Secure Coding.
Practical steps to defend your web apps.

Copyright SANS Institute
Author Retains Full Rights

This paper is from the SANS Software Security site. Reposting is not permitted without express written permission.

Interested in learning more?
Check out the list of upcoming events offering
"Defending Web Applications Security Essentials (DEV522)"
at http://software-security.sans.orghttp://software-security.sans.org/events/
Sponsored by Oracle

Smart Strategies for Securing Extranet Access

A SANS Whitepaper – March 2010

Written by: Dave Shackleford

Snapshot of a Secure Extranet

Risk-Based Authentication

What You’re Entitled To

Federation

Advisor:

David Rice,
SANS Institute Leadership Council
Director, Monterey Group,
Author, “Geekonomics: The Real Cost of Insecure Software”
Introduction

In today’s competitive business landscape, organizations are compelled to share data, files, and applications with external partners, customers, and remote workers. Primarily, they are doing this over extranets. In one report, Cathay Pacific Airways implemented an extranet to increase travel agent and partner adoption of online booking and realized an average savings of over $1 million annually.¹

What exactly is an extranet? Simply stated, an extranet is a private network with services, applications, and data made available to external users—often via a private web portal.

With numerous types of users accessing the same web-based portal environment, authorization and entitlement policies must be accurate and complete. Policies that are not granular enough, or permissions that are not designed or managed appropriately, could easily lead to data exposure or other types of compromise. According to Verizon’s 2009 Data Breach Report, 22 percent of breaches involved privileged misuse by insiders. In addition, 32 percent of external breaches implicated business partners.² The same Verizon report states: “A very large proportion of attackers gain access to enterprise networks via default, shared, or stolen credentials.”

Business-to-business portals can also be problematic. For example, small and medium businesses banking through their business banking portals lost $100 million to thieves accessing their credentials last year, according to the FBI.³

This paper discusses how to use risk-based authentication and entitlement management to enforce authentication security and achieve granular authorization using centralized role-based policies. Together, these security technologies can help organizations secure popular portal and collaboration technologies used for extranets, such as Microsoft SharePoint, and meet audit and compliance requirements at the same time.

³ searchfinancialsecurity.techtarget.com/news/article/0,289142,sid185_gci1373452,00.html
Snapshot of a Secure Extranet

In many current extranet deployments, the major architectural components consist of web services (the presentation layer), specific application services (such as customer relationship management and enterprise resource planning applications, search and indexing services, etc.) and back-end components such as databases and Active Directory servers.

The two pivotal security functions that should be designed within an extranet are authentication and authorization:

- **Authentication** is the process through which a user provides credentials that definitively identify him/her as a legitimate user. These can be integrated with numerous other factors, as will be covered later.

- **Authorization** is the determination of what role(s) a user may possess after being successfully authenticated and what privileges and actions are permissible within the role definition(s). It may also include dynamic factors such as level of risk when granting permissions.

Authentication is often left to whatever native methods are in place within the environment (such as LDAP or Active Directory lookups), and all authorization decisions reside within the individual applications. In addition, some organizations have integrated services like SharePoint and others with single sign-on (SSO) solutions that manage credentials for multiple services and integrate with user repositories.

There are, however, a number of shortcomings with this traditional approach that can create problems in large enterprise extranet deployments. Here are some examples:

- With many users accessing the extranet from multiple locations, simple usernames and passwords are no longer adequate. Additional access management controls, such as evaluating contextual data (location and computer used to access), are needed to prevent fraud.

- With multiple types of users needing different levels of access to a wide variety of data via the extranet, defining roles and privileges in a secure fashion is difficult. When this task is left to the individual applications, the roles that can be defined are often not granular enough. Management of multiple role-based systems is tedious, so for convenience, users are often given permission to all resources or to groups that include data types they shouldn’t have access to.
Lack of support for existing user data stores raises the costs and time needed to manage user identities. Services and applications leveraging these stores would also have to be modified. Access management and authorization systems need to integrate with existing user directory stores such as Lightweight Directory Access Protocol (LDAP) and Active Directory (AD). In addition, they should support current and developing standards such as XACML (Extensible Access Control Markup Language) for externalizing authorization controls from applications.

Many authentication schemes lack detailed audit trails. Auditors are closely scrutinizing access to resources covered by policies and compliance specifications. They need to see audit trails that clearly demonstrate which users access which resources, along with the authorization decisions and actions behind their access controls.

There are a number of new policy, access, and authentication components involved in securing today’s extranet architecture that can alleviate these issues. Figure 1 provides a conceptual diagram of the major services that organizations need in a more flexible and extensible authentication and authorization scheme for extranets.

Figure 1: Secure Extranet Components
Let’s take a brief look at each of these components.

• **Access Management Gateway.** In Figure 1, a user accesses the front-end Access Management Gateway as the initial point of entry into the extranet and is prompted for credentials (1). This is an optional service that can act as a front end for all web portals and other extranet services. Alternately, this can be integrated with some existing applications or not exist at all. In the latter case, credentials are encoded on the client side and simply passed through more traditional web-based extranet services (such as SharePoint) to user and policy data stores and risk analysis services.

• **Authentication Risk Analysis Services.** An additional platform that runs software integrated with user and policy data stores, a risk analysis engine leverages numerous factors involved in user access requests (such as user behavior profiling, system information, geographic location, browser and cookie data, etc.) to calculate the likelihood of fraudulent behavior. In the diagram above (2), the user’s credentials are passed to risk-based (or other traditional) authentication systems to verify the user’s identity. Once authenticated successfully, the credentials are passed back through the gateway (3).

• **Entitlement.** Once the user is authenticated, there are multiple mechanisms to associate them with their data stores, including:

  - **User and Policy Data Stores.** Such stores are traditional stores of user and organizational policy data, such as Lightweight Directory Access Protocol repositories and Active Directory servers.

  - **Policy Enforcement Points (PEPs).** The PEP triggers entitlement policy evaluation and is usually integrated at the point of user interaction, typically an existing portal login screen or other web-based service. In the diagram, SharePoint is acting as the PEP, where a user’s credentials are sent to the PDP and evaluated against policy (3).

  - **Policy Decision Points (PDPs).** The PDP or Security Module (SM) is the service that manages entitlement and authorization policy decisions when queried by a PEP. SharePoint (the PEP) passes the credentials through for evaluation against policy within the Security Module (4). A policy decision is made and passed back to the PEP for enforcement and allocation of privileges.
**Policy Administration Points (PAPs).** Within an entitlement infrastructure, the PAP is the policy definition server through which administrators can manage policy pushed out to the PDPs.

**Policy Information Points (PIPs).** A PIP may consist of any system or application that provides additional data to a PDP for evaluating authorization and entitlement policy decisions. Examples may include risk analytics; user data stores; authentication services such as RADIUS; applications with specific user, group and role definitions; and so on. In Figure 1, the LDAP and AD user stores are acting as PIPs, which contribute information to the PDP.

- Traditional Authentication Services. Ranging from traditional two-factor, token-based solutions to RADIUS and simple username/password solutions, traditional authentication services can still be integrated with extranet architectures, although more risk-based, context-driven solutions should be on the roadmap.

In many cases, organizations have already made investments in consolidated authentication tools such as SSO, and numerous existing applications employ this technology. To interact properly with other applications and web services, extranets need to support standard protocols such as Security Assertion Markup Language (SAML), XACML (Extensible Access Control Markup Language), Simple Object Access Protocol (SOAP), and others. Auditing permissions (who can do what within an extranet application environment) are critical as well—driven, in many cases, by compliance regulations. Gartner analyst Roberta Witty says, “It’s the regulations that have really brought this to a head in the last couple of years because when the auditor says, ‘Show me everyone who can access this application and show me what they can do,’ that’s a pretty tall order in most companies.”

All of the authentication and authorization systems in the new extranet security model should produce detailed audit and log data for review by auditors and security teams. With a centralized policy administration console, dissemination and enforcement of role definitions and policy actions becomes much simpler and supports the goals of audit and compliance.
Risks to traditional username and password authentication include:

- **Phishing.** This type of attack, usually propagated through website links or emails, involves harvesting of authentication credentials for account hijacking.

- **Malware.** Malware is becoming more and more customized for sensitive data compromise. Backdoors, keystroke and mouse click logging, and screen capture applications solely focus on harvesting user credentials. One particularly insidious example is the Trojan.Clampi malware, which attempts to steal online banking credentials.\(^5\)

- **Password theft.** These attacks involve stealing system credentials so an attacker can log in later.

- **Session hijacking.** Here an attacker intercepts an active session and impersonates a user currently logged in to conduct illicit activity in the compromised account.

To combat these risks, organizations are working on behavior analysis for transactions conducted in their systems. They are also implementing more stringent access controls for their consumer and business-facing portals in order to improve security and comply with regulations such as HIPAA/HITECH, PCI DSS and the Federal Financial Institutions Examination Council (FFIEC) security mandates.

The threat of litigation is also becoming a driver for more stringent access controls. In September 2009, a Citizens Bank customer was granted permission to proceed with the first-ever court case alleging lack of sufficient multifactor authentication for account protection.\(^6\) The customer’s account had been compromised due to the use of a simple username/password authentication scheme (and, likely, the user shared usernames and passwords with other accounts as so many others do). Lawsuits from businesses whose connections are exploited via a business partner are likely to follow. Extranets require the same risk analytics and authentication controls as those called for on these customer-facing portals. One of the main means of providing these controls is the use of risk-based multifactor authentication.


Multifactor Authentication

The use of multifactor authentication has been gaining popularity in the portal space as threats against them continue to grow. One of the most common methods of providing multifactor authentication is traditional hardware tokens with dynamic PIN code generation. In addition, many organizations are turning to soft tokens, or software-based PIN generation tools that are either on a server in their environment or installed on mobile phones. Often, the second factor becomes the phone in which users can generate PIN codes on their mobile devices when prompted or receive one on the phone via SMS.7

Another traditional method that has been in place for several years is the use of the one-time password (OTP) or “bingo cards” with unique values tied to a user’s account. Some authentication systems generate OTPs and send them via SMS, email, and phone calls. Newer authentication systems leverage “fingerprinting,” which tags, knows and assesses the specific computer a user logs in from. Fingerprinting performs analytics on system data and places a cookie (and/or a Flash cookie) onto an approved system. This allows the user of that fingerprinted system to log in by entering a username and password if no other risk factors are present. When users attempt to log in from systems other than these fingerprinted machines, they are prompted to answer one or more pre-established questions to authenticate.

One final type of multifactor authentication is data based on location. Although somewhat related to the “something you have” paradigm, this newer two-factor model relies on geo-location from IP addresses, ISP connectivity, and other coordinates to correlate users with pre-existing profile information.

The second key element of a more secure authentication structure is the incorporation of risk-based profile and session analysis that evaluates the following factors to determine whether a user should be authenticated successfully:

- **Something you know.** A user’s password or PIN, as well as answers to specific knowledge-based questions, would fall into this category.

- **Something you have.** This has traditionally consisted of a token with changing values or a user’s computer. Secure cookies, Flash objects, location data, and system state and hardware information could all satisfy this authentication factor.

- **Behavior.** An important facet of the authentication risk analysis is the behavior of the user and system(s) the user leverages to access the extranet. Behavior patterns range from the simple, such as multiple failed logins for a user account or from a specific device, to more sophisticated analysis, including geo-location changes, multiple users on a device, and multiple location changes for a specific user within a certain time frame.

These elements, coupled with historical user transaction information, are then matched with defined policies and used to generate an overall risk score for the user profile and session attempt. For instance, user “john_smith” tries to authenticate with the following information:

- Proper username (john_smith)
- Two failed static password attempts, success on the third try
- When not logging in with the regular system that has Flash cookie, the login window prompts the user to answer several pre-established security questions
- Log in outside the normal times the associated user typically accesses the system

The results of this analysis will determine what actions are taken: **allow** authentication, but **log/flag** certain actions; **disallow** the attempt; and so on. Based on deviations from a standard behavior baseline, the user john_smith may not be successfully authenticated.

Several Identity and Access Management (IAM) vendors have technologies that deliver risk-based profiling. When evaluating IAM vendors and solutions, security teams should look for tools to provide real-time risk analytics, behavioral analysis, and authentication strengthening capabilities.

Analytics and analysis can evaluate the level of risk for each individual access request or transaction based on the location, device, user behavior and other factors in real-time. The results trigger proactive actions such as dynamic authorization adjustments or secondary authentication, which can help prevent fraud before it occurs. For example, if a user’s system is compromised and the attacker tries to gain access using the User ID with successive failed passwords, the account could be disabled and SMS messages sent to the user’s phone. Another example might be the login location changing too often in a certain time period, triggering anti-fraud rules that prohibit user access to sensitive resources.
Once a user is successfully authenticated, what happens next? In most cases, the individual application the user is accessing controls the roles granted to the user. For an extranet scenario, this might include a portal application such as SharePoint. Consider a CRM solution integrated with SharePoint for content creation and distribution and integrated into the environment’s existing user authentication and authorization solutions, such as Microsoft Active Directory. With multiple tiers of applications involved, and multiple distinct groups managing each component, the likelihood of errors in role definition and assignment is significant.

For example, consider the role definition described in Table 1, with different types of users who need to access a SharePoint portal application. Each group of users has unique needs within the portal environment.

<table>
<thead>
<tr>
<th>Groups and Users</th>
<th>Roles</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Your Organization</strong></td>
<td>Your Organization</td>
</tr>
<tr>
<td>Administrators</td>
<td>Manage portal servers and configuration settings, local users and groups, Windows Active Directory policy and users, etc.</td>
</tr>
<tr>
<td>Users</td>
<td>Interact with defined portal projects involving partners and customers</td>
</tr>
<tr>
<td>Managers</td>
<td>Interact with portal projects, but also manage specific line of business project components and tasks</td>
</tr>
<tr>
<td>Auditors</td>
<td>Review transactions within specific projects involving sensitive data and review server and portal configuration settings as a “read only” user</td>
</tr>
<tr>
<td>Developers</td>
<td>Add and manage extranet application code</td>
</tr>
<tr>
<td><strong>Partners</strong></td>
<td><strong>Partners</strong></td>
</tr>
<tr>
<td>Users</td>
<td>Interact with defined portal projects involving your organization and customers</td>
</tr>
<tr>
<td>Managers</td>
<td>Interact with portal projects and approve project changes and additions as needed</td>
</tr>
<tr>
<td><strong>Customers</strong></td>
<td><strong>Customers</strong></td>
</tr>
<tr>
<td>Users</td>
<td>Interact with defined portal projects involving your organization and partners</td>
</tr>
</tbody>
</table>

*Table 1: Example Users, Groups, and Roles*
How can these individual roles be successfully set up using granular privilege assignment and maintained with flexible policies in line with the above requirements? How do you handle multiple organizations accessing the extranet with multiple levels of complex hierarchies defined among each group?

With most traditional extranet technologies, solving these problems is often difficult. In SharePoint, for example, rigid authentication and authorization mechanisms make defining multiple different types of administrators for different components very difficult. In many cases, one or more all-powerful users are often created out of convenience, which is in strict violation of separation of duties for PCI DSS and a number of other regulations that call for separation of administrative duties. (Administrators should not have access to critical data.) To serve the larger population of extranet users, SharePoint has a limited set of built-in permissions (read, design, contribute, etc.), but they are not granular enough. Custom permissions can be set, for example, within SharePoint, permissions can be defined for sites, lists, items, web parts, etc. These roles can get unwieldy quickly and are difficult to maintain for large infrastructures.

Without centralized management, permissions are often unspecified; and, if they are specified, management becomes unwieldy. This is where the concept of entitlement fits in. The term entitlement refers to the complete set of resources a user or group is permitted access to. Entitlement enables application layer access to be centrally defined and managed via policies entirely distinct from the extranet applications, web presentation layers, and any other components such as Single Sign-On (SSO) applications.

Why is entitlement important? Extranet applications require numerous roles and privileges for many different types of users. Unless authorization decisions are separated from the portal and other application logic, the different types of policy controlling role membership and permissions will need to be individually defined and managed within the application infrastructure itself. This can quickly become unwieldy and may not integrate with other technologies in place, such as SSO. In addition, traditional applications are not developed with auditing in mind, leaving much to be desired for compliance monitoring of users, groups, and roles accessing the various extranet components.

By centralizing management of policies, leveraging integrated or standalone PDPs to manage policy decision requests from applications, and leveraging additional systems such as LDAP repositories and Relational Database Management Systems (RDBMS) data, fine-grained policy definition and role management can be easily managed and scaled as the organization dictates.
Policies are the key aspect to entitlement rules calling for fine-grained access control to specific resources. Policies are created to perform two specific actions: evaluate authorization levels and map users and groups into defined roles. So, based on the users listed in Table 1, the following is an example of a simple authorization policy in which the Subject (CustomerXYZ_Users) is GRANTED read access to the /SharePoint/ProjectXYZ/Schedule resource if the subject is a member of the general group XYZ:

GRANT (read, /SharePoint/ProjectXYZ/Schedule, /group/CustomerXYZ_Users) if group = “XYZ”

- **Effect.** This is the end result of the policy – whether access is granted, delegated, or denied.
- **Action.** What type of access is granted – in the example, “read” access is specified, but this can also be “write” or others.
- **Resource.** The resources and objects that are being protected.
- **Subject.** Users, groups, and roles that the policy is being applied to.
- **Constraints.** Optional constraints defined for the policy. These are often Boolean statements, evaluations, or other attribute comparisons.

Another very similar type of policy is role assignment. Instead of the Action element, a Role element is included in the policy definition. In the following example, the Partner Manager role is granted for any members of the group CustomerXYZ_Partners trying to access the /SharePoint/ProjectXYZ/Reports resource as long as their level (some defined attribute) is greater than or equal to 3:

GRANT (/roles/Partner_Mgr, /SharePoint/ProjectXYZ/Reports, /group/CustomerXYZ_Partners) if auth_level >= 3
Entitlement should address a number of key areas within each Policy Decision Point (PDP) component:

- **Authentication.** Entitlement solutions should integrate with existing authentication technologies such as LDAP, Active Directory, or more robust risk-based authentication options, as discussed earlier.

- **Authorization and Role Mapping.** These are the key activities performed by a PDP—assessing credentials against defined policies and then returning authorization decisions to each Policy Enforcement Point (PEP).

- **Auditing.** A detailed audit trail of authentication and authorization decisions and transactions is critical for meeting both compliance needs and security best practices. Any robust entitlement management system should provide extensive audit details of all activity pertaining to extranet application access and policy decisions.

- **Credential mapping.** In order to integrate with other authentication systems and applications, a PDP should be able to mask (skew/obscure via tokenization) credentials based on policy of applications.

A PDP may be integrated directly with applications or act as a standalone policy service that provides centralized policy decisions to multiple applications. The PDP can also accept additional data from a number of different applications and systems, referred to as Policy Information Points (PIPs). These are often Relational Database Management Systems, user data repositories like LDAP and Active Directory, and custom stores maintained by organizations. This integration allows entitlement management to be extensible and customizable, especially if APIs are available to integrate custom applications and data sources that may have relevance to policy decisions.

The set of functions that come together to create entitlement policies that are able to scale to new technologies and needs is complex. However, entitlement management will be critical in keeping extranets competitive as they expand to fit new business requirements.

Entitlement management systems must be able to handle multiple user roles, implement their associated policies and map to their credentials. Look for technologies that simplify and externalize application-level security management by removing security decisions from applications and creating a unified policy administration system.
Federation

Because extranets serve such a decentralized population, more organizations are incorporating the concept of Federation—or decentralized authentication—where participating organizations will accept the identity and authentication information sent from another participating organization. Federation programs, such as OpenID, Liberty Alliance and others, allow organizations to share identity and authentication information without mirroring other organizations’ authentication methods, abdicating control of their user repositories, or acquiring the burden of managing user repositories other than their own. The user experience is compelling: Federation allows a user to log in once and potentially access multiple, disparate systems including external partner web sites.

Essentially, Federation extends the concept of SSO to systems and applications across multiple organizations. This allows organizations to extend internal applications to query and interact with systems in partner extranet environments without notifying the user that they’re interacting with a different system.

Consider an example: A financial broker accesses her employer, an online brokerage firm, to check the status of several customers’ mutual funds. To access her account, she provides credentials for authentication and is then authorized to access her user fund manager account. The brokerage firm, in turn, federates her identity and passes the authentication and authorization credentials through to the extranet applications at several mutual fund companies, enabling her to retrieve data and allow the broker to view information about all of her clients’ fund holdings while only having to authenticate once.

Some Federation solutions tie leading SSO and Entitlements platforms together, leveraging a variety of open standards such as the Security Assertion Markup Language (SAML) that provide well-formed and structured authentication, authorization, and entity attribute statements for use between systems and applications in a federation. For extranet applications and environments that support standards like SAML and SOAP, integrating with external users’ authentication and authorization platforms is much simpler, as this information is passed from one entity to another in a standardized way. This information would be accepted and interpreted in multiple organizations’ extranets just like internal authentication and authorization data at PEPs and PDPs within the entitlement framework.

Federation will be valuable in extranet applications with wide, decentralized user bases because ease of use increases due to SSO and business partnerships can be extended through extranet-based trust models. Look to tools that can integrate into leading Federation models and can be managed for critical functions such as provisioning of new users as well as deprovisioning expired users across the extranets in the Federation.
Conclusion

Most organizations are currently doing or plan to do business with remote customers and partners and developing Internet-enabled collaboration environments for project teams. Extranets become more useful as organizations outsource more of their work to contractors, collaborate with outside businesses, and manage large bases of telecommuting workers.

Extranet portals are the technology of choice for many of these business needs. Extranets connect users to everything from development environments to document management workflow systems to CRM and ERP solutions.

With hundreds to tens of thousands of external partners, remote workers, and employees logging into resources via extranets, controlling access to protected resources in these environments is complex. Insufficient granularity in the definition and execution of users and roles has resulted in a number of publicized breaches. In some cases, such as the popular SharePoint model, it’s nearly impossible to achieve basic separation of duties, let alone any granularity in role selection for masses of users without the help of additional tools and technologies.

Proper application of risk-based authentication reduces the likelihood of account compromise and fraudulent activity inside the extranet. Risk-based authentication works by analyzing credentials and behavior to produce an overall risk rating for each authentication attempt. That rating is then compared to policies to grant or deny access.

With the addition of entitlement, extranets can leverage robust and flexible policies for authorization and role management. This ensures that user privilege management is scalable, audit-able, and manageable in even the largest environments—including being extended outside the organization using a Federated model.
Dave Shackleford, Director of Security Assessments and Risk & Compliance at Sword & Shield Enterprise Security, is a SANS Analyst, instructor and GIAC technical director. He has consulted with hundreds of organizations in the areas of regulatory compliance, security, and network architecture and engineering. He’s worked as CSO for Configuresoft, CTO for the Center for Internet Security, and as a security architect, analyst, and manager for several Fortune 500 companies.
SANS would like to thank this paper’s sponsor:

Oracle

To learn more about Oracle’s standards-based Identity and Access Management technologies, visit http://www.oracle.com/identity.
# Upcoming SANS App Sec Training

<table>
<thead>
<tr>
<th>Course</th>
<th>Location</th>
<th>Dates</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>Univ. of California - Davis - DEV522: Defending Web Applications Security Essentials</td>
<td>Online Only, CA</td>
<td>May 11, 2020 - May 16, 2020</td>
<td>vLive</td>
</tr>
<tr>
<td>SANS Network Security 2020</td>
<td>Las Vegas, NV</td>
<td>Sep 20, 2020 - Sep 27, 2020</td>
<td>Live Event</td>
</tr>
<tr>
<td>SANS OnDemand</td>
<td>Online</td>
<td>Anytime</td>
<td>Self Paced</td>
</tr>
<tr>
<td>SANS SelfStudy</td>
<td>Books &amp; MP3s Only</td>
<td>Anytime</td>
<td>Self Paced</td>
</tr>
</tbody>
</table>