OutSystems Platform security overview

OutSystems Platform has an extensive set of built-in security features for applications, IT, and end-users. Every application created with OutSystems Platform is secured over its entire lifecycle.

Application vulnerability prevention and detection

All applications built with the OutSystems Platform include a number of vulnerability prevention measures that are applied at different stages of the application development and deployment process.

Application design

During application design, developers set configuration attributes that direct the code generator and the deployment service to set up the way in which applications can be accessed as follows:

- HTTP/SSL encryption is per page and web service when data encryption is required.
- Windows Integrated Authentication uses operating system credentials to automatically login into a given page or application (available in on-premises installations of OutSystems Platform).
- Active Directory/LDAP authentication centralizes all end-user login information in a single Active Directory/LDAP server.
- Role-based access control restricts access to pages depending on specific application level roles.
- Network-based security is used when access needs to be restricted to a specific IP range.

Developers define application level permissions by using visual access control building blocks (called roles) to declare a set of capabilities under a given access restriction. These can, for example, aggregate access to every application page that involves changing a specific database table.

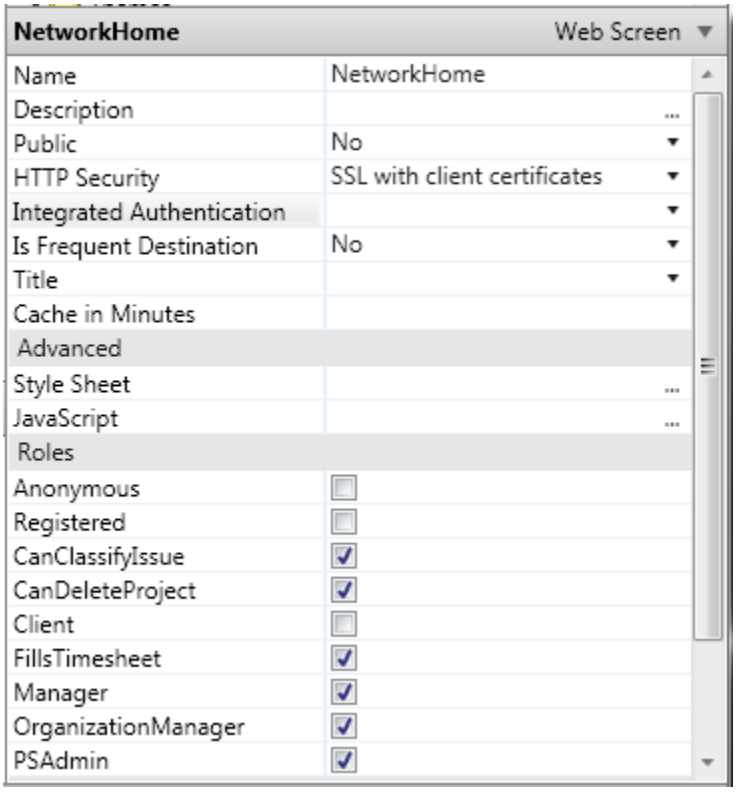
Developers can also create access control logic to implement flow control for users who are not authorized to access a specific resource. These requests will raise an exception that can be handled by an error message or by directing the users to a different area of the application.

Application validation

Before submitting any application to the code generator and the deployment service, there is a validation process with a number of security checks, including:

- A potential violation of data isolation warning will appear when defining queries to different databases
- Developer access control validation is in place to ensure individuals have permission to generate and deploy the application and if they're able to use any external components, APIs and data models

This example shows how OutSystems Platform enables teams to set user profiles access rights and enable authentication, encryption, or both by means of a graphical utility. See below:



Code generation, optimization and compilation

The OutSystems Platform generates, optimizes and compiles C# and Java code using secure code patterns, as well as introducing enhancements to the base framework. These features include:

- HTTPS support to prevent eavesdropping and session hijacking.
- Strong session identifier validation mechanisms leveraging those provided by the Java and .NET frameworks to prevent intrusion on existing sessions from multiple devices.
- Automatic escaping of the generated HTML and built-in functions to sanitize HTML when developers handcraft HTML code to prevent cross-site scripting prevention.
- An encrypted password option for database connections to securely create and manage database access.
- SQL parameterization and built-in functions to sanitize the strings that developers include in their queries to prevent SQL code injection.
- No late binding or runtime access to any pre-compiled code to prevent C# and Java code injection.
- Dedicated and isolated database connection pools per each pair of applications/databases to prevent cross-application and cross-database access.
- Total runtime isolation and containment with code generation patterns that ensure there is no way to exploit low-level process or thread configurations.
- Full exception handling (including encryption, authentication and authorization) in the generated code and logging for later audit (even when handling was not created during development) to prevent the exploitation of any vulnerability that arises from specific exception or error code in the responses provided to the browser.

Application deployment

The OutSystems Platform deployment engine configures Microsoft's Internet Information Services (IIS) security settings according to the most demanding application design and security best practices. These are:

- SSL certificates configured on a per-site, per-virtual-directory and per-page level

- Client-side SSL certificate management and configuration to enable stronger authentication of selected clients
- Windows Authentication configured on a per-site and per-virtual directory level
- Override of security and access control defaults to files placed in virtual directories to prevent “by default” vulnerabilities of IIS
- Deployment of applications across multiple farms in different network zones (in on-premises installations of OutSystems Platform) according to centrally managed configurations to ensure intranet functionality binaries are never installed in internet or extranet servers.
- Optional use of operating system credentials to execute the application processes.

Database network data encryption

OutSystems Platform is fully operational with Oracle’s network data encryption using the RC4 algorithm, which is the international standard for high-speed data encryption, up to 256-bit key length.

Generated code vulnerability scanning

To systematically ensure high-security standards for its generated applications, OutSystems leverages security assessment tools as part of its automated quality assurance process on every product release. Integration with HP Fortify Static Code Analyzer has been set up for automatic code vulnerability scans during regression testing. These tests, supported by an aggressive criteria for release acceptance fixes all critical, high and medium reported code vulnerabilities and ensures that the generated code is inherently secure.

As new code vulnerabilities are found in the generated code, a security patch is issued that permanently fixes them for all applications of all customers.

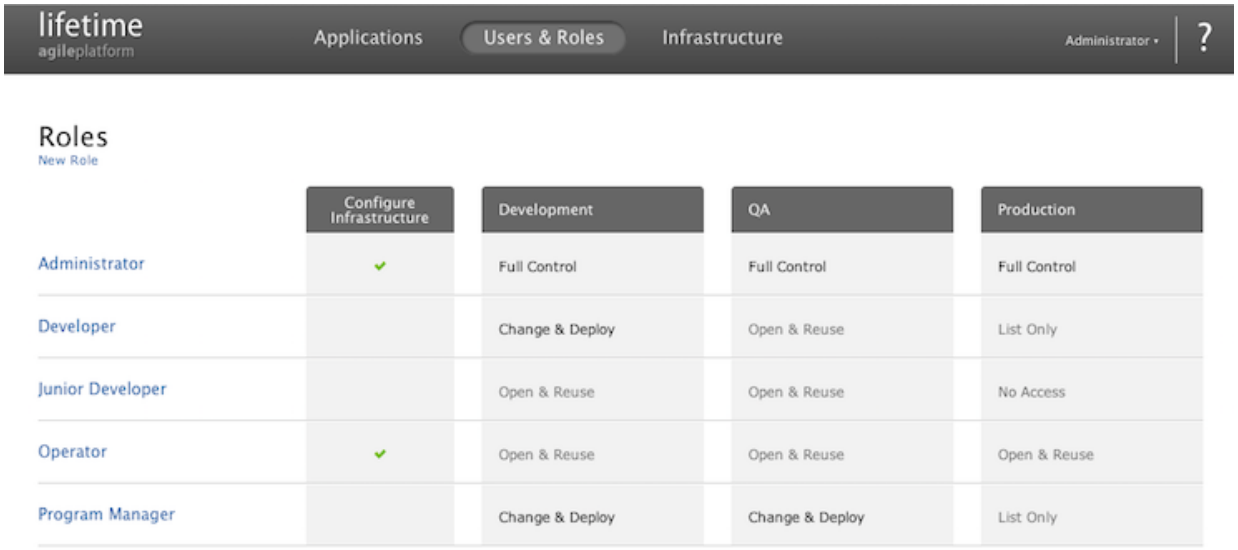
IT security management and auditing

OutSystems Platform includes access control management for all application resources, providing flexible permissions to define the access rights for any given resource. This helps organizations manage large teams with different profiles, as well as clearly separate the access to the platform's integration, assembly, deployment and change services in multiple development, QA and production environments. Moreover, it provides organizations with full access to the system audits required for IT-level SOX/ITIL controls and control deployment zones.

Role-based resource access control

IT team responsibilities are defined based on roles. For each role, it's possible to configure which applications the role can access and if the role is allowed to create and change them. Built-in access levels range from List Only (which tells IT users that the resources exist), to Full Control, allowing IT users to fully change, manage and deploy resources.

This example shows how to review the access levels of each role in the IT team:



Each built-in access level incrementally builds upon on the other when determining the development and management capabilities that are available to IT users with that role.

IT process auditing

Every activity performed by developers, application managers and system administrators is tracked in a system log for future audit. Events tracked include:

- Storing a new version of an application or component
- Deleting an application or component
- Deploying a new version
- Modifying user configurations

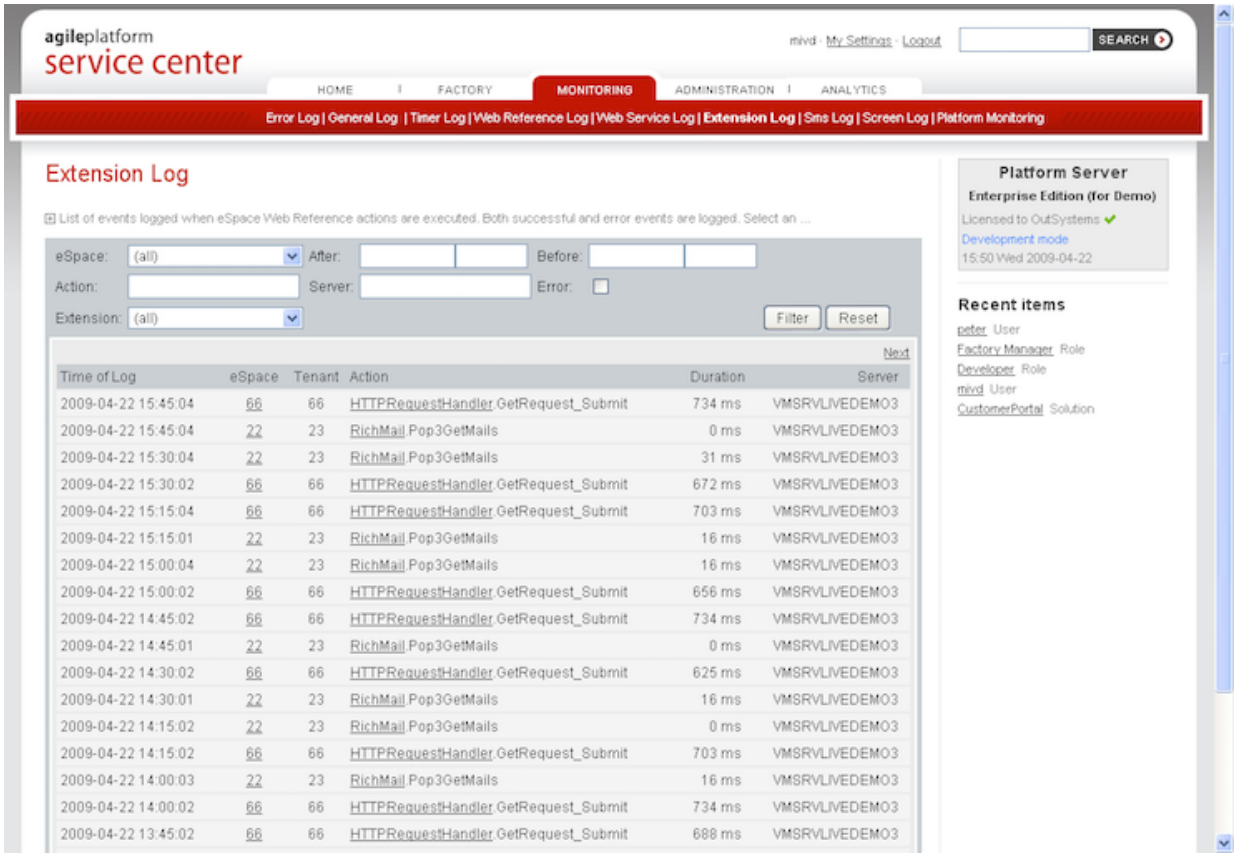
- Logging into the system

Furthermore, the system audits and version control subsystems of the OutSystems Platform allow auditors to identify when a modification to an application was applied, by which user, and even allows for the inspection of the exact content of that change using the OutSystems Service Studio visual difference and merge tool.

IT runtime auditing

OutSystems Platform logs all access to external systems performed through web services or custom integration logic, as well as all web service requests addressed to applications in OutSystems Platform. The logs keep a record of who made the request, the request's target, the method called, how long the request took, and the exact time of the request. This enables developers to efficiently and effectively track down any security issues that may arise.

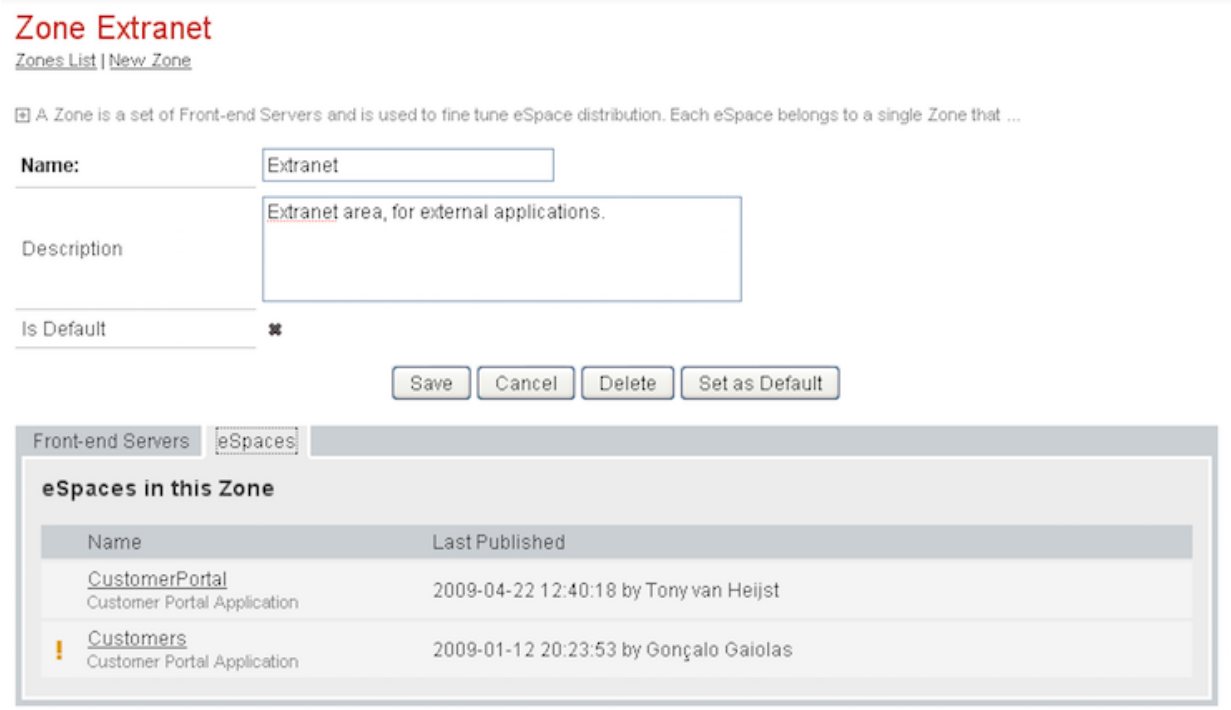
This example shows detailed logs of all calls to external systems:



Network zones management

With OutSystems Platform, it's possible to configure how front-end servers are spread across the various configured networks (internet, intranet, extranet) and define which applications are deployed to which clusters of the front-end servers. For example, internal applications can run in the internal network zone, and websites can run on a demilitarized zone.

This is an example of detailed configuration of front-end servers and eSpaces associated with a network zone:



End-user security management and auditing

Once users are registered to use an application, proper access control measures need to be set up to ensure that only authorized users are allowed to perform specific business functions.

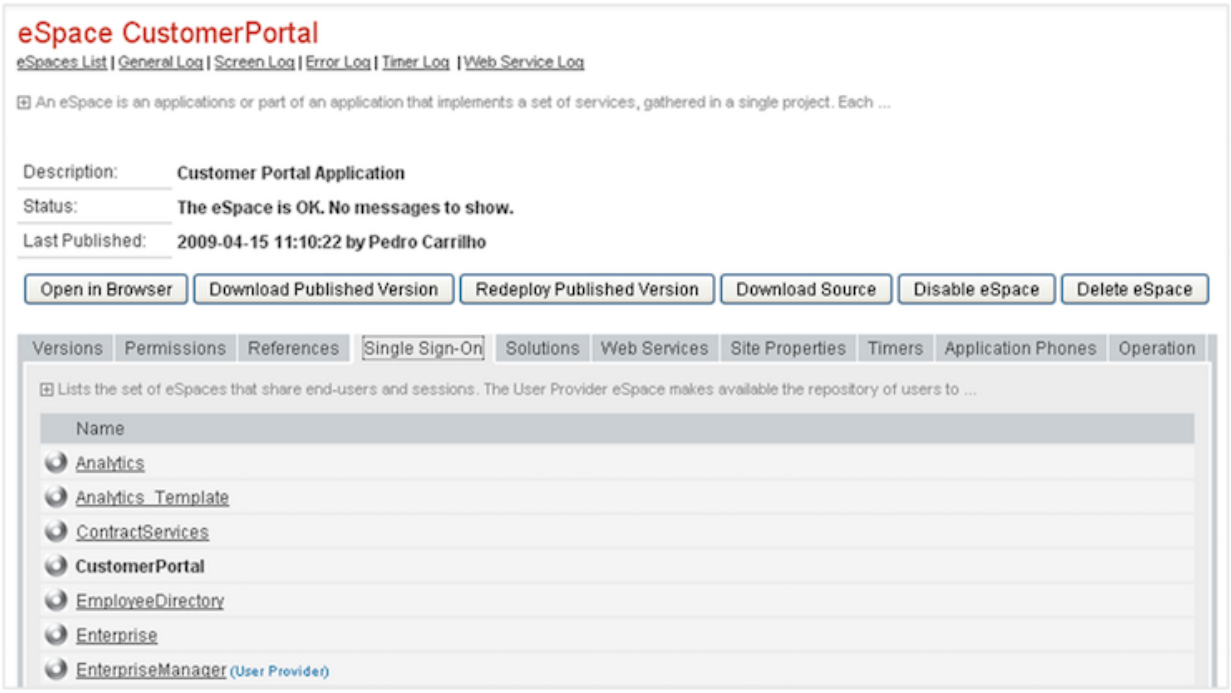
Role-based access control

Users can be provisioned and granted access to one or more roles. User management can be done from the back-office or through the applications using APIs that are available to developers. Application managers can use a metadata driven back-office to create and configure specific user roles. The definition of a user role is completely dynamic and independent of the application development phase.

Single sign-on

The OutSystems Platform single sign-on capability allows developers to unify logins across all the applications they choose. The user is then able to move seamlessly across applications without additional logins being required.

This example shows how end-user login can be unified across any number of eSpaces:



End-user access auditing

Every access to an application's screens is tracked in detail by default in the OutSystems Platform. These logs include the component and screen accessed, which user accessed it, when the access occurred, and exactly which node served the screen. This allows organizations to effectively track down any security issues that may arise.