Telerik Platform

SECURITY

Table of Contents

Overview

At Telerik®, we realize how important security is for you and the users of your app. This is why our team approaches security as a fundamental component of the Telerik PlatformSM solution. Telerik takes extensive measures to protect our customers from threats by applying security controls across all layers of our solutions. Today, over one million developers from 450+ of the Fortune 500 companies rely on Telerik solutions to create compelling and secure experiences across web, mobile and desktop applications. In addition to our security practices, Telerik provides our enterprise customers with unlimited phone assistance, remote web assistance, access to all Telerik community forums and more. We have set an industry standard for global customer support. This document provides an overview of security measures taken by Telerik to protect Telerik Platform customers against cyber threats.

About Telerik Platform Solution

Telerik Platform solution enables developers to build, connect, test, deploy and analyze cross-platform mobile apps using any development approach (web, hybrid or native). It is built to help developers and designers streamline their app development tasks while enhancing app performance and usability:

- **Design:** Create interactive prototypes and intuitive UI
- **Build:** Rapidly create cross-platform mobile apps
- **Connect:** Securely leverage existing apps and data
- **Test:** Confirm apps run on every device
- **Deploy:** Publish to public or private app stores
- **Manage:** Centrally control access to your entire app library
- **Measure:** View crash reports and app usage

Taken together, our end-to-end, adaptive platform uniquely combines industry-leading UI tools with cloud services to simplify the entire software development lifecycle.

Static Code Analysis

Telerik utilizes Veracode's Binary Static Application Security Testing (SAST) procedure to carry out static code analysis. This analysis is carried out on a regular

Software Security

basis–minimum once a month, as well as on every major change to the functionality or the architecture, across server-side framework and client-side code to highlight possible vulnerabilities within “static” (non-running) source code. We use techniques such as Taint Analysis and Data Flow Analysis.

Dynamic Code Analysis and Vulnerability Testing

Telerik leverages Veracode’s Dynamic Analysis to carry out dynamic code analysis and vulnerability testing on a regular basis and upon every major change to the application functionality or architecture. Dynamic Application Security Testing identifies architectural weaknesses and vulnerabilities in our running web applications to help prevent others from finding and exploiting them.

Quality Assurance Process

Telerik Platform solution is tested throughout the entire software development lifecycle. Every code commit triggers execution of a set of integration and UI functional tests in several environments. There is daily performance benchmarking, observing a number of metrics for key operations.

Our quality assurance organization implements the following to assess and improve the quality of the software:

- Daily standup meetings
- Iteration planning
- Milestone planning
- Milestone overviews
- Retrospective meetings
- Formal and informal code reviews
- Formal and informal software architecture reviews
- Automation testing
- Continuous delivery
- Pair programming

Standards Compliance

Telerik complies with the following standards:

OWASP Top 10

The OWASP Top 10 represents a broad consensus on the most critical web application security flaws. The errors on this list occur frequently in web applications, and are both easy to find and exploit. To prevent such flaws in our solutions, the OWASP Top 10 are considered during design, development and deployment of Telerik Platform. The latest OWASP Top 10 (2013) web application security risks includes the vulnerabilities below:

1. Injection: Injection flaws, such as SQL, OS and LDAP injection occur when untrusted data is sent to an interpreter as part of a command or query. The attacker's hostile data can trick the interpreter into executing unintended commands or accessing data without proper authorization.

2. Broken Authentication and Session Management: Application functions related to authentication and session management are often not implemented correctly, allowing attackers to compromise passwords, keys or session tokens, or to exploit other implementation flaws to assume other user identities.

3. Cross-Site Scripting (XSS): XSS flaws occur whenever an application takes untrusted data and sends it to a web browser without proper validation or escaping. XSS allows attackers to execute scripts in the victim's browser which can hijack user sessions, deface websites or redirect the user to malicious sites.

4. Insecure Direct Object References: A direct object reference occurs when a developer exposes a reference to an internal implementation object, such as a file, directory or database key. Without an access control check or other protection, attackers can manipulate these references to access unauthorized data.

5. Security Misconfiguration: Good security requires having a secure configuration defined and deployed for the application, frameworks, application server, web server, database server and platform. Secure settings should be defined, implemented and maintained, as defaults are often insecure. Additionally, software should be kept up to date.

6. Sensitive Data Exposure: Many web applications do not properly protect sensitive data, such as credit cards, tax IDs and authentication credentials. Attackers may steal or modify such weakly protected data to conduct credit card fraud, identity theft or other crimes. Sensitive data deserves extra protection such as encryption at rest or in transit, as well as special precautions when exchanged with the browser.

7. Missing Function Level Access Control: Most web applications verify function level access rights before making that functionality visible in the UI. However, applications are designed to perform the same access control checks on the server when each function is accessed. If requests are not verified, attackers will be able to forge requests to gain unauthorized access to functionality.

8. Cross-Site Request Forgery (CSRF): A CSRF attack triggers a logged-on victim's browser to send a forged HTTP request, including the victim's session cookie and any other automatically included authentication information to the vulnerable web application. The attacker can then generate requests to the vulnerable application, which are subsequently recognized as legitimate requests from the victim.

9. Using Known Vulnerable Components: Components, such as libraries, frameworks and other software modules almost always run with full privileges. If a vulnerable component is exploited, such an attack can facilitate serious data loss or server takeover. Applications using components with known vulnerabilities may undermine application defenses and enable a range of possible attacks and negative impact.

10. Unvalidated Redirects and Forwards: Web applications frequently redirect and forward users to other pages and websites, using untrusted data to determine the destination pages. Without proper validation, attackers can redirect victims to phishing or malware sites or use forwards to access unauthorized pages.

CME/SANS 25

The 2011 CWE/SANS Top 25 Most Dangerous Programming Errors is a list of the most significant errors that can lead to serious software vulnerabilities. The errors on this list occur frequently, and are often easy to find and easy to exploit. This list is considered during development and peer review process to maximize the security of our offerings.

Veracode Level 4 Rating

The Veracode Level (VL) achieved by an application is determined by type of testing performed on the application, and the severity and types of flaws detected. A minimum security score is also required for each level. Veracode conducts periodic audits of the Telerik Platform solution using automated static, automated dynamic and/or manual security analysis techniques to identify security flaws identified in the application. Discovered weaknesses are categorized based on FIRST's Common Vulnerability Scoring System (CVSS) for severity and ease of exploitability.

Safe Harbor Certification

In light of the international nature of our business, Telerik privacy practices are self-certified to the Safe Harbor Program codified by the U.S. Department of Commerce and the European Commission. Telerik complies with the US-EU Safe Harbor Framework and the U.S.-Swiss Safe Harbor Framework as set forth by the U.S. Department of Commerce regarding the collection, use and retention of personal information from European Union member countries and Switzerland. Telerik has certified that it adheres to the Safe Harbor Privacy Principles of notice, choice, onward transfer, security, data integrity, access and enforcement. To learn more about the Safe Harbor program, and to view our certification page, please visit www.export.gov/safeharbor.

Security Policy

Processes for Certification

Telerik has initiated the Learning and Development Policy to certify the development and quality assurance staff. The aim of Telerik Learning and Development is to ensure that all Telerik employees are given the necessary resources to enhance the knowledge, skills and attitudes required to carry out their jobs effectively. In order to ensure that training activities support both the individual and organization's objectives and are cost-effective, all training activities are planned, monitored and approved by the Learning and Development team.

Developer Certification: Software developers are certified with Microsoft Certification program. Each software engineer is encouraged to take at least one MS certification exam during the first year at Telerik. For the period of 2009-2014, 166 of our software developers have obtained at least one Microsoft Certification.

Transport Layer Security

Telerik Platform solution is equipped with a robust set of security measures to encrypt and securely transport data. The Platform encrypts the communications using the 128-bit SSL protocol, including service calls made between customer apps and Telerik Platform services. Additionally, to ensure privacy protection and data security, Telerik supports user authentication using Active Directory integration (ADFS) and Oauth (with support for Google, Facebook, Yahoo and LiveID).

SANS, CERT AND NIST

Additionally, Telerik dedicated security staff monitors security bulletins from SANS, CERT and NIST on a periodic basis to stay up to date with the latest security threats. SANS, CERT and NIST provide cyber threat and Internet security monitor and alert resources to inform software developers about the latest security threats.

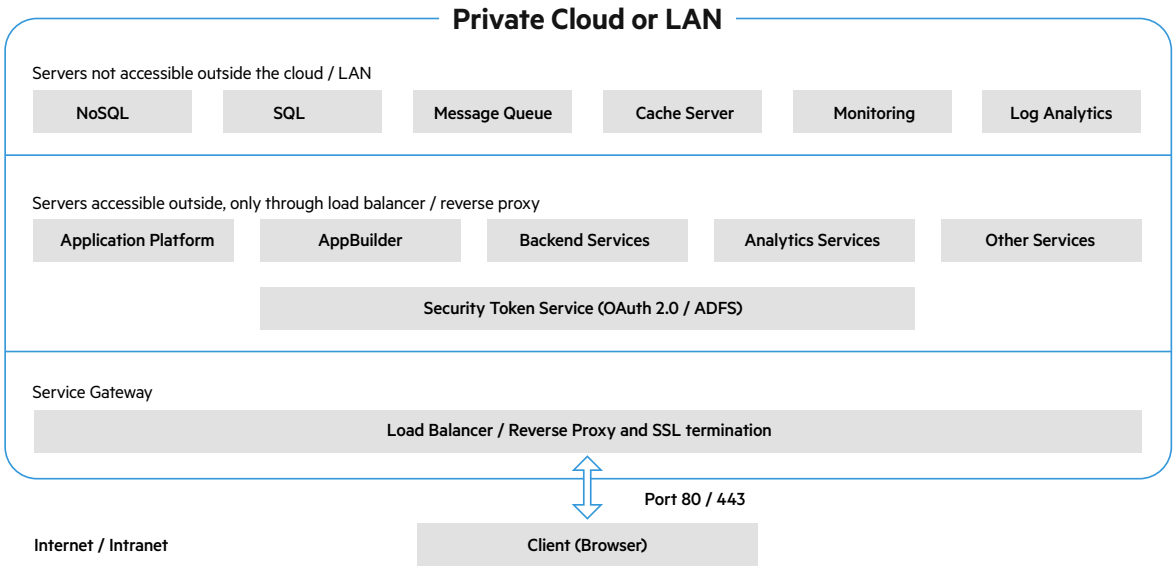
Infrastructure Security

Telerik Platform Architecture and Infrastructure Security

The Telerik Platform architecture consists of multiple servers providing a variety of services, consumed either by client users or internally between services themselves. In such a configuration, there are two main security aspects: infrastructure (network) security and application (service) level security.

Infrastructure Security

The infrastructure security defines the isolation of logical groups (zones) of servers and services on the communication transport layer. Roughly speaking, the security on this layer is applied through TCP/IP routing and firewalls. The actual implementation depends on the underlying virtualization framework.



All communications outside the cloud are proxied through hardware or software load balancer/router and SSL termination appliance.

Application Level Security

Application-level security typically consists of two phases: authentication and authorization.

Authentication is the phase of determining the identity of a user. Telerik Platform solution integrates with third-party identity providers, such as Active Directory or social networks (Facebook, Google, Yahoo and LiveID). For social networks integration, Telerik Platform solution uses OAuth 2.0 protocol, while for Active Directory, ADFS (active directory federated services) is required.

Authorization is the phase in which an identified user is permitted or denied to perform a certain action or access a resource, based on the permissions set for that user. Typically there are predefined roles such as Administrators, Contributors, Visitors and so on, and user actions are allowed or denied based on the user's membership. Authorization rules are specific to every service and are enforced per service individually.

Below is an example of simplified flow for application-level security:

1. A client application or browser makes a request to Telerik Platform solution for the first time.
2. The request is redirected to the Security Token Service (STS) where the user chooses its identity provider.
3. The user authenticates with the chosen identity provider and then is directed back to the STS.
4. The STS issues a security token for the established identity and redirects the client to the original request.
5. Based on the security token a session between the client application and Telerik Platform solution is established.
6. The client is given access to resources based on its membership.

Customer Security

Integration with In-House Source Control Systems

We are offering the ability to integrate with an in-house source control server, which helps assure your source code will stay secure within your own network infrastructure. If you use our Telerik Platform Visual Studio Extension or Command Line Interface, you can keep your code in any type of source control server within a private or public infrastructure. If you use the web or Windows client, you can use any public Git server or on-premise Git server (for on-premise customers only).

Service-Level Agreement

In addition to advanced security features, Telerik Platform solution now comes with the additional protection of a Service-Level Agreement (SLA), guaranteeing a strong percentage of Backend Services API Servers uptime. Our Enterprise Edition subscribers can take advantage of this SLA. For more information about the SLA, please contact us sales@telerik.com.

Dedicated Customer Database

Our customers have the option of separating their database instance from the rest within the same database server cluster. This practice provides an additional layer of protection against breaches resulting from shared database usage in the public cloud. Additionally, it provides improved reliability and performance.

Content Delivery Network

Content Delivery Network (CDN) is the key enabling technology behind successful consumer-facing sites in verticals such as media and entertainment, software download delivery, gaming and ecommerce. Telerik Platform solution is integrated with a CDN to serve content to end-users with high availability and high performance.

SSO Support

Telerik Platform solution supports single sign-on (SSO) method for ensuring the tightest levels of security to help ensure that only authenticated users are able to access and interact with highly confidential enterprise data. The platform integrates seamlessly with Windows Authentication integration (ADFS) and OAuth (with support for many identity providers, like Google+ and Facebook). These capabilities can also be embedded in customer applications.

Conclusion

Telerik Platform solution provides a highly available, production-grade infrastructure that can scale based on customer needs. Our team takes security very seriously and approaches it as a fundamental component of the Platform. We take extensive measures to protect our customers from threats by applying security controls across all layers of our solutions.