PROTECTIMUS

Technical Overview

Version 1.0.2 RU
dated 10 July 2014

Contents

Solution Overview

Authentication based only on one factor – the password – cannot be considered reliable enough for systems with a high level of security requirements. Using several factors in the authentication system simultaneously significantly increases the system's level of protection against unauthorized access.

Many companies offer their solutions for implementing multifactor authentication, but their problem is that they are bureaucratic monsters with all the predictable consequences: poor communication and subpar service; cumbersome, inconvenient and inflexible products; unavailability of the current information and lack of means to obtain it promptly. Besides, these companies are monopolists; they bill their customers for huge amounts after completing the stage of negotiations with each client, without publically disclosing their prices.

Protectimus aims to offer the best solution in the sphere of two-factor authentication in terms of price and ease of use and ensure a high level of quality and system reliability.

The product developed will allow:

1. **Customers** to build more secure services based on the product in a simple and affordable way, regardless of the company's size
2. **Customers' users** to securely protect their accounts from unauthorized access

Protectimus offers a comprehensive solution that includes both an authentication system and a wide range of software and hardware tokens.

The authentication system can function both as a separate application installed on a customer's servers and as a cloud service providing a SaaS solution. Protectimus has already taken care of the stable and failure-free operation of its system.

Protectimus solves the problem of two-factor authentication at all levels. Each customer will be able to find a solution that best meets all their requirements.

System Features

The Protectimus platform supports a wide range of OS (from Linux, FreeBSD to any version of Windows).

The system supports the current and older versions of these popular browsers: Google Chrome, Mozilla Firefox, Internet Explorer.

All the system's components support the existing software development standards, as well as the OATH standards for OTP authentication, which makes it possible to use third-party manufacturers' or competitors' tokens in Protectimus.

Protectimus ensures the operation of several copies of the application that are not connected with each other in different geographic zones near a customer's users. Also, there is a possibility to have several nodes working for one large-scale customer with users located in various parts of the world.

Algorithms Used to Generate One-Time Passwords

The following algorithms are used to generate one-time passwords:

- HMAC - hash-based message authentication code: RFC2104
- HOTP - hash-based one-time password: RFC4226
- TOTP - time-based one-time password: RFC6238
- OCRA: OATH Challenge-Response Algorithms: RFC6287

They were developed by the Initiative For Open Authentication (OATH) group with the aim of standardizing authentication methods. These algorithms were thoroughly tested and proven reliable; they have become the standard in the field of two-factor authentication.

Technology Stack

#	Tools	Name (Version)
1	Java	7
2	Web/App Server	Tomcat 7.0
3	Framework	Spring 3.1.0, Apache Tapestry 5.3.7
4	GUI	Twitter Bootstrap, JQuery
5	ORM	Spring JDBC
6	Database	Postgres SQL 9.3
7	Building	Maven 3
8	High-performance, distributed memory object caching system	Memcached
9	Application Load Balancing and Content Caching	Nginx

Software Quality Attributes

In the process of software development, the following was used:

- Standard mechanisms and libraries
- Java Programming Style Guidelines (Java™ Coding Style Guide)
- DRY (Don’t Repeat Yourself) and DIE (Duplication Is Evil) principles
- Test Driven Development (TDD)

Solution Architecture

Protectimus is based on the best SOA, MVC, RESTful, and other practices. Let’s look at the overall solution architecture presented in Figure 1 below.

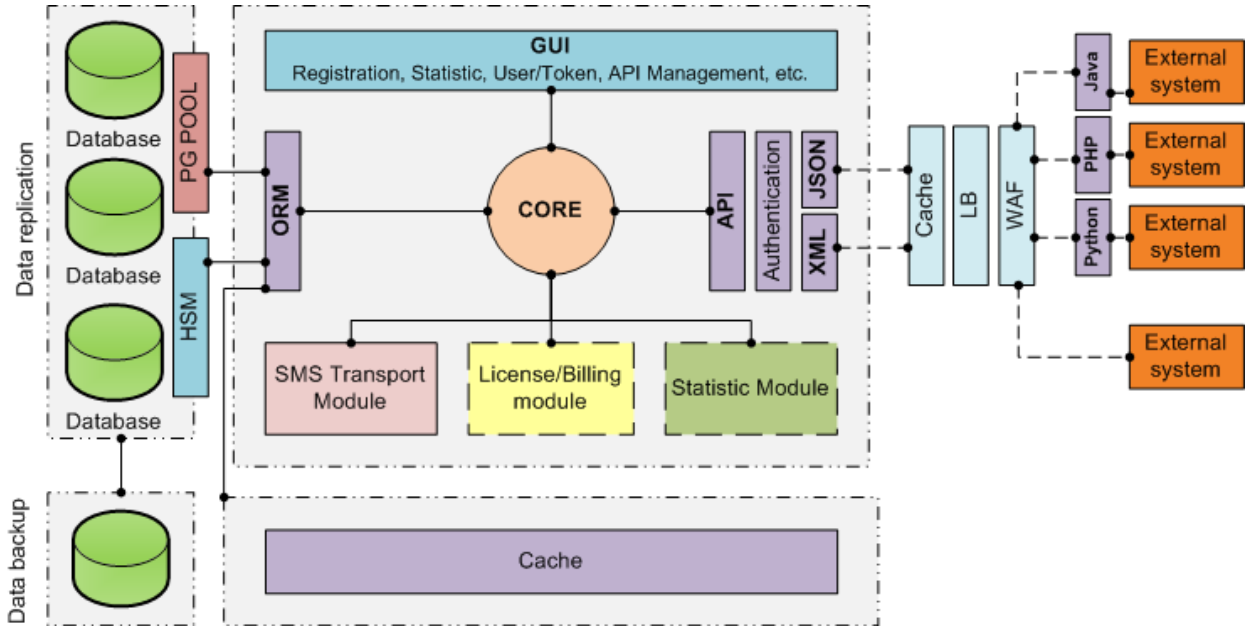


Figure 1. Overall Solution Architecture

To ensure reliable and uninterrupted operation, the system is deployed in a cluster of high-performance servers. The hardware Load Balancer is responsible for distributing and balancing the workload between the servers. The Monitoring System continuously monitors the state of the infrastructure and notifies Protectimus administrators of possible threats and emergencies.

The Hardware Security Module ensures secure storage of cryptographic information.

It is difficult to overestimate the value of the data stored in our system. We must prevent any data loss and provide a data recovery mechanism in the event of any possible failure or force majeure. That is why data backup and data replication are used.

Protectimus “holds” 10 backups: one for each day of the week, as well as backups that are two weeks, one month, and three months old, which allows "rolling back" in time far enough in the event of any data loss. Backups are stored on separate servers.

A common problem with all backups is that they always lag behind in time. In the event of a fatal failure of the primary server, restoring the system will only be

possible with some "rolling back" in time; solving this problem is the task of the replication system — replicating data changes from the database's main server on one or several dependent servers. The primary server is called the master server, and dependent servers are called replicas.
The data changes that occur on the master server are replicated on the replicas (but not vice versa). Therefore, data modification requests (INSERT, UPDATE, DELETE, etc.) are performed only on the master server, and data reading requests can be performed both on the replicas and the master server.
Replication is performed by means of binary logs kept on the master server. All requests resulting in database changes are saved and stored in them. These logs are sent to the replicas, and the requests stored are executed, starting at a certain position. During replication, it is only the queries resulting in database changes that are transmitted, not the modified data.
Thus, the use of backup and replication ensures enhanced system stability and data integrity.

To improve performance, optimization is used at various levels:

- On the client side: caching at the browser level, minimizing resources (HTML, CSS, and JavaScript), image optimization, CSS sprites, etc.;
- On the server side: caching (Memcached), static content processing optimization (Nginx), application server setting optimization, load balancing, etc.;
- On the database end: database setting optimization, use of indexes, partitioning.

The system is managed through a graphical user interface developed on the basis of Twitter Bootstrap, which provides adequate content display in different browsers and on different devices.

The interaction between Protectimus and the customers' systems is performed through the RESTful API; data is transmitted in the XML or JSON format. To facilitate the integration processes, libraries in Java, Python, and PHP were developed. The API documentation is available on the Protectimus website in the Documentation section.

The certified authentication module allows using any tokens that operate based on the standard OATH algorithms – HOTP, TOTP and OCRA - in Protectimus. The SMS delivery module will provide one-time passwords for those users that do not want to or cannot use other types of tokens.

Integration with Protectimus

To integrate Protectimus into your project, you can use these two methods:

1. Integration using the API. For integration through the API, we provide a set of auxiliary libraries for the following programming languages: Java, Python, and PHP. If there is no client for your programming language, you can use detailed descriptions of all the API methods using this link.
2. Integration using the IFrame widget for user authentication.

User authentication is always performed for a specific resource; therefore, a user must be assigned to the resource to which this user should have access. If a user is not assigned to a resource, this user will have no access to this resource. The method of assigning a user to a resource depends on the authentication method selected. Protectimus supports several user authentication methods:

1. **User authentication with a static password.** This method requires that a user should have a password, and that this user should be assigned to the resource for which authentication is performed.

Note: Protectimus will ensure secure storage of your users' passwords and will not use them for any purpose other than their intended purpose. You can find detailed information about this on the PRIVACY POLICY page. You can find information on how to securely transmit and use your users' passwords in Protectimus in the documentation on our API.

2. **User authentication with a one-time password.** This method requires that a user should have a token, and that this user should be assigned to a resource WITH this token. This method will not work if a user and a token are assigned to a resource separately from each other.
3. **User authentication with a static password and a one-time password.** It is a combination of the two methods described above. A user must be assigned to a resource WITH a token. This user must have a password. If a user's token is deactivated, OTP authentication will not be performed, in which case only this user's static password and this user's compliance with the filters' requirements, if any, will be authenticated.
4. **Token authentication on a resource.** This method allows you not to assign a token to any specific user, but simply to verify the validity of a one-time password generated by the token. This method requires that a token should be assigned to a resource.

Note: During authentication, it is verified, among other things, whether or not a request meets the requirements of the filters on this resource.

Note: If a user's authentication is not successful because this user enters an incorrect OTP, the value of the counter of failed authentication attempts for this

token will be increased. When the threshold of failed authentication attempts for the specified resource is exceeded, a token will be locked. A token can be unlocked either through the web interface or through the API (the Editing Token method). If a user’s authentication is successful, the counter of failed authentication attempts is reset to zero if it has not exceeded the allowable limit for this resource, and if this user has not yet been blocked.

Tokens Used in Protectimus

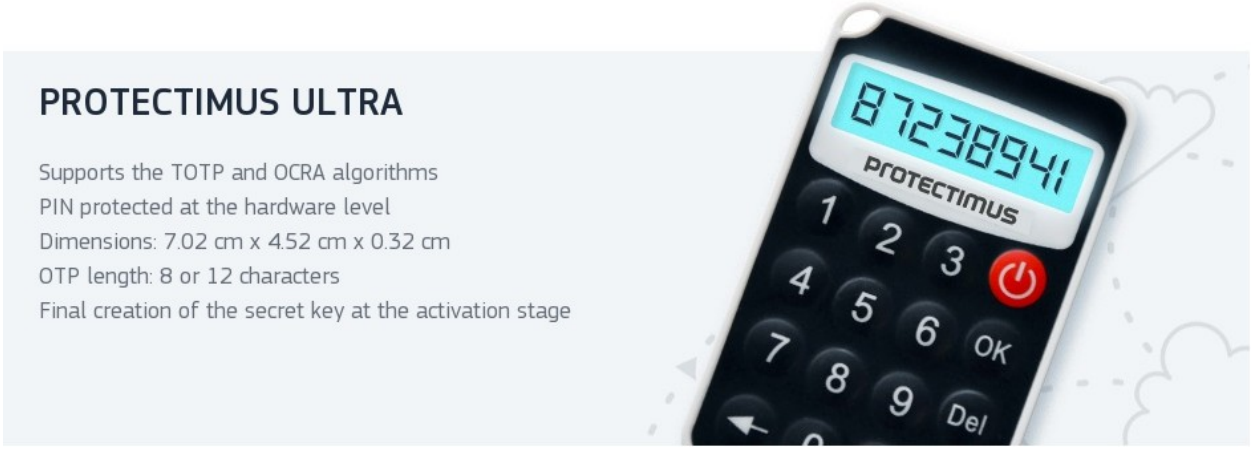
You can use different types of tokens to generate one-time passwords.

Hardware Tokens

Hardware tokens are devices intended specially for generating OTPs. They are typically small and can be used as a keychain. We offer the following hardware token models: Protectimus ONE, Protectimus ULTRA, and Protectimus SLIM. Let’s look at these tokens in more detail.

Protectimus ONE tokens are a basic solution suitable for most users that need hardware tokens.

To ensure maximum protection, Protectimus ULTRA tokens are an ideal solution; they operate based on the OCRA algorithm: the server displays a challenge, the user enters it into the token, and the token uses it to generate an OTP. The distinct feature of this token is that the secret key is finally created during the process of token activation by a user or a customer’s employee.

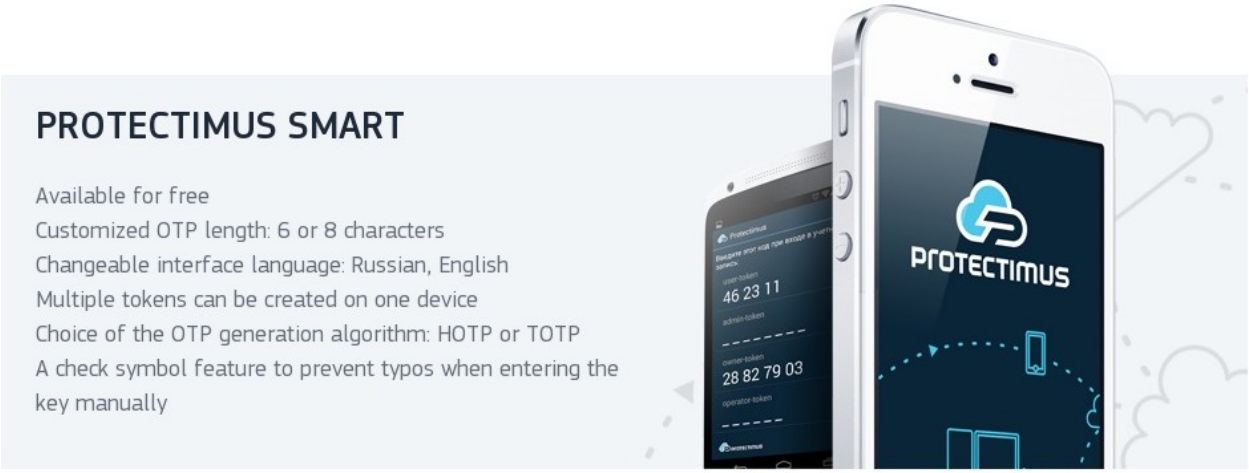


The SLIM model token is the size of a regular credit card, including its thickness. That makes it very convenient to use; this token can be designed in your corporate style.

Software Tokens

Protectimus SMART is an application from Protectimus used to generate OTPs; it can be installed on Android and iOS. The application has no connection with the server, which eliminates the risk of any unauthorized impact on the authentication system by means of taking over the communications channel.

Protectimus SMART allows customers to save money on the implementation of two-factor authentication and allows users to simplify its use. You can create several tokens in the application, which allows working with different independent systems with two-factor authentication using one device.

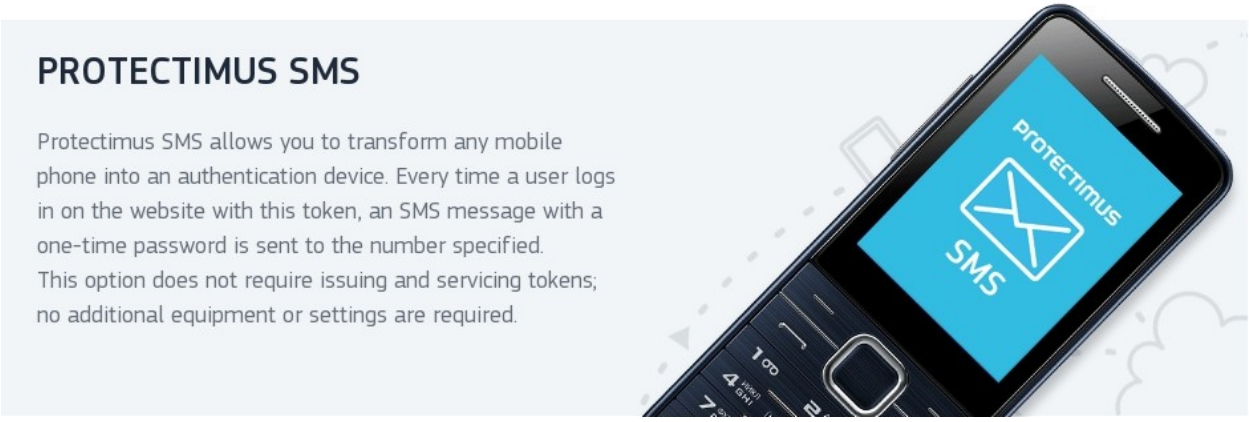


SMS and Email Tokens

One-time passwords will be sent to the phone number or the email address specified. Using such tokens simplifies interaction with users; users are not required to perform any additional actions, and they can be located in various parts of the solar system.

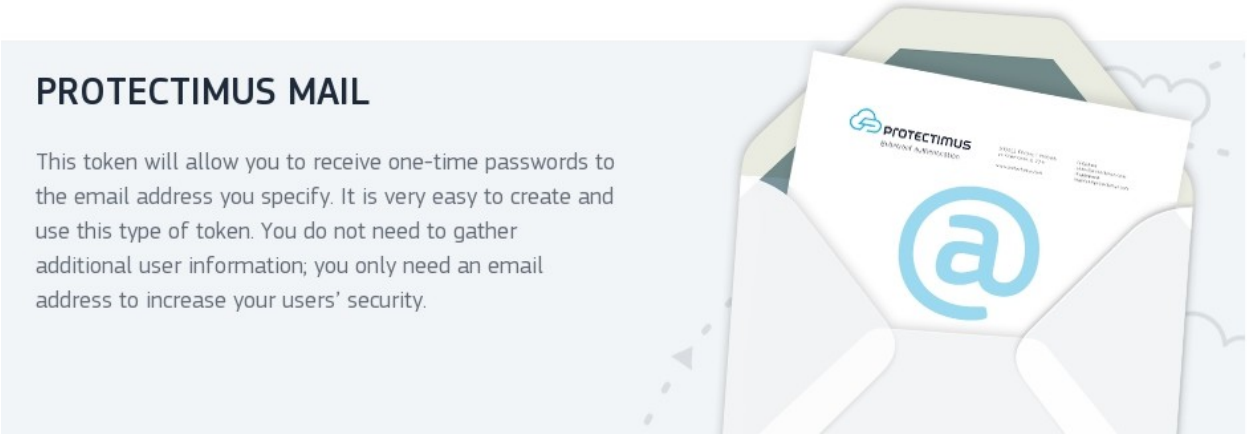
Users are already familiar with the practice of confirming their identity with codes received by phone, which means that you will not have to train them or explain why it is necessary.

When SMS tokens are used, a fee is charged for the services of message delivery operators. We do everything possible to lower this fee and make this instrument even more convenient and affordable for you.



You can use email tokens free of charge; besides, it is the most simplified protection method because access to an email inbox is typically protected with a password. It is important to understand that the level of protection provided by this token is

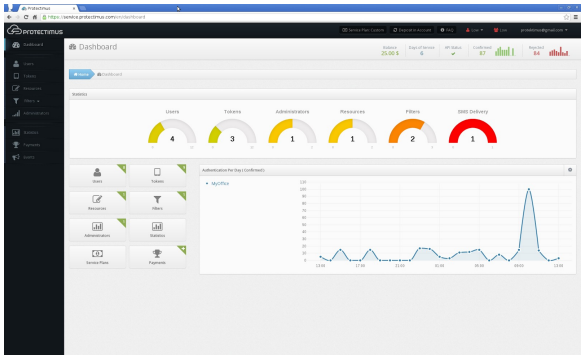
lower than that of other types of tokens if the same password is used to access email and the protected resource.



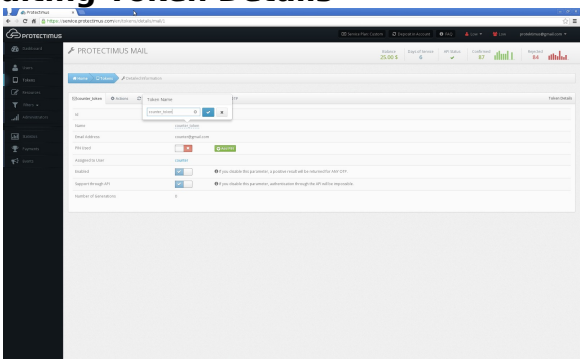
Graphical User Interface

The graphical user interface is built on the basis of the Tapestry 5.3 framework with the use of Bootstrap, jQuery, AJAX, and JavaScript. Below you can see several screenshots from the Protectimus system.

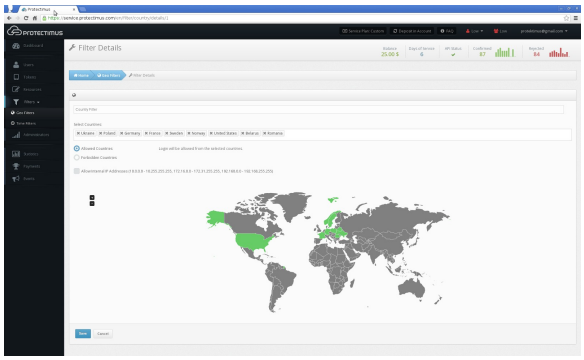
Control Panel



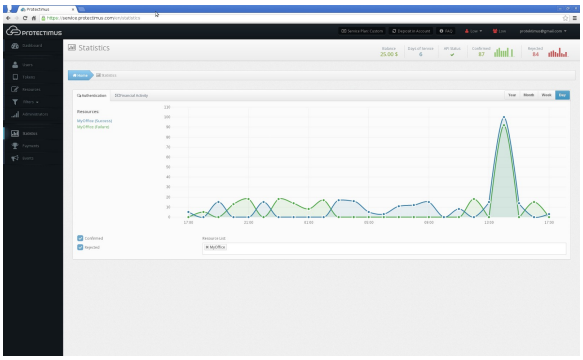
Editing Token Details



Geo Filters



Statistics



Special priority is given to users' convenience; any action can be performed in a maximum of three clicks. Most operations are performed via an AJAX request, which increases the speed of the system's operation. Prompts and recommendations allow quickly familiarizing oneself with the system and the way it operates.

Contact Information

Our Services

Potential partnership, sales:

sales@protectimus.com

Problems, questions, feedback:

support@protectimus.com

Corporate Information

Protectimus Solutions LLP

Tel.: +38(057) 751 62 34