

API How to Guide

The Agent-XML API is an XML based API used for integrating PINsafe with other applications.

There are 4 subsets of the API, that cover the following areas

- **Authentication**

Authentication, Change PIN, PIN Reset, Start Authentication Session, Request security Strings etc

- **Admin functions**

Add new users, set user details (eg mobile phone number), synchronise a repository, delete users

- **Helpdesk functions**

Unlock user, send user new credentials, set user policies etc

- **Reporting functions**

Retrieving lists of idle users etc

All these APIs follow the structure described in this article.

Contents

- **1 Principles**
- **2 Structure**
- **3 Commands**
- **4 Examples**

Principles

The APIs are based around XML documents for both request and reply.

The basic idea is to build a document containing details of operation(s) to be performed and the user(s) on which they are to be performed, submit it to PINsafe via HTTP and PINsafe will reply with a document detailing which operations succeeded and which, if any, failed. All operations via the API will be logged in the standard PINsafe log.

Only authorised PINsafe Agents will be able to submit requests. This is in line with Agents used for authorisation. The API is case sensitive, the correct case shown in the included examples.

Requests are sent to PINsafe via an HTTP POST using http or https depending on the configuration of the PINsafe server. For appliances the default will be https over port 8080.

The requests are posted to the pinsafe context followed by AgentXML for the authentication API and AdminXML for the Admin, Helpdesk and reporting API.

For example `http://<ip address>:8080/pinsafe/AgentXML` or `https://<ip address>:8080/pinsafe/AdminXML`

Structure

The structure of a request that all API requests are contained within a request element. This request element specifies the type of request and also includes the shared secret for agent authentication.

eg an authentication request would be contained within an SASRequest tag

```
<?xml version="1.0" ?>
<SASRequest secret="MyAdminAgent" version="3.4">
  .
  .
</SASRequest>
```

Whereas an admin API request would be contained within an AdminRequest element.

```
<?xml version="1.0" ?>
<AdminRequest secret="MyAdminAgent" version="3.4">
  .
  .
</AdminRequest>
```

...or else a HelpdeskRequest element.

```
<?xml version="1.0" ?>
<HelpdeskRequest secret="MyAdminAgent" version="3.4">
  .
  .
</HelpdeskRequest>
```

The syntax of AdminRequest and HelpdeskRequest are more or less identical. The major difference is that AdminRequest can make major changes to the database, such as adding and deleting users, whereas HelpdeskRequest is largely read-only, although it does support some changes to existing users, such as PIN change. Conversely, AdminRequest can only affect users in the repository with the same name as the Agent, so the Agent must have "Can Act as Repository" set to Yes. HelpdeskRequest can affect any users.

The responses from the PINsafe server are similarly contained within response tags eg

```
<?xml version="1.0" ?>
<SASResponse secret="MyAdminAgent" version="3.4">
  .
</SASResponse>
```

and

```
<?xml version="1.0" ?>
<AdminResponse secret="MyAdminAgent" version="3.4">
  .
</AdminResponse>
```

Within these elements for elements that specify the particular request. For details of these elements refer to the article relating to that part of the API.

Commands

Authentication

<changePIN> : Change Users Credentials

<exists> : Does a user have an account.

<login> : Perform PINsafe authentication

<ping> : Ping PINsafe application

<reset> : Perform a PIN reset

<resetcode> : Request a PIN Reset code

<startsession> : Start an authentication session

<securitystrings> : Request a set of security strings

Helpdesk functions

<reset> : Send user new credentials

<strings> : send dual channel security string(s) to user

<update> : Change user policies or credentials

Admin functions

<create> : create a new user and optionally set user details.

<delete> : delete a user.

<read> : read the details of a given user.

<reset> : Send user new credentials

<sync> : Synchronise with a repository

<update> : Change user policies, credentials, transports etc

Reporting functions

<disabled> : Report on disabled users

<idle> : Report on idle users

<locked> : Report on locked users

Examples

We have a number of applications that use these APIs, available as Windows executable programs. These can be found on a separate page.