macromedia®

WHITE PAPER

Macromedia® Flash® Platform Security and Macromedia Enterprise Solutions

Adrian Ludwig

September 2005

Macromedia, Inc.
601 Townsend Street
San Francisco, CA 94103
415-832-2000

Contents

In a world where most digital experiences fall flat, the Macromedia Flash Platform offers something different. It's a lightweight, cross-platform runtime that can be used not just for rich media, but also for enterprise applications, communications, and mobile applications. The Flash Platform is fueling an increasing number of Rich Internet Applications (RIAs). And as a result a growing number of employees, partners, and customers have access to enterprise data and processes. This access, combined with the requirement to comply with industry regulations such as the Sarbanes-Oxley Act and the Health Insurance Portability and Accountability Act (HIPAA), has enterprises interested in the level of security provided by this framework. The Flash Platform and the Flex product family address this concern by leveraging an organization's existing security solutions and technologies.

Addressing Security Concerns About the Flash Platform

The Macromedia approach is to implement robust security within its own products while avoiding new exposures to the rest of the environment. However, the Flash Platform technologies are not security products—they only leverage existing security tools and approaches that are already in place, while minimizing additional investments in security. For example, the Flash Platform integrates seamlessly into an organization's existing architecture at the browser level through a plug-in and at the presentation tier through Flex software or a static HTML solution with script and Flash (see Figure 1). Security is handled by existing security solutions and protocols (see Figure 2). Because the Flash Platform leverages SSL and authentication technologies and requires no changes to access control or other security settings, organizations do not need to deploy additional security solutions to use the Flash Platform.

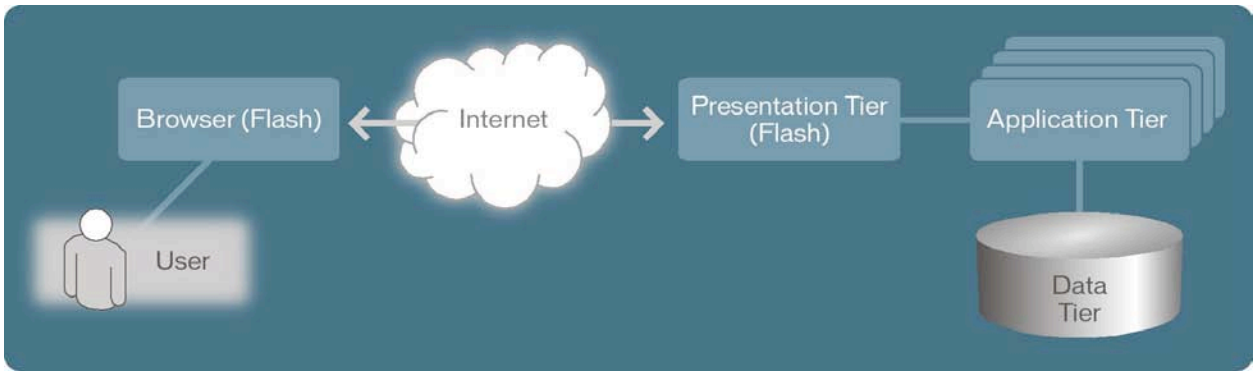

Figure 1: The Macromedia Flash Platform leverages an organization's existing infrastructure.

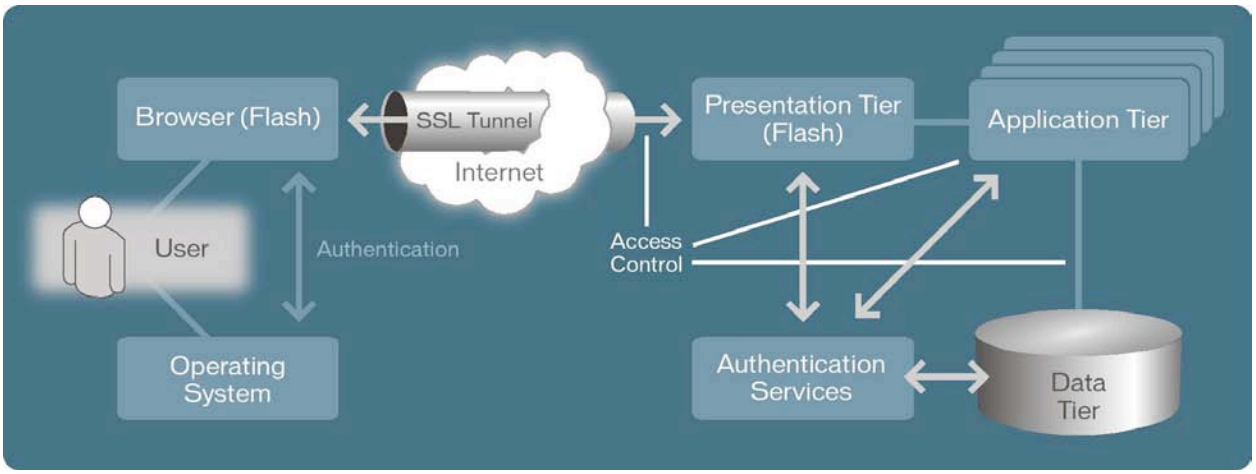

Figure 2: In Flash environments, security is handled by existing security solutions and protocols.

The Flash Platform is a true multiplatform environment that leverages the core security capabilities of the underlying operating systems, browsers, and application servers. The Flash Platform is based on proven and accepted security standards such as SSL and HTTPS for data transport. It has a layered architecture that encompasses the key elements shown in Figure 3. This paper focuses on the servers and runtimes (for example, Macromedia Flash Player and Macromedia Flex software), which are used to deliver Flash applications, content, and communications, and which act as the platform, provide the controls, and specify the architecture. The paper also includes examples of solutions such as Macromedia® Breeze™ and Macromedia® FlashCast™ that are implemented using this framework.

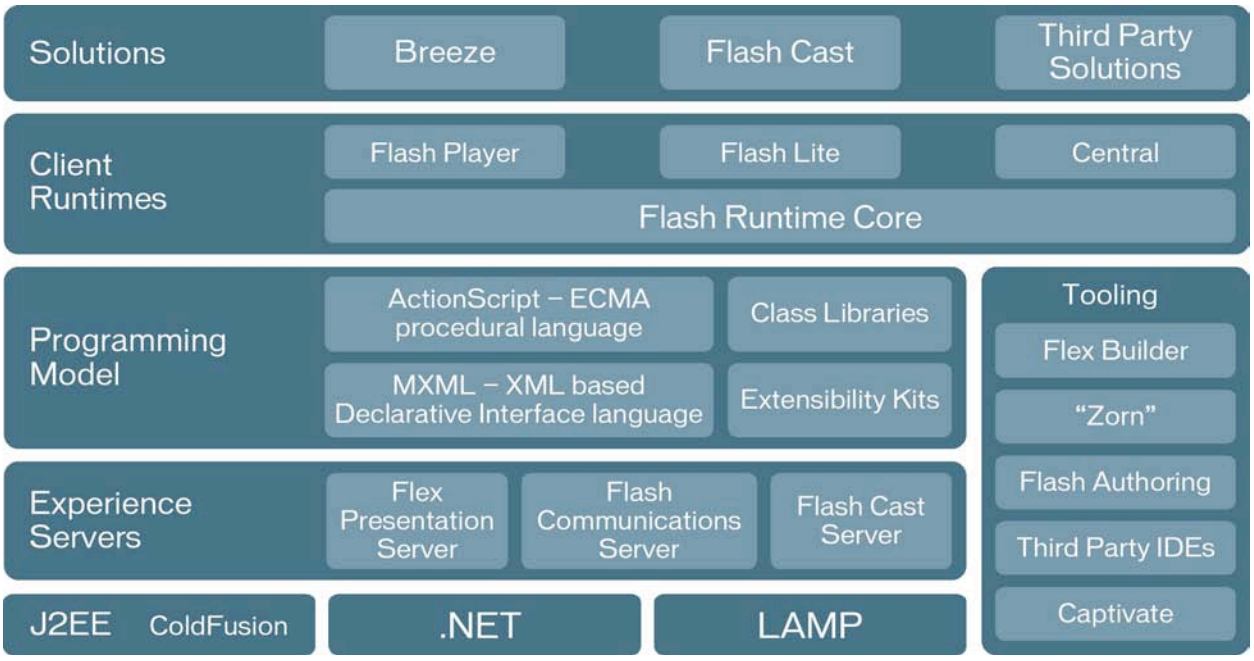

Figure 3: The Flash Platform has a layered architecture that encompasses the key elements shown here.

For more information on the Flash Platform, see the Macromedia white paper entitled "Delivering Enterprise Applications, Content, and Communications with the Flash® Platform."

Authentication

Due to the increasing pressures to comply with a range of industry regulations and the fact that a growing number of partners, contractors, and customers have access to corporate networks, enterprises are investing significant amounts in authentication and authorization services. These include single sign-on, VPN integration, specialized hardware (for example, smart cards), PKI, RSA SecurID®, or other physical tokens. At the same time, industry-specific requirements are mandating organizations to deploy authentication solutions. For example, both federal agencies and financial services organizations are required to utilize two-factor authentication measures to secure electronic transactions. Similarly, pharmaceuticals and health care organizations are facing tremendous pressure to protect the privacy of individuals through regulations such as HIPAA.

Fortunately, organizations that use the Flash Platform can leverage their existing infrastructure and security investments to address these requirements. Flex presentation server sits on top of a Java server and integrates with standard protocols for authentication, such as LDAP and other directory services. On the client side, the Flash client runtime takes advantage of the common security technologies available in web technologies, such as the transparent authentication handling by browsers.

Solution Example: Macromedia® Breeze™

Macromedia Breeze, a rich web communication solution that delivers high-impact online communications that can be accessed instantly through Flash Player, is built on the Flash Platform. Organizations can securely deliver data, voice, and video between Breeze applications and users using 128-bit Secure Socket Layer (SSL) encryption. In addition, Breeze enables integration with an organization's existing user management system, such as LDAP, so organizations can manage users and groups from a single location. Finally, Breeze Single Sign-On supports direct integration of corporate authentication systems, such as eTrust™ SiteMinder® from Computer Associates. This provides a seamless experience for users by eliminating the need for multiple user names and password prompts.

During an application penetration assessment conducted by Symantec Professional Services, Symantec found Breeze to be designed and implemented with security best practices in mind and observed that the Breeze security model offers integrated protection for the application data and environment. Specifically, the assessment showed that Breeze 5 prevents unauthenticated and unauthorized users from gaining access to Breeze assets.

Access Control

In addition to authentication, access control is increasingly being used to determine who has access to which content and applications within a corporate network. While access control requirements vary by application, the Flash Platform incorporates a number of features that help organizations address these needs. Some of these access control features come pre-set, and in some cases, administrators or users can customize them to their needs.

Server-Side Access Controls

Through the Flex presentation server, the Flash Platform offers access control to server-side data by utilizing existing access controls on the host servers. In addition, administrators can employ a *white list* to control access to all data. By using a sophisticated permissions model that normalizes data access requests, Flex can prevent character decoding and interpretation.

@stake, Inc., a leading digital security company, independently tested the Flex presentation server white list feature against simulated attacks and validated the server's ability to handle the most common Internet attacks appropriately. In fact, in its Macromedia Flex Product Briefing, @stake found that the "robust input validation [of Flex] proved to be powerful in diminishing the ability of a malicious attacker to obtain confidential information or disrupt Flex application services."

Client-Side Access Controls

Much like the model employed for Java and JavaScript, Flash Player runs content inside a virtual machine that implements a security sandbox. Within this sandbox, all Flash Player resources (applications, data, network URLs, and so on) are essentially isolated from the rest of the computing environment, as well as other sandbox instances. This approach provides an advantage over traditional web-enabled applications, such as ActiveX solutions, which often have complete access to the operating system environment. While Flash Player applications may interact freely with resources within the same sandbox, the Flash Player sandbox prevents unauthorized access to the operating system environment as well as other local instances of Flash Player.

Solution Example: FlashCast

The sandbox approach is used to support mobile applications such as Macromedia® FlashCast™ software. Similar to Flash Player, the FlashCast client that resides on mobile devices communicates with the FlashCast server for content updates, and runs content and manages resources—such as local storage—inside a sandbox. This sandbox approach enables organizations to communicate through multiple channels while minimizing security risks.

Unauthorized Access to Host System Resources

Unauthorized access to host system resources includes gaining control of applications, devices, or resources attached to the system for the purposes of disabling, denying, or redirecting access to those resources, for example, through buffer overruns or denial-of-service (DoS) attacks. Flash Player allows only limited access to specific resources. For example, Flash Player does not allow content to allocate its own memory, modify operating system settings, or make changes to the system registry. Unlike other client-side technologies, Flash Player contains a controlled set of objects and operations that are predominantly exclusive constructs within the Flash execution environment. Because the system functionality that Flash Player can access is limited, the risk of creating content that gains unauthorized access to the host system or resources attached to it is minimized.

By monitoring its usage of key system resources, such as disk space and system memory, Flash Player limits the potential of DoS attacks. Flash Player sets initial default limits of 100K for each domain to conserve disk space. If needed, Flash Player or the content it runs will proactively prompt the user to increase disk space allotment. However, the disk space limit is maintained until the user grants permission for an increase to the allotment of a particular domain.

Flash Player runtime provides well-defined secure interfaces to other web applications and content. The inherent client runtime design prevents development of malicious Flash applications that could take control of applications that are not based on the Flash architecture. While Flash applications can communicate with each other, the sandbox security model ensures that content originating from different domains is segregated into logical sandboxes. Applications and content can communicate freely within the sandbox, and communication across the perimeter of the sandbox is securely guarded. This includes scenarios in which multiple Flash applications are executing within a single instance of Flash Player and in which communication is attempted between two discrete instances of Flash Player.

Unauthorized Access to Data

Unauthorized access to data refers to data on local disks, networked disks, or web servers that are communicated over the network or stored in memory by an application or process (for example, password lists, address books, privileged documents, and application code).

An ActionScript program in Flash Player cannot write, modify, or delete any files on the client machine other than shared objects (small, Flash-specific files), and it can only access shared objects on a per-domain basis. Internet-based Flash applications cannot read any other local files, or any sensitive or private data. In fact, no ActionScript methods available to Flash applications can create, modify, or delete directories or files directly.

In order for web-based Flash Player content to access server data, the domain serving the Flash Player content must get explicit permission from the domain hosting the requested data (AKA the provider domain). Without permission, the load will fail. These permissions are specified by a policy file located on the server of the provider domain. This file enables access control by explicitly listing the domains that have permission to access data on that server.

Unauthorized Access to Private User Information

Personal and financial data—as well as information about the user's security settings for Flash Player—often resides on a user's machine, and users are rightly concerned about others accessing this information. However, users should be aware that Flash Player does not collect information about them.

Users have control over the Flash Player behavior when encountering decisions concerning privacy. Through the Flash Player Settings user interface and Settings Manager, users can fine-tune the following settings related to privacy and security:

- Local storage of data using the local shared objects mechanism
- Access to cameras and microphones connected to the system
- Notification of updates to Flash Player

In an enterprise environment, network administrators can control settings for Flash Player centrally to ensure that all clients conform to the corporate security policy.

In addition to the fundamental protections provided by the sandbox and virtual machine, the Flash Player client also provides *stakeholders* (those who own or administer a resource) with flexible, easy-to-use controls to permit (or limit) access to sensitive resources such as network files and databases. The Flash Player security model is organized in a way that enables enterprises to delegate control of permissions to the appropriate stakeholder (see Figure 4). This model also supports the distributed architectures that are commonly used for applications built on the Flash Platform.

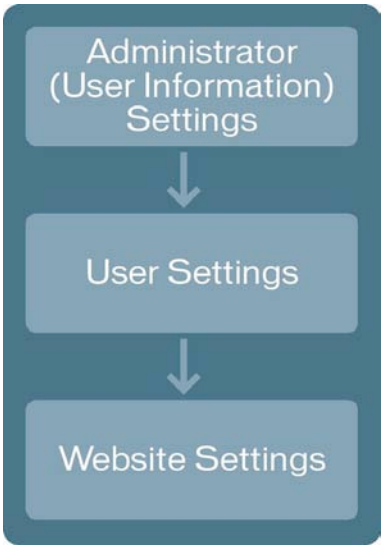

Figure 4: Security controls for Flash Player are organized hierarchically.

Malicious Code

All organizations face the potential for malicious code infection that can spread quickly throughout the corporate network. For example, Internet users could download what appears to be a legitimate program that in reality carries a threat such as a Trojan Horse program, which could expose the network to hackers. Or code authorizing remote access to a network can reside unnoticed in browser cookies or Web applets.

The Sandbox Approach: Protection Against Malicious Code and Activity

As discussed previously, because of the sandbox security approach on the client side and the use of Java on the server side, the Flash Platform uses in-place security tools to maintain resistance to malicious code, such as viruses, Trojan Horse programs, back door worms, and spyware. In addition, the design of Flash Player includes architectural characteristics that minimize malicious code threats compared to ActiveX or JavaScript solutions. Because all Flash Player resources are isolated from the rest of the computing environment—as well as other sandbox instances—through the sandbox approach, the host system is protected against malicious activity and potentially harmful programs and content. In fact, in a memorandum from the Joint Chiefs of Staff regarding policy guidance for the use of mobile code technologies in the Department of Defense (DoD) information systems, Flash Player is listed under category 3, the most secure of the three categories.

Minimized SQL Injection and Cross-Scripting Vulnerabilities

Solutions that use runtime interpreted string-based languages—such as JavaScript and DHTML—are especially susceptible to SQL injection and cross-site scripting, which both are listed among the top 10 vulnerabilities on the Open Web Application Security Project site (www.owasp.org). In contrast, Flash content is delivered as a series of instructions in binary format to Flash Player over web protocols in the SWF file format. The SWF files themselves are typically hosted on a server and then downloaded to, and displayed on, the client computer when requested. Because Flash Player is binary and compiled, it inherently minimizes these threats compared to string-based language solutions that may leave back-end data vulnerable and unprotected.

Solution Example: Macromedia Breeze

Typically, applications access databases through dynamically generated SQL statements, because these statements are fairly easy to implement and provide for looser coordination with the database. However, it is difficult to produce dynamically generated SQL statements that are resistant to SQL injection. In addition, dynamic statements often require broad access permissions to database objects. Breeze software uses prepared statements and stored procedures for calls to the database. Prepared statements protect against SQL injection, while stored procedures allow the database to be more tightly locked down.

During the application penetration assessment conducted by Symantec Professional Services mentioned previously, Symantec found that the implementation of stored procedures in Breeze software prevented attempts to compromise application data through the use of SQL injection and manipulation attacks.

Data Transport

Clearly, the secure transport of data between Flash and Flex hosts and applications is critical to ensuring the integrity of the data, as well as making sure others do not use that data for malicious purposes.

Standards Compliance

Both Flash Player and the Flex product line use standards-based protocols for data transport. Flash Player knows whether its data was obtained over a secure HTTPS (HTTP over Secure Sockets Layer) connection and records that fact using separate sandboxes. Data loaded from HTTPS sites is subsequently treated differently than data from HTTP or other, less secure sources. This client data segmentation is a natural extension of the most common PKI models, which use x509 certificates to identify clients and servers. Cryptographic standards such as x509 certificates are implemented by the browsers with which Flash Player interoperates. On the server side, these standards are implemented by the hosting environment. By using XML and SOAP standards for data transport, the Flex product line benefits from common security technologies such as HTTPS, which is supported for all operations.

Wireless Security

As the corporate network extends to provide access to a variety of constituents—such as contractors, partners, customers, and telecommuters—organizations must protect an increasing number of remote users. Without effective wireless security, not only is the data in transit vulnerable to access and manipulation, but the enterprise network itself is vulnerable to Internet threats and malicious code that can be introduced through wireless devices. By using SSL, native encryption, and the security on the operating system, Flash Player and the Flex product line minimize wireless security concerns.

Since Flash applications running within a browser use the browser for almost all communication with the server, they can take advantage of the browser's built-in SSL support for encryption. In addition, the actual bytes of a Macromedia Flash application can be encrypted while they are being loaded into the browser. By playing a Flash application within an SSL-enabled browser through an HTTPS connection with the server, organizations and users can ensure that the communication between Flash Player and the server is encrypted and secure.

Ease of Integration with SSL Accelerators and Load Balancers

Integration with SSL accelerators and standard load balancers is simple. For example, because Flex presentation server handles requests that are initially received by a web server, the Flex server does not need to know what protocol is being used. To switch from HTTP to HTTPS, the server administrator simply modifies the web server as he or she would have done without the Flex server installed.

Support for Encrypted Tunneling

Applications built with Flash Media Server use the Real-time Messaging Protocol (RTMP) for high-performance transmission of audio, video, and data messages in a single data channel between the client and the server. While RTMP does not include security-specific features, Flash communications applications can perform secure transactions and secure authentication through an SSL-enabled web server. When running within a browser, Flash Player can use secure encrypted HTTPS tunneling to communicate through RTMP. This tunneling support provides users behind a typical corporate firewall with a transparent experience while ensuring secure data transport.

Solution Example: Speedera Flash Video Streaming Service

Macromedia partner Speedera provides secure Flash Video over SSL through Flash Media Server. Users visit a content provider's site and are authenticated through a password. A hash key is then generated and the user is transparently re-directed to the Speedera server after verification. With secure Flash Video delivery, content can be played only on the intended website; it cannot be posted to other sites. In addition, the streaming URL cannot be mass-mailed to users who have not been authorized to use it.

Conclusion

With the Flash Platform, organizations can develop, deploy, and distribute with confidence RIAs, enterprise and mobile applications, and communications to employees, partners, and customers. Flash Player and the Flex product line leverage an organization's existing security infrastructure (which means they are security independent) are based on existing accepted standards, and use secure technologies. By virtue of the way that the Flash Platform and the Flex product line integrate with existing authentication, access control, data transport, and malicious code prevention solutions, they do not adversely affect an organization's ability to meet security requirements. Just as importantly, this approach supports continued compliance security best practices and regulations such as the Sarbanes-Oxley Act of 2002 and HIPAA. And by leveraging an organization's existing security infrastructure, the Flash Platform enables the successful deployment of secure applications without further investments.

According to an independent security assessment by @stake, Macromedia has developed a strong information protection model against client-side threats. "[The Flex] architecture mitigates many common client-side attacks such as cross-site scripting, denial-of-service [attacks], SQL injection, man-in-the-middle [attacks], and session hijacking." In addition, server-side security is maintained by leveraging J2EE security to mitigate common attacks against infrastructure components, such as buffer overflows, heap corruption, and cross-site scripting.

For More Information

For more information about the Flash Platform, call a sales representative at 1-888-649-2990 (US and Canada) or find an international sales line at **www.macromedia.com/international/buy/numbers.html**. To purchase online, visit **www.macromedia.com/store**. Or, use any of the following links:

- For more information about the Flash Platform, visit **www.macromedia.com/platform**
- For more information about Flash Player, visit **www.macromedia.com/software/flashplayer/**
- For more information about the Flash authoring tool, visit **www.macromedia.com/software/flash/**
- For more information about Flex Builder, visit **www.macromedia.com/software/flex/flexbuilder/**
- For more information about the Flex presentation server, visit **www.macromedia.com/software/flex/**
- For more information about Flash Media Server, visit **www.macromedia.com/software/flashcom/**
- For more information about Flash Video Streaming Services, visit **www.macromedia.com/software/flashcom/fvss/**
- For more information about Breeze, visit **www.macromedia.com/software/breeze/**
- For more information about Macromedia Security, visit **www.macromedia.com/resources/security**

References

Defense in Depth: Information Assurance and Computer Network Defense (CJCSM 6510.01), Joint Chiefs of Staff, August 2004 (www.dtic.mi/cjcs_directives)

Macromedia Breeze 5 Security Assessment, Symantec, July 2005 (www.macromedia.com/support/breeze/licensed_docs/macromedia-cfd-breeze5.pdf)

Macromedia Flash Player 8 Security, Macromedia, August 2005 (www.macromedia.com/deent/plashplayer/articles/flash_player_8_security.pdf)

Macromedia Flex Product Briefing, @Stake, August 2004 (www.macromedia.com/devnet/flex/articles/flex_security_wp/flex_security_wp)

OWASP Top Ten Most Critical Web Application Security Vulnerabilities, The Open Web Application Security Project (www.owasp.org/documentation/topten.html)

Policy Guidance for use of Mobile Code Technologies in Department of Defense (DoD) Information Systems Memorandum, U.S. Department of Defense, November 7, 2000 (www.dod.mil/nii/org/cio/doc/mobile-code11-7-00.html)