 2003). Baseline policy checks can include a broad range of components, including the presence of anti-virus and personal firewall software, registry keys, active services, and system elements. Being able to add custom checks on the endpoint and integrating the results seamlessly into the

policy mechanism ensures that the administrator has control over the compliance checking mechanism and can test any conceivable condition on the endpoint prior to providing access.

The IAG 2007 endpoint security features include session residue and non-persistent cookies cleaning as well as session security including encryption and host address translation that can be implemented on an application-specific basis. IAG 2007 endpoint checking engine can also support access controls based on checks for security tools, patches, security hot fixes and service pack information

Another aspect the endpoint detection agent is the issue of controlled vs. uncontrolled endpoint. Depending on the specific corporate requirements, an endpoint will be considered controlled or trusted if it has a client certificate, belongs to a certain domain, has a specific user logged onto it, is running specific software or containing specific registry keys, has specific hardware, etc. Several trust levels can be assigned to the endpoint, based on the elements found.

The endpoint detection component assures the gateway that the Attachment Wiper has downloaded and launched successfully. In addition, although not malicious, desktop search engines might save their own internal cache and thus disclose privileged information after the user has left the endpoint. For this reason IAG 2007 endpoint detection component detects the presence of desktop search engines and mitigates the danger. It also reports on service packs and security hot fixes, etc. The endpoint detection component is scriptable, making it the most flexible solution of its kind. It can be configured to interrogate any registry key, file, process, security attribute, software, and much more using a standard and familiar scripting language.

Attachment Wiper cache cleaner

IAG 2007's session cleanup agent is called the Attachment Wiper. It downloads and executes as a process separate from the browser, cleaning up regardless of how the session was brought to an end – including logoff, timeouts, browser being closed, browser crashes, orderly shutdowns and power failures. Files are wiped in conformance with US Department of Defense standard 5220.22-M.

As part of IAG 2007's Application Intelligence architecture, each application's protective shell contains information regarding the location of its cache, which may depend on the endpoint operating system. As with all other components, the Attachment Wiper is an extensible framework allowing for the addition of new applications and new cache locations.

Custom cache cleaning

The cache cleaner is required to purge all user session data and downloaded files and thus safely extend access to unmanaged endpoints. Sensitive information that may accumulate on the client during the session includes downloaded files, pages, images, user credentials, cookies, auto-complete forms, auto-complete URLs, and URL history.

The cache cleaner should be able to act transparently as a cleanup agent downloaded to the endpoint browser and must have an independent implementation process. This allows for initiation of session cleanup not only when the user logs out or closes the browser, but also when the browser crashes, or when the system is shut down. In addition, the cleanup agent must be prepared for a sudden power failure so that sensitive files left on the disk are deleted as soon as the system boots.

Simply deleting the files is provably insecure. The contents of the files can be recovered easily using forensic tools available on the Web. To ensure that the data has been erased in its entirety, it must be either electronically "shredded" or written over using a pattern. File

names should also be thoroughly deleted. File shredding is mandatory according to the latest ICSA Labs SSL VPN criteria.

Deleting browser caches is a serious consideration, but frequently applications will make use of multiple caches or client-side files that will need to be purged to ensure total session security. Web applications such as Domino Web Access, SharePoint Portal Server 2003 and EMC Documentum Webtop cache files in proprietary locations which are not normally deleted with generic browser cache cleaners. Cleaning the custom caches of specific applications requires preconfigured logic that can identify the caches or files when the session is launched and specific controls that implement data purges on termination of the session.

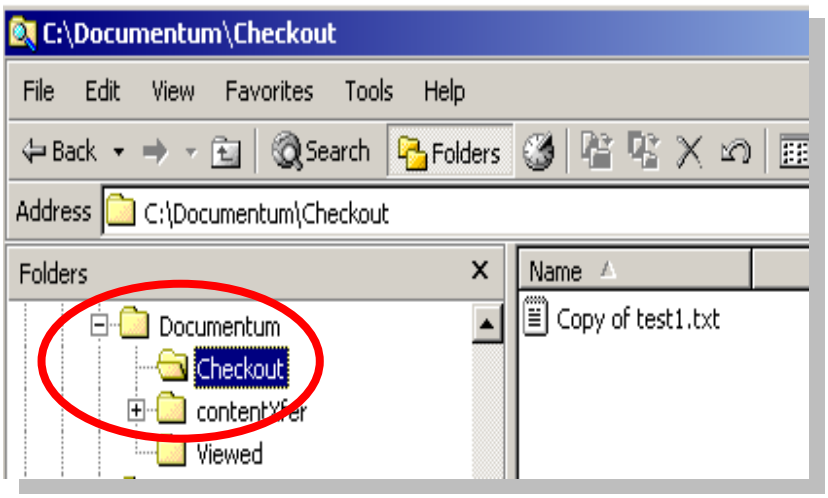

Figure 1: EMC Documentum Webtop custom cache

Application-specific action policies

Based on authentication, authorization and endpoint compliance, an IAG 2007 administrator can create comprehensive policies for portal login, application access rights, and acceptable application functionality usage parameters. Access policies must be informed by the results of the endpoint check and where no results exist, the policy must incorporate the flexibility to configure access rights based on their absence.

The ability to define and configure policies based on multiple variables and application context enables an elegant balance between enforcing security and encouraging improved productivity. Examples of the benefits:

In the event that the endpoint detection component could not be installed on the endpoint, the policy can dictate that the user will not be presented with a login screen.

The administrator can set a policy for drive mapping where unless an approved anti-virus is running and updated, the user will not have access to shared folders via mapped drives.

If the endpoint is determined to be a controlled laptop, the policy can dictate that timeouts are longer and that access to the corporate sales application is permitted.

Tighter access policies to application functions can be applied through policy settings. Application functions include document download, document upload, document check out, document check in, edit document, delete document, edit properties, etc.

These granular policies are only possible when the gateway has visibility into the application stream and incorporates preconfigured logic for specific treatment of application functions.

Authentication and authorization

In order to enforce varying degrees of access control based on user identity, endpoint profile and application sensitivity, enterprises should define strong authentication schemas based on existing directory policy logic or even build custom authentication schemas to deal with diverse user groups. The IAG 2007 can pair access permissions to applications, functions and other resources with users, groups and endpoint characteristics. Information about users and groups is stored in corporate directories or in directories that are built into the gateway. The gateway uses a detection software component at the client that reports to the gateway the endpoint parameters and security level. Access privileges are set according to the access policy that matches the user, applications and endpoint.

The IAG 2007 supports custom authentication schemes, enabling the administrator to stack authentication methods, tightening access security while allowing the users to enter all credentials on the same page. It can also combine authentication against one authentication server (such as RSA SecurID®) and authorization data from another server (such as Microsoft® Active Directory®). Besides standard authentication methods, the Intelligent Application Gateway supports custom authentication via simple, updateable scripts. IAG 2007 also supports strong authentication in the form of PKI, i.e. X.509 client certificates, as well as two factor authentication schemas such as RSA SecurID, VASCO Digipass® and Swivel PINsafe™.

Support of various directories in the IAG 2007 includes the ability to support different schemas (in the case of LDAP), attributes (in the case of RADIUS groups), and vendor specific concepts such as Local, Global and Universal groups and the Global Catalog in Active Directory. When a user enters multiple credentials either at login or at some other time during the session, all credentials are combined to determine the user's access rights to a specific application or resource. IAG 2007 strongly integrates with specific and generic LDAP servers, Active Directory and RADIUS groups.

Application Intelligence Controls

The embedded logic developed by the IAG Application Group to handle transactions based on specific insights into application behavior underpins capabilities that are typologically distinct from generic remote access. Rather than simply leverage the Web browser's embedded SSL encryption, separate the corporate LAN from the Internet and act as a point of authentication and authorization, Intelligent Application Optimizers deliver control over the requests transferred to the back-end server and the presentation of applications to end-users based on access scenario and identity. This additional layer of controls intervenes in the presentation of the application to the user, and enables the definition of policies to govern the user's interactions with the server. The application intelligent controls extend to application actions such as upload and download based on specific insights into URL categories. The ability to define the parameters of acceptable user actions within the context of the permissions and rights enables enterprises to broaden access while maintaining existing security policies.

Application Behavior Modification

Application behavior modification refers to the ability of IAG solution to embed features within an application that was not originally designed to be accessed over the public Internet. Many applications do not integrate sufficient internal application security mechanisms to support external access from non-trusted networks without creating vulnerabilities and security risks.

In these instances, it is necessary to modify the application's presentation in order to enhance security and add security-related functions. For instance, some Web applications lack a logoff function that disconnects the HTTP session and terminates the application. Instead, the application relies on external mechanisms such as user-initiated session termination by closing out the browser or administrator-defined session timeout based on user inactivity. In this case, the application intelligent control should enable the inclusion of a logoff button to allow a secure termination of the application.

Further, the application behavior modification feature should enable intervention in the outgoing HTTP stream in order to modify the application's presentation in accordance with corporate policies. When endpoint policy dictates that a user be restricted from performing specific application functions such as email forwarding, the application behavior modification should guide the user by incorporating mechanisms to disable or hide function buttons or icons, or alternatively include pop-up messages or insert other appropriate messages into the page when the user attempts to perform an operation that does not conform to policy. This will reduce the possibility that users flood help desks with support issues when functionality is blocked.

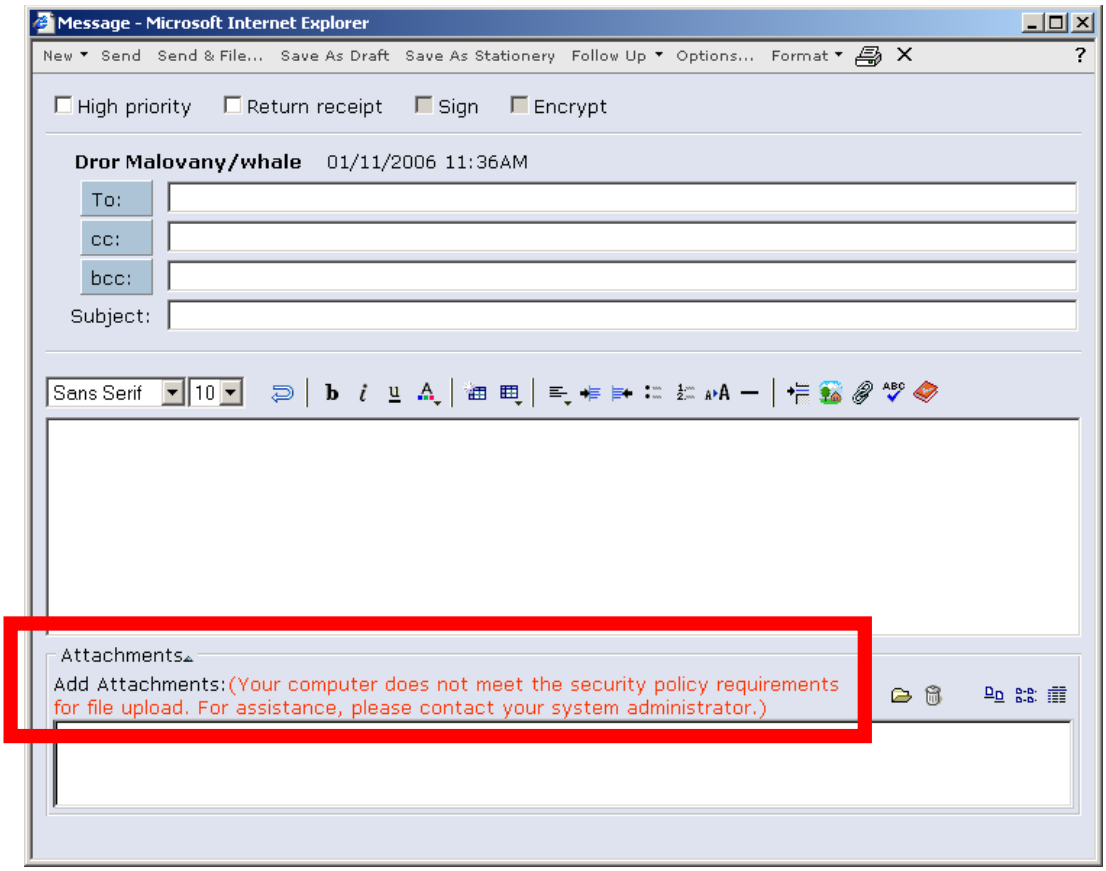

Figure 2: Example of application modification

Intra-application policy enforcement

Organizations that allow a broad range of users with varying authorization levels to access corporate applications from several categories of endpoint (corporate laptop, home PC or unmanaged endpoints) require tools that enable granular enforcement of policies. Users should be subject to access rules based on their role within an organization – such as groups for employees, or partner or customer groups for non-corporate users – and their access scenario. These conditions should determine their degree of access to applications and functionality within applications in an effort to align their access with corporate policies and the need for data security. Intra-application policy enforcement protects the organization from data leakage by preventing the extraction of sensitive information to non-trusted endpoints.

An example is blocking file downloads to an endpoint that does not have a cache cleaner to wipe session data from the client files. Another example is denying access to administrative functions for users who do not have the appropriate security level or are connecting from non-trusted endpoints which may harbor malware or viruses. The application intelligence control should identify the sensitive parts of the application through preconfigured logic and enforce the organization security policy by allowing or denying the user's access to those application functions.

Application Firewall and filtering

In a normal, risk-free environment, administrators can assume that in the context of Web-based connectivity the user's browser is only sending legitimate HTTP queries to the Web server, but there are a number of scenarios that mandate the implementation of application

filtering. A public browser or other non-corporate machine might be contaminated with a worm "sitting and waiting" for someone to authenticate in order to launch an attack. In addition, legitimate user credentials can be hijacked by a hacker looking to gain unauthorized access to corporate data. Another risk is that a rogue user potentially exploiting a semi-trusted connection as a partner or even a non-trusted connection as a customer can assume control over the application access gateway before or after authentication, essentially creating an open door to the internal network. Application-layer attacks may result in level of impact from a small disturbance in network availability to information theft and unauthorized control of back-end application servers. The Application Firewall's role is to protect the Web application servers and the application access gateway from these exploits and malicious attacks while allowing legitimate requests to pass through to the server, enabling the business benefits of browser-based application access.

The Application Firewall is located between the browser and the application server, intercepts each request, and inspects it before it reaches its destination. It analyzes whether it is a legitimate request or a hacker trying to penetrate internal systems and/or retrieve information. In order to identify an application "legitimate request" the Application Firewall must include knowledge of the application's commands.

Application-level control includes thoroughly inspecting URLs, methods, and parameters, and any other incoming data. The inspection rules can be based on the positive logic of the application, indicating a controlled set of legitimate URLs, method, and parameter combinations to which the requests are expected to conform. This prevents application-level attacks based on malformed URLs or HTTP requests.

Single Sign-On

The convenience of Web Single Sign-On (SSO) is an essential component of user satisfaction, and consequently productivity, for external access to applications. Users should be provided with the option to enter their credentials once only when logging into the gateway, avoiding multiple requests for credentials. For instance, user adoption of external access to Microsoft SharePoint Portal Server and SharePoint Services can be dramatically and negatively impacted if users are constantly prompted for credentials for each application action. When an application is launched, the gateway should provide the application these user credentials. Web applications can request two types of authentication:

- HTTP 401 or basic authentication – this is easy to implement and does not require application knowledge.
- Web forms (static or dynamic) – this requires application knowledge of the specific Web page and input fields.

Web-based SSO is supported with HTTP 401 basic authentication and web forms – both static and dynamic. Web forms SSO is an application intelligent component since the gateway must parse the form, locate the relevant field and return an answer to the application without displaying the authentication page to the user. Dynamic Web forms are more complex than static forms since the application may use several types of forms with varying fields. For instance Citrix NFuse may ask for username and password or may use a Guest account that does not require username and password. Citrix may ask for the domain or not, as shown in Figure 3 below.

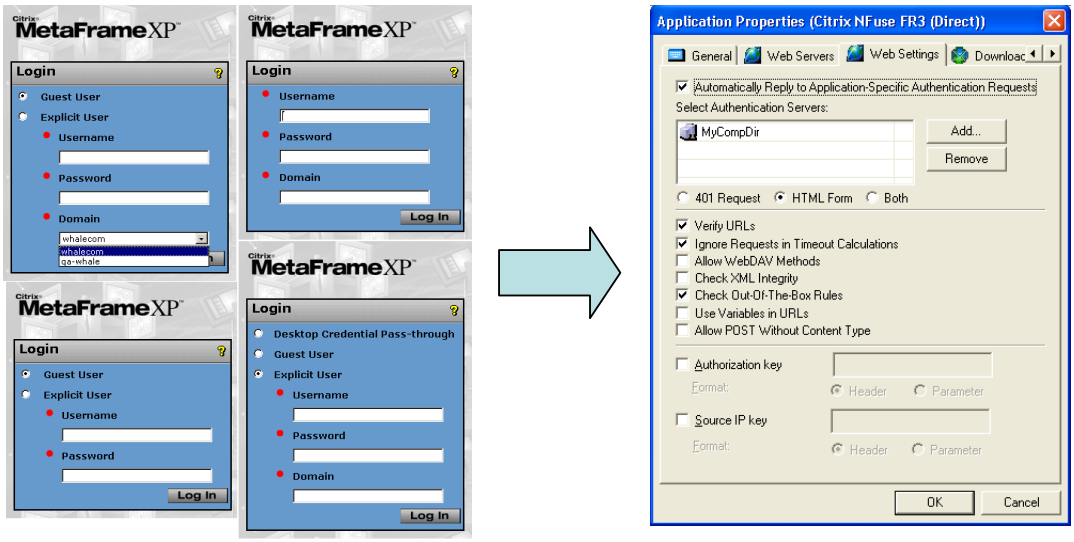

Figure 3: Citrix NFuse dynamic Web forms Single Sign-On

Native application launcher

After establishing an SSL VPN session, an application can be launched either automatically by the gateway or on-demand by the user by clicking the application icon or link from within a portal. In both instances, the gateway must incorporate embedded logic on how to launch the application on the client machine. But launching is not enough: When running a client/server application in application-layer mode (as opposed to over a network-layer tunnel), the gateway should cause the client application to search for the application server in the corporate network through the SSL VPN tunnel. Some tunneling methods require the modification of application initialization files or provide specific redirection parameters to the application, preventing applications launch directly after login.

User activity-based timeout

Inactivity timeouts are best determined by activity on the endpoint such as keystroke entries, mouse clicks and mouse movements. This is not possible, however, in a Web-only scenario where endpoint security policy does not permit download of executable components or applets. In these instances, the server has to rely on HTTP traffic to determine user inactivity and force a logoff through a timeout.

Some popular Web applications poll the server periodically, making it virtually impossible to base timeout and user inactivity determinations on HTTP traffic.

An application intelligent timeout should incorporate specific logic on how to classify polling requests. The gateway should ignore standard polling requests (such as browser-based email applications in determining the appropriate interval for timeouts. Instead, the gateway should be able to identify when no user-based activity has been detected for a preconfigured period of time in order to determine timeouts.

The inactivity timeout should be non-intrusive, giving users the chance to prolong the session without losing their work. This should be achieved through internally generated message windows so that users have the opportunity to intervene if their session is still active.

Conclusion

For further information on Application Intelligence, the IAG 2007, or to arrange for a live demonstration, please contact: whaleinf@microsoft.com (USA) or whaleeur@microsoft.com (international).

IAG 2007 Application Intelligence-WP-200702.doc