Unified Access and Application Delivery Methodology:
A New Paradigm in Information Security and Network Architecture

Overview

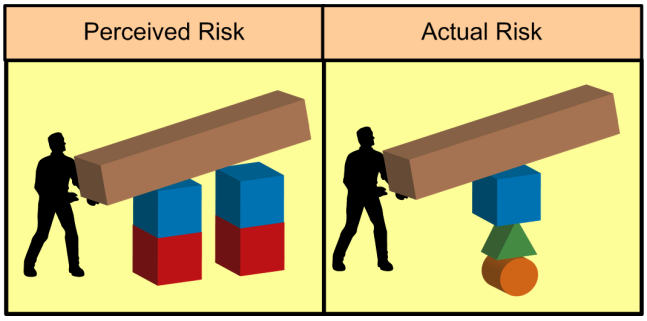
The concept of Unified Access Control or Network Admission Control is being talked about a great deal within the information security and network architecture industries. Many manufacturers are proposing their point solutions and products help to better the security posture of traditional network-based paradigms by repositioning traditional solutions like remote-access technology and applying it to the internal LAN environment. One can hardly read anything about network architecture or information security without at least a passing mention of "de-perimeterization" or "re-perimeterization". The problem with all of these discussions and announcements is that no one, to date, has really defined the depth and breadth of the pitfalls with current architectures and how these new solutions may solve them. No one has defined what a unified access control network is, how it should work, why it is superior or even attempted to give us a common lexicon to discuss any of these issues.

Challenge

Why Traditional Security has Failed

If it weren't for the advent of distributed computing and its culmination into the Internet, the world of information security would be a sparsely populated field of study, limited primarily to physical-security and user-management specialists. It was only with the dissemination of information and processing power that today's security concerns have arisen. This constantly changing and evolving security dilemma has resulted in the reactivity that has been the hallmark of the information security industry. The primary drivers behind these issues are the difference between perceived risk and actual risk, the lack of extensibility of information security solutions, the complexity of modern business systems and their interaction, and the fact that security decisions tend to be made in the vacuum of "pure security" without contemplating the whole of the system.

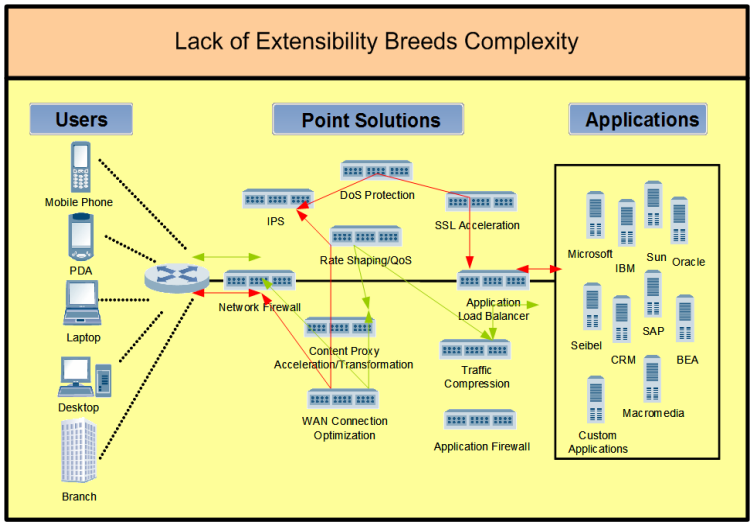
The first problem is the difference between perceived risk and actual risk. As any system is being developed, most modern businesses take great pains to understand, assess and address the risks of deploying the system. Most systems, however, fall prey to either a lack of due diligence or a lack of due care. Due diligence, or the process of determining all the risks associated with the system, often falls short of identifying all the true risks of the system. This is most often a fault of not understanding the technology to be deployed or the reliance on 3rd party technology that has not been adequately vetted for possible exposure; it is difficult, if not impossible, to determine that a technology or product has an inherent flaw or bug that poses a significant risk prior to mass market adoption. Due care, or the process of determining the cost/benefit of mitigating identified risks within the system, also often falls short of correctly prioritizing or assessing the impact of the possible risks. This is most often the fault of insufficient knowledge, misinterpretation, or the dynamitic nature of the financial, political, social and legal ramifications associated with exposure from a risk. The loss of personally identifiable information had no real impact on corporate America until changes in the political and social climate caused regulations and laws to be enacted that created financial liability. Systems designed around a risk assessment in the absence of these regulations and laws will not have the same prioritization of information security risks as those created or analyzed after. These two basic issues create a riff between what information security experts perceive the risks to be and the real risks presented by business applications.



Most business applications also lack the extensibility and flexibility to adapt to risks not understood at the time of their development. This has led to the cottage-industry of specific function devices

that are either added to the network architecture or "bolted" to the side of existing applications; and even these solutions often lack extensibility and flexibility. As each new technology trend delivers a new risk, we must add new systems and processes to the infrastructure to address them. We are left with an array of firewalls, application firewalls, IDS/IPS, Anti-Virus systems, Anti-Spyware systems, proxies, Single Sign-On systems, authentication systems, wireless security solutions and others, with new ones arriving daily. Without the ability to dynamically and easily adapt to new threats the time it takes to mitigate them is now often longer than it takes for malicious attackers to exploit the system.

This brings us naturally to the third basic problem with the information security practice today: complexity is the enemy of good security. The complexity of the modern security infrastructure—encompassing myriad of devices and applications—creates significant issues in the creation, implementation and management of an enterprise-encompassing security policy; and auditing of such a policy is nearly impossible to accomplish. Complexity not only lessens the assurance that security policies are implemented correctly and appropriately, but the complexity in the interaction between security devices can, in its own right, create a risk. Let us also not forget that complexity is also the enemy of availability and performance. As more and more systems become involved in the application of security policy, the reliability of the system (in terms of availability and performance) reacts inversely; troubleshooting of such a complex system to address reliability issues becomes as complex as the system itself. These complexity issues combine to create a system that requires constant attention and management to even ensure that it is running appropriately regardless of whether the security policies are correctly implemented and being enforced.



The final issue with current security and network architectures is that they are developed independently from one another and often have opposing aims. Network design is about connectivity and allowing access to resources while security design is concerned with limiting that access. Network architects often use technology to increase performance, throughput and availability which is inherently contrary in design to the goals of the security architect. A caching proxy, for instance, helps to deliver content more efficiently, but does so by putting copies of potentially confidential data outside the bounds of its original security context on a system that the security architects have no control over. On the other hand, most security implementations are undertaken without any consideration to the impact such solutions have on the value of business applications in terms of performance and availability. Because these two groups design with different aims—and then have to find a way for them to interoperate—both designs, no matter how good, become less than optimal. The whole is much less than the sum of the individual efforts.

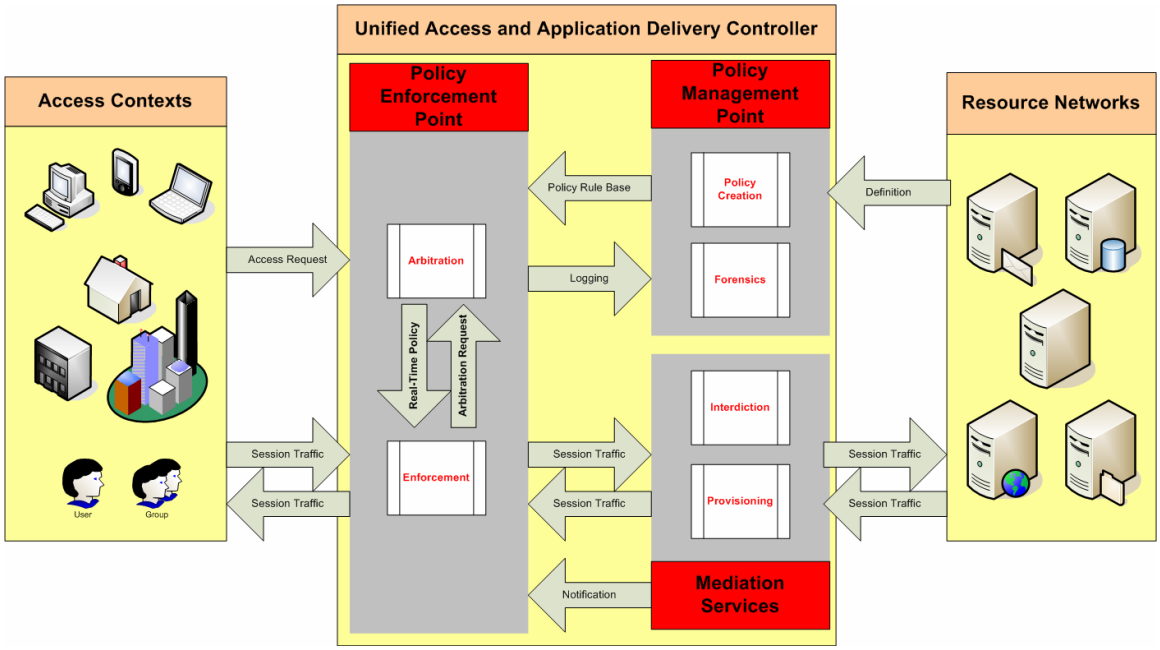
As business systems—and the access to those systems—continues to expand, the issues with complexity, extensibility, design inconsistencies and inability to assess risk correctly combine exponentially within this "bubble-gum and bailing-wire" approach, not to solve issues of security and application delivery, but to become the most prevalent issue in modern network and security design. There must be a better way.

Solution

Unified Access and Application Delivery Methodology

The critical security flaw in today's network design is simple; it wasn't designed to be
The unified access and application delivery methodology (UAADM) revolves around, not the network per se, but how the network is used to connect users and the applications they need, the context with which that access is requested and granted and the security profiles that accompany the context and the resource being accessed. This methodology design breaks the process into three distinct constituent parts: Access Contexts, Resource Networks and a Unified Access and Application Delivery Controller (UAADC).

Access Contexts include the access devices and users themselves. In addition, you have contextual-based information that accompanies access requests between the device and the resource. The contextual information includes not only the traditional information (source, destination, port and user authorization), but also dynamic information (integrity state of the access device, type of network used, level of encryption for securing communications, etc.). Resource Networks are simply collections of individual resources that are defined for access and the requirements necessary to access them. The Unified Controller is the central device which examines the access context, compares it to the available resources and defines what resources are available and how they may be accessed.



The Controller: The Power of Intelligence

A Unified Access and Application Delivery Controller (UAADC), or Unified Controller (UC) is a single physical or logical boundary between the consumers of application services and the devices which provide those services. While this is consistent with legacy ideals such as the network firewall, the difference is the intelligence used to determine which services are accessible to which consumers. The controller itself is the synergistic combination of security and application delivery services through three basic processes: Policy Management Point, Policy Enforcement Point and Mediation Services.

The UC has the ability to interrogate the user and device making an access request to determine who they are and what the integrity state of the device is at the time of access. Using this information, in combination with the physical port or VLAN origination of the request, the controller can match the current context of the request against the list of available resources and their access requirements. The arbitration process allows the controller to intelligently permit only traffic

that the current access request context is valid for. A unified approach applied to all requests for services provides a streamlined and simplified method of access control while adding additional intelligence to the process. This unification and fortification of access control is major advantage to the design methodology; however, addressing the security of the system without addressing the network needs of application delivery is still only part of the solution.

Since the access control mechanism of the controller inherently knows the context of the access request and the specific set or sets of application services that need to be delivered, it is also in the enviable position of being capable of intelligently applying application delivery and service specific security services. By understanding the environment of the access request, the controller is perfectly suited for implementing services like caching, compression, encryption and QoS—and only applying these services to the traffic that needs them, the contexts which will benefit from them and where they will not compromise security. Compression, for example, does not produce consistent results across all contexts for all traffic; its benefits are lost to most broadband users or content which is, by nature, already compressed. The controller can use the context of the access (broadband vs. dial-up) as well as the requested application service (highly compressible vs. non-compressible) to determine if compression services should be applied to an access request.

In the same manner, the controller can intelligently apply interdiction services to traffic, but only when and where it makes sense to do so. HTTP traffic could be routed through a web application firewall service, file transfers through anti-virus screening services and traffic from less-trusted contexts might be directed through IDS services as a prerequisite for their ability to access an application. Once again, having the ability to know both the context of the request and the specific services gives the controller a whole new dimension of intelligence about the transactions being performed and their validity.

Lastly, the UC must also monitor the traffic content and changes in the context of the access request. An access request originating from the Internet, using mobile WiFi access, might roam from a high-latency, high-loss link (in which compression services would be beneficial) to a low-latency, low-loss link (in which compression might provide negligible benefit); or, traffic utilizing an IDS service might trigger an alert. Either of these cases, and hundreds of similar scenarios, either change the context of the access request or represent potential threats within the content of an access request that the controller must adapt for. The controller provides great flexibility to implement whatever response is appropriate. If the traffic processed through IDS services is not the only authorized traffic, perhaps the proper response would be to simply terminate access to that specific service; on the other hand, it might be more appropriate to simply terminate all access, dynamically start monitoring other traffic from that context with IDS or simply do nothing other than log the event (including the detailed context and application service involved).

Addressing the Issues

Changing the entire philosophy of information security implementation and application delivery without showing that there are problems in the current methodology is certainly a losing prospect. Highlighting problems without suggesting solutions to address those problems is also insufficient. So how does the UAADM address the issues?

While simply changing the methodology used to secure the enterprise will not necessarily fix the problem of exhaustive risk identification, UAADM does mitigate the impact of those issues by addressing the remaining three: lack of extensibility, complexity of design and disparate network and security designs. Using “pluggable” mitigation services allows the controller to easily adapt to new threats and new mitigation technology without the need to redesign the entire network or the addition of yet another appliance in the path of all traffic. This allows organizations to quickly react to previously unforeseen risks without changing user experience. In addition, the ability to integrate new functionality into the existing process drastically reduces the complexity of the environment and enables a single, unified, enterprise-wide policy giving organizations unprecedented capability to analyze, define, manage and audit their security posture. An

integrated environment that also accounts for mitigating the impact of security on the performance and availability of enterprise application services also provides a convergence between the network and security enclaves—a common ground on which to build services which best support the business' needs while protecting the business' assets.

The Unified Access and Application Delivery architecture is a revolution in the application of access control and application delivery; the time for revolution has come.

UAADM: A Holistic Approach

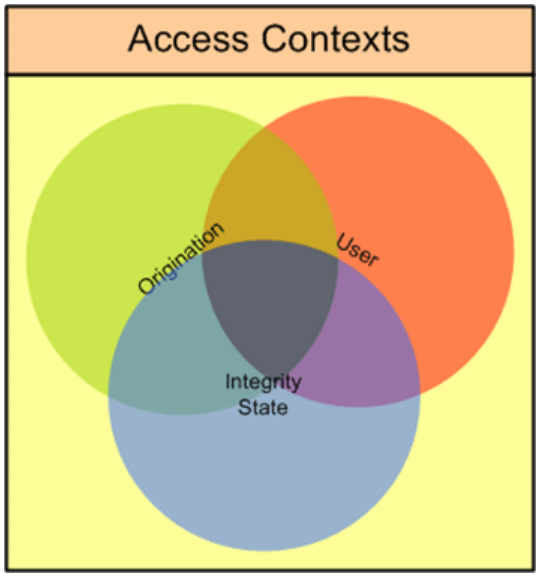
The UAADM represents multiple changes in perspective when looking at how to apply access control and application delivery to the enterprise architecture. While unified access control is a hot-topic within the industry, most proposed solutions focus solely on the security implementation, dismissing the implications to application performance and availability. These solutions, by adding more devices to the network or requiring redesigning of existing architectures simply perpetuate the failures and limitations of the past. You cannot discuss security without addressing performance and availability. Even the hallmark definition of security, Confidentiality, Integrity and Availability (CIA), suggests that simple fact. The disconnect is that very few of these solution providers have the ability to address the problem in a holistic fashion and even fewer understand the need. The power of the design, however, is in the combination of the constituent parts.

Access Contexts

As enterprise networks continue to proliferate and mobile technologies permeate every aspect of our lives, it is increasingly important to know more about access requests than simply to authenticate and authorize the user. This is true from both a security context as well as an application delivery context. From the security viewpoint, it is just as important to know where the user is making a request from, the type of device the user is using, the type of connection and the type of information they are accessing. These same characteristics, coincidentally, are also important from the application delivery standpoint. Access Contexts are the combination of these characteristics to create a clear picture of the access request.

Users: Being able to classify and restrict certain access attempts based on user authentication is a mandatory component of access control. While not all resources may require specific credentials, many will.

Origination: It is also important, in many situations, to restrict the device that has access to the network based on the locality, type, etc. It may be necessary that certain systems within the network are not allowed to access certain information due to its physical location, access media or operating system. Knowing whether the system is attached to the local LAN versus WLAN or even via the WAN is a critical component of the context.



Integrity State: Increasingly, it is necessary that you are able to classify access attempts based on the ability to verify the integrity of the machine itself at the time of access. This involves being able to ascertain whether the system has anti-virus running (and whether it is up-to-date), whether the system has personal firewalls, anti-spyware, anti-malware, whether the operating system is up-to-date, etc.

The context of an application request gives you the intelligence to apply appropriate security (make sure that AV is running), but it also gives you the intelligence to account for application delivery. If the client is running a trusted AV engine that is up-to-date, isn't running that user's traffic through an enterprise AV engine simply adding latency and no real value? Security and application delivery are inseparable—but you need the intelligence to gather the information and the control to act on it.

Access contexts, by themselves, are not "valid" or "invalid"; they are simply an ephemeral state that the controller uses to arbitrate access requests. Whether the context is valid or not relies entirely upon whether any resources are available given the context of the request.

Resources and Resource Networks

Resources are obviously a critical component of an access control and application delivery solution. Without application services to be accessed, there isn't any need for access control or application delivery. Under this methodology, resources are also extremely critical and, other than the policy rule-base (discussed later), the only static, completely definable quantity in the access control and application delivery equation. Resources are simply the definition of services to be offered.

The definition of resources in this design, however, includes much more than the traditional attributes (IP or hostname, ports, protocol and user/group). It also includes the definition of context requirements, interdiction services (discussed in more detail later) and provisioning services (also discussed later). This allows the application owner to define, not only the services to be offered, but the access contexts that will be allowed to access the resource (must have AV running, be from a corporate resource and/or apply protocol sanitization) as well as how the traffic will be processed (must go through IDS, Web Application Firewall and/or must be encrypted from point-to-point; request QoS and/or compression services). These attributes are the key to mapping an access request to a resource based on the context and defining how that traffic will be handled by the controller.

Since each resource defined is an application service, a single, physical machine could potentially have multiple resources associated with it; each one with potentially unique requirements, but sharing some basic attributes. Additionally, in load balancing and Service Oriented Architectures (SOA) implementations, multiple physical machines may have similar resource definitions, or the organization may want/need to define high-level definitions for multiple resources within certain networks or multiple resources owned by a single business unit. Resource Networks are container objects which alleviate some of the tediousness and complexity of defining each and every resource. Resource Networks do not define resources themselves, but allow the definition of attributes that apply to any resources within the container.

Once resources are defined, the attributes that define them determine the potentially valid access contexts that may access them. This is the domain of the Unified Controller.

Unified Access and Application Delivery Controller

The Unified Access and Application Delivery Controller, or simply the Unified Controller (UC), provides the bridging of access contexts and resource definitions into a holistic, unified solution. It represents the physical and/or logical collaboration of three unique elements necessary for providing the intelligence, integration and management of a unified architecture. Those elements are: the Policy Management Point, the Policy Enforcement Point and Mediation Services.

Policy Management Point: Business "Rules"!

The Policy Management Point (PMP) is the main point of contact for configuring and auditing the UC and, consequently, consists of two specific functions: Policy Creation and Forensics.

Policy Creation is the component of the PMP that provides for the definition of resources, resource networks, non-resource specific rules and system configuration functions that control the entire access control and application delivery architecture. Non-resource specific rules allow the definition of global, all-encompassing security restrictions that over-ride resource specific attributes. This allows an organization to put in place basic, business rules that dictate the general behavior of their information architecture and gives them the ability to implement compliance controls that cover all aspects of information access. These might be enterprise wide authentication sources, attributes dictating restrictions to certain parts of the network from specific contexts or default responses to triggered mediation events (terminate all access, terminate service or simply log). Policy Creation also provides the capability of archiving rule-bases, managing multiple rule-bases (it is possible to apply different rule-bases to different Policy Enforcement Points based on architecture) and pushing rule-bases to Policy Enforcement Points (that may reside outside of the physical system that the PMP resides on).

The Forensics function is responsible for the detailed logging and correlation of all Policy Enforcement Events (including mediation notification and response) as well as all PMP events (who configured what and when). The level of detail possible in a UAADC system is what elevates this component from being simple "logging". The Forensics function has the ability to log the entire access context, the specific services accessed, any mediation events, the services that triggered the events and the system response to those events. This gives a very detailed picture of what has transpired and leaves little other "correlation" necessary. Forensics are essential in enforcing compliance, providing assurance and enabling the business to take action on misuse of the system.

Policy Enforcement Point: Real-Time Intelligence

While administrators, application owners and security personnel may see the PMP as the "brains" of the UC, it is really only the source of the rule-base which guides the intelligence of the Policy Enforcement Point (PEP). The PEP is the critical component that makes the policy rule-base come alive and become real-time policy for any access request. This "intelligence" is comprised of two basic functions: Arbitration and Enforcement.

Arbitration is the process that determines if a valid context exists in relation to the application services available. There are two inputs into this process, the rule-base from the PMP and the results of client interrogation. Client interrogation is the process which allows the PEP to determine the current context of the access request and entails passive information gathering in combination with active client "probing". Passive information can be determined from the request itself, like the source IP address, the physical port or VLAN the request is made from or the characteristics of the TCP connection. Active information must be provided by the use of agents pushed down to the client and can discover more specific client information like the presence of AV and its current operational state (running, infected, etc.). Because client interrogation can potentially present a performance hit for application access, ideally the PEP would only attempt to interrogate for information that it knows is required by the rule-base it received from the PMP (e.g. if rule-base doesn't have any restrictions concerning the use of AV on the client, then it shouldn't bother asking about it). Once client interrogation is complete, the arbitration process identifies any potential resources that the current context is valid for—and if user authentication is required for any of them, requests authentication from the user. Arbitration therefore identifies any and all resources that the current context is valid for and the user is authorized for and pushes this information to the Enforcement module in the form of a real-time policy for this unique access request.

While Arbitration provides the basics for access control based on the access context and the resource definitions, Enforcement ensures that only the identified resources are accessed and applies mitigation services to control how those resources are accessed. The most basic function is to take the real-time policy from the Arbitration process and use it as a basic packet-filtering template only allowing communication from the client to the identified resources on the ports

specified and applying standard network-layer security provisions (SYN-flood protection, DoS protection, etc.). Secondly, the Enforcement engine ensures that traffic from the client is processed according to the real-time policy. If the real-time policy mandates that FTP traffic to a specific resource be processed through a protocol sanitization service, the enforcement engine ensures that this mediation service is applied to that traffic; if file transfer traffic is mandated to be scanned by an IDS system, the enforcement engine makes that happen.

Enforcement also is responsible for the ongoing monitoring of context and content to make sure that the real-time policy is correct and relevant. Context monitoring is accomplished through periodic context validation and/or through event triggers from active interrogation components as they record a change in context. Content monitoring is accomplished through events from mediation services signifying that something in the traffic flow has changed or has been identified as potentially malicious. In most cases, these events will require the enforcement engine to request re-arbitration of the existing access—resulting in a new real-time policy or, potentially, termination of access. This is the real "intelligence" of the UAADC.

Mediation Services: Defense in Breadth, Perhaps?

Even if all the current security and application delivery functionality could be built into a single, scalable and reliable physical appliance, the reality of modern network demands and security threats would make it obsolete long before it made it to market. We've already demonstrated that this "closed box" strategy is a key component in the failure of modern security and network design. Mediation services provide the extensibility and flexibility to add new services to the controller to adjust to changes in these demands and threats. There are two current categories of mediation services: Interdiction Services and Provisioning Services.

Interdiction services, as the name might suggest, are real-time traffic processing services which primarily relate to additional layers of security beyond what is provided for in the basic PEP deployment. One example previously mentioned was the application of protocol sanitization to access traffic. Such an interdiction services would become part of a specific resource access flow and verify that the traffic is indeed the type of traffic allowed. The results of this Interdiction process is binary; either the traffic is valid or it isn't. If the traffic isn't valid, the Interdiction device simply notifies the PEP of the fact and the PEP, based on the real-time policy, determines what to do with that information. Other examples of possible interdiction services would be: Anti-Virus Scanning, SPAM mail filtering, Application Firewall processing, IDS scanning, etc.

Provisioning services are real-time traffic processing services which primarily relate to application delivery functionality: compression, caching, rate-shaping, QoS, load balancing, etc. Just like Interdiction services, these are ancillary services that may or may not reside within the physical appliance where the PEP resides. The real-time policy specifies which traffic should utilize which (if any) provisioning services and the PEP ensures that traffic is processed by the service. Just like Interdiction services, provisioning services simple become part of the resource access flow and handle traffic as prescribed.

Mediation services allow a single-touch system like the UC to still apply the same application delivery and security tools that exists in the modern network, but to apply them intelligently on an "as needed" basis. This not only provides a unified platform from which to manage all of your access and application delivery decisions, but also provides for the economic use of processing power and services.

Separation of Duties

The UAADC provides for a robust and dynamic access control and application delivery system, but it also represents a delicate balance between application owners, security managers and network administrators. Fortunately, its design makes it inherently positioned to provide a neutral alliance between these groups.

Application owners are ultimately the people who know what services they have to offer and the conditions under which they want to offer them. By providing for "resource roles" in the PMP that only application owners can access, they are easily able to publish new services by simply entering the resource definition. In addition, they have the ability to specify the security restrictions they feel are necessary as well as request any provisioning services they feel they require.

Security administrators require the ability to apply enterprise-wide security mandates, control user authentication requirements and audit access to the information systems. By providing security administrators with "security roles" that do not allow them to change the resource definitions, but apply higher-level, enterprise or network-wide rules using resource networks or resource-independent rules, security administrators can affect the "global" security requirements they desire. They would also be able to see and report on the Forensics of the system, and provide configuration of additional mediation services that they might own.

Network administrators usually end up physically deploying technology like the UC in addition to many of the mediation services since they don't fit neatly into anyone else's domain of responsibility. Consequently, network administrators need the ability to configure the physical deployments of the devices including adding external mediation service configurations to the system. Network administrators also need the ability to monitor and evaluate the performance of the systems, capacity and the effects of provisioning services.

Because these actions are all interrelated, but independent actions, the UC provides a unique opportunity to compartmentalize the responsibility of these functions and provide each user the appropriate level of accessibility to the system. Application owners can now publish their services without the need to ask for security or network administration approval as their services would inherit the global settings. Security administrators can apply enterprise-wide policies and security enforcement with transparency for the application owners and network administrators. Network administrators can deploy new mediation services and PEPs within the network without having to worry about which services will use them or needing to configure a policy. Auditors can watch them all.

Identifying Unique User Sessions

It has been assumed that each individual user session was capable of being defined and managed separately from all other user sessions. In fact, the ability to generate unique real-time policy on the context of each request and apply mediation services based on that context, absolutely requires such a capability. There are two unique schools of thought concerning the subject.

The first school of thought relies on the 802.1x architecture that is the basis of many Network Access Control (NAC) models today. Using the 802.1x architecture, it is possible to isolate users based on the control of physical switch port configuration after the user is authenticated/authorized in some manner. This allows for the discrete identification and control of an individual user session and the ability to isolate it (via VLAN controls and port forwarding restrictions) from the other users on the network. Unfortunately, the NAC model fails to take into consideration the fact that most, if not all, access requests from remote networks are physically attached to hardware beyond the management scope of the enterprise (i.e. the enterprise doesn't own the switch ports in a users home or at the local coffee shop). The NAC model also presents issues when talking about "public" access on the corporate network (for consultants, visitors, etc.); these users are unlikely to have credentials to "attach" to the physical network and therefore not have any access or require "holes" to be made. While this might be perfectly fine based on the security doctrine of the organization, some may want more flexibility. The NAC model also does not specifically provide for transport security. All transactions remain viewable over the network to their destinations which is unacceptable with remote access and wireless access attempts and unadvisable with local network attempts. Finally, the NAC model is based on the premise that all physical network devices are capable of participating within the 802.1x environment which simply isn't the case; it

would require many organizations to spend countless amounts of money to upgrade existing components. These issues result in an inability to apply "unified" processes to all access requests.

The second school of thought derives its basis from the VPN marketplace. The idea here is to leave the "physical access" network wide open for anyone to use, but require users requesting access to corporate resources to attach to the UC and establish a secure tunnel with it. The secure tunnel does two things in this case: it provides transport security by encrypting all traffic; and it generates a unique, controllable and isolated session for the user from other users. This is often referred to as the "Encryption Everywhere" model. This model is generally agnostic about where the access request originates from and allows the access device to become part of the enterprise network without requiring control of the physical attachment points or intermediaries. This model also has some potential drawbacks. First, it typically requires the user to know how to access and authenticate to the UC prior to attempting access to any resources. The second is the fact that prior to authentication and subsequent isolation via the encrypted tunnel—all users have free-reign. This means that the physical access network could potentially become the "wild west" where viruses, malicious users and miscellaneous Internet threats pervade without any real control by the organization even when they have ownership of the devices themselves. This could allow for misuse and attacks on the physical network resources.

Since the NAC model can't account for, or provide real security for remote users, the VPN model provides the most unified, consistent approach to all access requests. Since the new methodology is designed based on access contexts, it seems appropriate to implement the VPN model, but to add 802.1x information as a context attribute for further refining access control. This makes it possible to have a unified process regardless of the availability of 802.1x information, but to account for it only if it is available.

Security Philosophy

Most people inherently understand the "defense in depth" strategy to enterprise security and point to the fact that its basis is a tried-and-true methodology for security dating back hundreds of years. What most people fail to realize is that a computer network is quite a bit different than a castle or other physical building. The original intent in many of these "defense in depth" strategies was to slow down the attackers in order to give the security forces time to react and defend. Unfortunately, these same strategies also slowed down legitimate traffic even when active defenses where not being deployed. This has been the primary conflict between the network and security teams—and the fact that security practitioners miss this subtle fact is proof positive of the issue with making security decisions in a vacuum.

Unlike a physical building, the UC can use context to erect defenses specific to and only applied to unique users or unique transactions. Because the UAADM invariably uses context to apply security, it can still provide the "defense in depth" strategy, but also dynamically change which defenses are deployed based on the threat presented by the unique session. In this way, you get all the benefits of the defensive strategy, even being able to add additional depth dynamically, without negatively impacting all other non-threatening traffic. Context gives us a much better understanding of who is approaching the castle and this methodology gives us the ability to match the security to that specific person.

Analyzing the Existing Market

Despite the marketing hype and vendor positioning, no one has yet delivered a complete, unified UAADM compliant solution; that's not to say that you cannot approximate it with current technology, but that it still requires multiple boxes with multiple points of management, policy creation and audit. It is relevant then, to examine what is available today and what is still missing. In order to do this, we should reiterate what is required to have a complete solution, then discuss what is currently available and what remains to be delivered.

A Complete Solution

In order to have a complete UAM compliant solution you need the capability, within a single physical or logical device to do the following:

1. Interrogate the Context of Access
2. Arbitrate between Access Contexts and Resources
    a. Must provide unique, individual session management
    b. Must provide session (contextual) based access control
3. Enforce how Access will be Provided
    a. Must enforce a real-time, session specific policy
    b. Must provide for extensible mediation services
    c. Must react to changes in context or content
4. Unified Management
    a. Must provide for unified policy creation across all functions including Interrogation, Arbitration, Enforcement, and Mediation
5. Correlated Forensics
    a. Must log all events, both user and management, across all functions

What we Have?

As previously stated, we already have many of the pieces, just not a complete solution. So, what are those pieces?

Requirements 1 & 2, Interrogation and Arbitration: these are current components that should be recognizable to anyone who has evaluated or deployed SSL VPN technology. These devices routinely implement some form of client integrity scanning and use the information gathered as a determinant in the resources that the client is allowed to access. Implementations may vary, and the granularity of control as described within this paper is not necessarily available in any existing products (certainly not a unified solution), but the basic components are there. In addition, since SSL VPN devices inherently isolate users based on unique encrypted tunnels, they also satisfy that need as well.

Requirements 3, Enforcement: is currently technology that should be well known to users of any of the application delivery controllers (once called “load balancers”) in the market. The ability to decrypt SSL traffic, apply compression and rate-shaping, as well as arbitrarily redirect traffic to different ancillary services based on service and content are all common functions of these devices today. Although most of them lack the dynamic ability to re-arbitrate (since few even allow for the concept of arbitration in the first place), many of them can dynamically adjust what enforcement techniques they apply based on changes in the traffic (e.g. compress text, but not graphics or send HTML requests through a web firewall, but pass FTP traffic through to the destination). None of them, however, can make these decisions based on a real-time, context derived policy.

Requirement 4, Unified Management: this is one of the key missing elements. Even manufactures who have the components to satisfy the first 3 requirements (or pieces thereof), fail to provide a unified mechanism to configure and manage their own devices. Certainly, none of them currently take into account the dynamic ability to apply information gathered on one device to the configuration of another.

Requirement 5, Correlated Forensics: this is one that could potentially be satisfied with third-party systems, but since no device is capable of actually performing all the requirements, these tools would rely heavily on correlation and still not have the level of complete detail that a unified solution would present.

This overview should make it readily apparent that many of the requirements can be met as long as a truly “unified” solution is not the end goal. Many features of the unified architecture could be

simulated by combing products like SSL VPN, Application Delivery Controllers and some form of log consolidation and correlation system. While this falls woefully short of the ideals of UAADM—it is real and could be deployed today.

What's Still Needed?

As we can see, many of the constituent parts needed to create a unified network already exist to some degree. If we also stipulate that extensibility requires us to conclude that a UC, while still maintaining policy management and enforcement, will most likely never completely consolidate all of the supplemental mediation technologies that we will need, how do we manage the difference; create a unified device that allows for independent, adjuvant components?

The answer is simultaneously simple and grandiose. The final requirement to make a unified design a reality is a shared control and data plane architecture that can be used to consolidate and unify these independent functions, either physically or logically, into a single point of reference. The Policy Enforcement Point must be capable, based on policy, to invoke the services of an ancillary process in order to fulfill the requirements of a particular access request and the Policy Management Point must be aware of the presence of these systems. Simultaneously, the subordinate process must be capable of not only providing service but of informing the PEP of the result of such requests. For example, if the policy requires that traffic be processed by and IDS service, the PEP must be able to route traffic through an ancillary process that provides IDS services. This service must also be capable of informing the PEP of suspicious traffic so that the PEP may take action on that trigger. This, above all, is the principle hurdle to making Unified Access and Application Delivery a reality.

Conclusion

To most technophiles, it is obvious that the current shortcomings in information security must be addressed and that some sort of Unified Access Methodology will be the basis for that solution. What isn't necessarily understood is that addressing the shortcomings without accounting for all of them (namely application delivery in concert with security) is not a viable solution. The primary issue will be the creation of the services architecture that allows multiple—perhaps disparate and possibly competitive products—operate as a unified whole. This is no easy task and one that may forever remain elusive as long as independent vendors refuse to work together. That being said, the market will migrate towards devices based on this theory; the difference will be a) how they build the shared data and control planes and what they base it on, b) how many ancillary services they can interoperate with, and c) whether they learn that application delivery is a critical component of the solution. Network and security design will slowly, but surely evolve to a unified design and the vendor who can provide the most services in the most unified manner, addressing the largest number of issues will be the eventual winner.

About F5

F5 Networks is the global leader in Application Delivery Networking. F5 provides solutions that make applications secure, fast and available for everyone, helping organizations get the most out of their investment. By adding intelligence and manageability into the network to offload applications, F5 optimizes applications and allows them to work faster and consume fewer resources. F5's extensible architecture intelligently integrates application optimization, protects the application and the network, and delivers application reliability—all on one universal platform. Over 10,000 organizations and service providers worldwide trust F5 to keep their applications running. The company is headquartered in Seattle, Washington with offices worldwide. For more information, go to www.f5.com.