

# DELIVERING SECURITY OVER NATIVE FULL-DUPLEX WEB CONNECTIONS

SECURITY STRATEGY AND FEATURES OVERVIEW

# DELIVERING SECURITY OVER NATIVE FULL-DUPLEX WEB CONNECTIONS

## SECURITY STRATEGY AND FEATURES OVERVIEW

Kaazing WebSocket Gateway is a high performance platform that enables full-duplex Web communication from a browser to any TCP-based back-end services such as JMS or AMQP-based message brokers, XMPP chat servers and custom socket servers. It both adheres to and enhances the WebSocket specification of the HTML5 standard. As a native full-duplex connection, it delivers significant advantages over existing methods - long polling, hanging gets, Comet, etc. - which can only emulate full-duplex interaction.

Deploying applications over the Web—especially in uncontrolled web environments—presents an additional layer of challenge and complexity when it comes to security. For Kaazing and its customers, security is a central concern. Kaazing WebSocket Gateway offers enterprise-strength security that keeps your users and information safe over the Web. It protects your data, and lets you authenticate that users are who they say they are, and that they take only authorized actions.

Some of the role security plays may be obvious, such as encrypting traffic. But much of the impact of security—on development and in production—is more subtle. While a strong security policy protects data from threats, unless it's designed elegantly, it may negatively impact the user experience or impede the development process.

This document provides an overview of the security features that are built into Kaazing WebSocket Gateway. Furthermore, it highlights our design principles, and the value in the approach we've taken.

Specifically, it describes:

- An overview of the security strategy on which the Kaazing WebSocket Gateway is built.
- Highlights of important features within Kaazing WebSocket Gateway that enhance the user experience, streamline development, and strengthen security.
- Best practices for Kaazing WebSocket Gateway security implementation.

## SECURITY STRATEGY OVERVIEW

Security within the WebSocket standard is simple and certain, as long as the WebSocket solution you use implements it, since it's not enabled by default. The WebSocket standard takes care of core security by providing for unencrypted and encrypted transport, and by defining WebSocket as a frame within which all existing security protocols can operate. However, because WebSocket is a standard and not a development environment, the inherent security features are somewhat limited.

Often, security features have been limited at a high cost – one that obstructs the creation of robust, full duplex web applications. Developers are often faced with the difficult design and coding challenges, trying to figure out how to work within or around limitations without frustrating their users with awkward and time consuming processes.

Kaazing WebSocket Gateway was designed to solve those challenges. Not only does it keep the traffic that flows through it safe and protected, it also ensures that the security features do not hinder the development process not get in the way of a positive user experience.

### SAFE FOR USERS. EASY FOR DEVELOPERS.

While users demand a safe and secure interaction, they equally demand that the experience be delivered as transparently as possible. This means the authentication process needs to be fast and simple. Kaazing WebSocket Gateway streamlines and enhances the authentication and authorization process for users by employing features such as credential caching and Single Sign-On. Additional features are described in more detail in this document.

### ENHANCE SECURITY

The WebSocket standard is sufficient to secure traffic (if its security features are used). It supports whatever security is in place. But securing the flow is only part of the real security challenges you face. That traffic has to be able to seamlessly traverse proxies and firewalls. Issues of authentication delay and obstruct the web experience

Furthermore, Kaazing WebSocket Gateway provides advanced capabilities at every critical, application-related area of security. Its security extends from protocol validation to encrypted cookie and token creation, from HTTP Authentication to Kerberos support.

For developers, the key is to leverage existing knowledge and skill set to minimize additional time and resources required by security features implementation. Kaazing WebSocket Gateway comes with robust and familiar APIs that support multiple platforms running under multiple protocols and a wealth of client libraries to facilitate smooth development process. The following are the pillars of our design goal as relates to developers:

## ENABLE BROAD AND FOCUSED REACH

*Objective Make WebSocket secure across protocols and platforms*

A core part of Kaazing WebSocket Gateway is its reach across multiple operating environments. Kaazing WebSocket Gateway works with major browsers, across multiple protocols, and multiple client platforms. For browsers that don't yet support WebSocket, we provide full emulation that can perform near the level of native support.

We currently support the following protocols: HTML5, JMS, AMQP and XMPP and the following client platforms: JavaScript, Microsoft Silverlight, Adobe Flash, Microsoft .NET.

## TOUCH EVERY BASE

*Objective Create a security structure that addresses challenges*

Security presents challenges at many points for both user and developer. Kaazing WebSocket Gateway addresses them all, from ensuring connections through proxies and firewalls to cross origin resource sharing, and much more.

## MAKE IT EASY TO IMPLEMENT

*Objective Make implementing security simple and efficient.*

For developers, Kaazing WebSocket Gateway facilitates application development by providing client-side APIs for multiple protocols and platforms such as AMQP for Microsoft Silverlight, Adobe Flash or JavaScript—and complete documentation for each. For certain protocols, server-side APIs allow for custom behavior, such as fine-grained access control for specific JMS topics and queues.

## SECURITY FEATURES

Here are highlights of some of the security capabilities delivered with Kaazing WebSocket Gateway.

### CERTIFICATE SHARING

WebSocket Secure (WSS) is WebSocket communication running on top of an encrypted TCP connection. On an encrypted connection, Kaazing WebSocket Gateway leverages the same certificate used for HTTPS connections.

### CREDENTIALS INJECTION

Kaazing WebSocket Gateway's protocol awareness allows for the injection of credentials directly in the protocol communication. This aligns the target protocol with HTTP layer authentication. It becomes straightforward to fail-fast at the gateway after an unsuccessful login, without requiring the user to re-enter credentials.

### CROSS ORIGIN RESOURCE SHARING

Kaazing WebSocket Gateway provides for fully secure and compliant cross origin browsing, according to the Cross Origin Resource Sharing (CORS) standard. Each

gateway service is by default denied access to cross-origin content, until you explicitly allow it. Web applications that use Kaazing WebSocket Gateway can access content that originates from a different origin, one with a different scheme, host, or port.

### ENCRYPTED SESSION COOKIES

A further level of security is deployed through encrypted session cookies and encrypted tokens.

### HTTP AUTHENTICATION AND AUTHORIZATION

Kaazing WebSocket Gateway takes full advantage of existing HTTP Authentication and Authorization methods and technologies.

### INTEGRATED WITH THE APPLICATION

Kaazing WebSocket Gateway with the browser, presenting login dialogs that match your application's style, instead of a generic browser login.

### INTEGRATES WITH TRADITIONAL HTTP SECURITY

Currently, the standard's WebSocket handshake doesn't support authorization headers; in fact, any response other than a 101 status is not supported. Kaazing WebSocket Gateway provides a client library that allows you to simply integrate HTTP authorization and authentication. The handshake that upgrades the connection looks like an HTTP handshake. Cookies and authorization headers are fully supported.

### KERBEROS

Unique to Kaazing WebSocket Gateway is its support for SPNEGO-based Kerberos security across a WebSocket connection. Integrate Kerberos with your existing infrastructure to provide Single Sign-On capability over the Web.

### PERSISTENT USER AWARENESS

Information about the end user—for tracking and compliance issues—is retained even after a message flows through the Kaazing WebSocket Gateway. For example, JMS consumers can tell which end-user published a message.

### PROTOCOL VALIDATION

Kaazing WebSocket Gateway has built-in protocol awareness, allowing it to verify that bytes passing through Kaazing Gateway match the expected protocol wire format. This helps prevent buffer overrun attacks or other malicious attacks on the back-end server.

### SINGLE SIGN ON

A number of features—including encrypted session cookies, HTTP Authentication and Authorization, and Kerberos support—let Kaazing WebSocket Gateway deliver Single Sign-On capability for your subscribers and users.

### WIRE TRAFFIC ENCRYPTION

Kaazing WebSocket Gateway supports wire traffic encryption, protecting you and your

users' data across the public Internet.

## BEST PRACTICES

### ALWAYS ENCRYPT

We explicitly recommend you encrypt traffic between the Kaazing WebSocket Gateway and the browser. We strongly recommend you always encrypt

### PLACE GATEWAY WITHIN DMZ

Kaazing WebSocket Gateway is ideal to place in the DMZ to act as a front-line access point for sharing data on the Web from back-end systems and applications without exposing it to everyone on the Internet.

## SUMMARY

Security is central to our product design. We have invested tremendous amount of resources to ensure that our customers can deliver the most compelling yet secure user experience: we take security very seriously. But we take your experience as a developer, and the experience of the users you design for just as seriously. All of that is part of the security strategy and security functionality built into the Kaazing WebSocket Gateway. Development is simplified, time-to-secure slashed and resource usage reduced. Most importantly, the security capabilities provided by Kaazing WebSocket Gateway let you deliver a full-duplex connection over the Web without sacrificing performance, safety or the quality of your users' experience.

For more information about Kaazing or its solutions, please visit our website at <http://www.kaazing.com> or send an email to [sales@kaazing.com](mailto:sales@kaazing.com)