

Safeguard Ecommerce Integration / API

Product Manual

Version 3

Revision 1.12



Table of Contents

1.	INTRODUCTION	4
1.1	Available commands	4
2.	HOW THE ADMIN SYSTEM IS EXPECTED TO BE USED OPERATIONALLY	6
3.	BUSINESS PROCESSES.....	8
4.	PROTOCOLS AVAILABLE FOR CALLING THE LOCKLIZARD SERVLET	11
5.	FORMAT OF SERVER RESPONSES.....	13
6.	QUOTED STRINGS.....	16
7.	DEFINITION OF COMMON PARAMETERS AND FIELD ENCODING.....	17
7.1	Use of non alpha-numeric characters in username / password.....	17
8.	DATE AND TIME	19
9.	DEFINITION OF AVAILABLE COMMANDS	20
9.1	Request: List customers.....	20
9.2	Request: List customer	22
9.3	Request: Get customers count.....	24
9.4	Request: Add customer	25
9.5	Request: Update customer account validity	28
9.6	Request: Set customer license count	30
9.7	Request: Update customer license count.....	31
9.8	Request: Get customer license	32
9.9	Request: Set customer Web Viewer access.....	35
9.10	Request: Get customer Web Viewer access	38
9.11	Request: Update customer views	39
9.12	Request: Update customer prints	40
9.13	Request: List publications.....	41
9.14	Request: List publication customers	42
9.15	Request: Get publications count.....	43
9.16	Request: Add publication	44
9.17	Request: Grant publication access.....	45
9.18	Request: List documents	48
9.19	Request: List publication documents	50
9.20	Request: List documents direct access	52
9.21	Request: Get documents count	53
9.22	Request: Grant document access.....	54
10.	FAQs.....	57
10.1	What are the steps involved when a customer purchases a protected PDF document from my web site?	57
10.2	How can I add a customer and give them access to a document?	58
10.3	How can I add a customer and give them access to a publication?	59
10.4	Do you provide sample code for integration into our shopping cart system?.....	60
10.5	How can I extend a customer's account for another year?	60

10.6	How can I extend a customer's subscription (publication) access?.....	60
11.	UPGRADING FROM SAFEGUARD V2 TO SAFEGUARD V3.....	61
11.1	General changes.....	61
11.2	New commands	62
11.3	Modified commands	62



1. Introduction

The Locklizard administration service provides a graphical user interface (GUI) that allows publishers to carry out the administration of their protected documents, publications and customers. Whilst this manual interface is well suited to the smaller publisher, larger organizations require automation between their own applications (web site customer administration, credit card processing, internal file management) and the Locklizard administration service.

To support automation and loose integration, Locklizard provide a servlet that publishers can call in order to pass information automatically into the Locklizard administration system, or to cause the Locklizard administration to take a specific action, such as provide information about a customer record.

This document describes the commands currently available, and describes how they are intended to be used in an operational system.

1.1 Available commands

Currently available commands are:

List customers	shows you a list of all current customers
List customer	shows the documents and publications a specific customer has access to and whether they have registered or not
Get customers count	Returns a count of the number of customers you have
Add customer	add a new or returning customer and one or more publications that they have access to
Update customer account validity	updates the start and expiry dates of a customer's account
Set customer license count	sets the license count for a given customer account to the number you specify
Update customer license count	increments the license count value set on the administration system for a given customer account

Get customer license	obtain the license file contents for a specific customer
Set customer web viewer access	grants or revokes customer access to the Web Viewer
Get customer web viewer access	displays the Web Viewer access status for a customer and their login credentials
Update customer views	updates the number of views the customer has available
Update customer prints	updates the number of prints the customer has available
List publications	obtain a list of current publications
List publications customers	obtain a list of all the customers given access to a specific publication
Get publications count	Returns a count of the number of publications you have
Add publication	adds a publication
Grant publication access	grant access to publication(s) to a single or multiple customers and update publication access periods
List documents	obtain a list of current documents
List publication documents	obtain a list of all the documents in a publication
List document direct access	obtain a list of all the documents that are not part of a publication or available automatically to all customers
Get documents count	Returns a count of the number of documents you have
Grant document access	grant access to document(s) to a single or multiple customers and update document access periods

2. How the Admin system is expected to be used operationally

The Locklizard Administration system is not intended to replace a CRM service, or to provide the features that would be considered normal for a CRM service.

It has been provided to support basic administrative requirements for issuing and monitoring license controls, for managing publications and document records, and for identifying customers using minimal details.

Publishers are expected to have their own administration and accounting systems, and there is no provision for any of those services within Locklizard Administration.

Locklizard Administration expects to be provided with certain minimal information in order to identify a customer: their name and their email address, so that basic licensing information may be sent to the customer so they can register their license with the supplying publisher.

The eCommerce integration feature provides for a number of automated activities. However, a number of manual activities must still be carried out by the Publisher to:

- Protect a specific document;
- Make the protected document available in one or more publications;
- Delete a customer, document, or publication.

Locklizard Administration maintains internal references (identities) that correspond to customers, publications and documents. The administration system can accept publications with the same title and documents with the same title, but will identify them as separate entities for administration purposes. (This feature allows a publisher to withdraw a document from within a publication, that may not have been published correctly, and substitute a replacement document seamlessly.)

Locklizard Administration provides the Publisher with lists of customer names, publication names and document names and the Locklizard internal identities that are used as absolute identifiers, which must be used in the integration commands.

Locklizard Administration requires that internal identifiers are used on a number of occasions, so the Publisher must develop their own method of determining the Locklizard identity information that will be used for the customer, the publication or the document that is to be referenced.

Locklizard Administration also anticipates that Publishers may wish to periodically list out the Locklizard codes and match them to information in their other data stores, and this capability is supported.

Publishers hosting the Locklizard administration system on their own server have access to the database dataset layouts, and we anticipate that they may prefer to implement their own access to these rather than use the servlet commands provided. However, Locklizard disclaim any responsibility for the interpretation of information held in the Administration database and specifically deny any responsibility where Publishers change information in that database except using the commands provided by Locklizard.

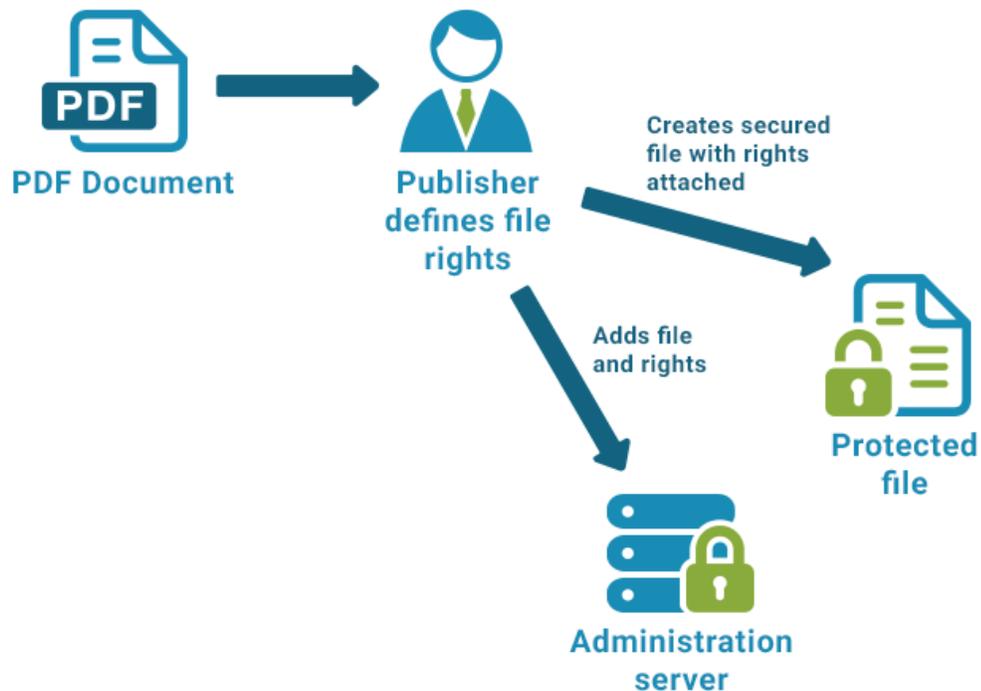
3. Business processes

It is important to understand the interaction with the Locklizard Administration system and surrounding management systems. The eCommerce integration service has been designed to fulfill specific scenarios.

The scenarios are characterized by the following diagrams (please note that we have not used a formal business process specification method for presenting the information in these diagrams since there are several competing methods that claim supremacy).

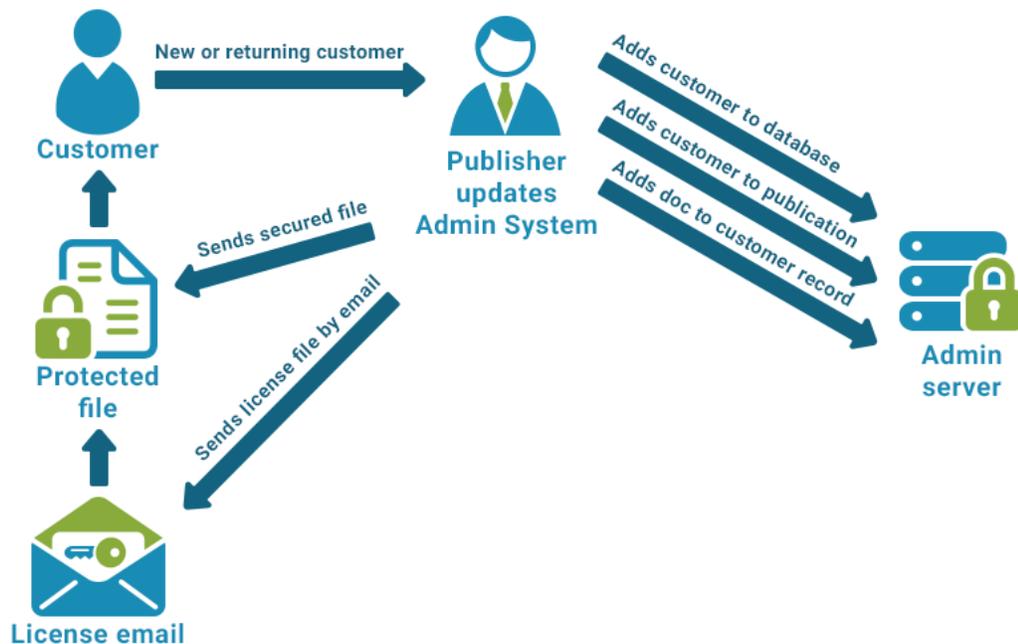
There are three basic processes that must be followed

The pre-sales process, in which the Publisher sets up publications and then protects documents and determines the rights that users of the document are granted. This can be expressed pictorially as follows:



During the selling process, the Publisher adds a new customer or sells additional publications or documents to an existing customer. In either case it is necessary for the Publisher to tell the Locklizard Administration system to authorize a specific customer to have access to either a publication or to a document (please recall that if a document is part of a publication the sales process is not required because the customer has already been granted rights to access the information as part of the publication).

The sales process is characterized in the following picture.



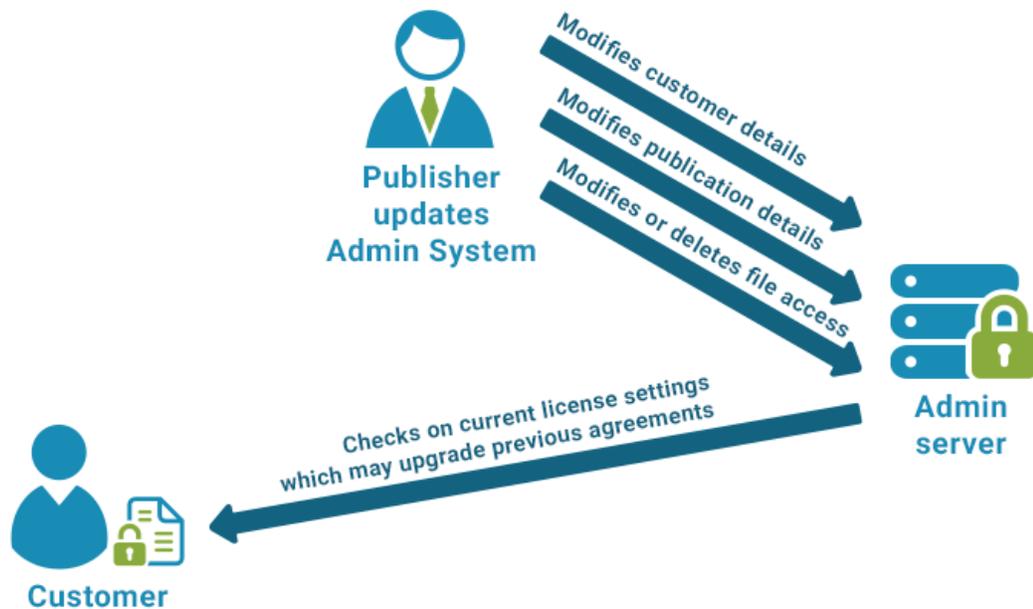
The eCommerce integration feature supports the ability of the publisher to:

- (1) Add a customer to the Administration system;
- (2) Link the customer to one or more publications;
- (3) allow specific customers access to specific documents.

Please note that if the option to protect a document for all customers has been used there is no need to use the integration feature because, by default, any valid customer gaining access to that protected document would be able to use it without any action being needed by the Administration system.

We have not shown the customer registration process as part of these processes. This is because it takes place outside the direct functions of the Publisher and is a fully automated part of the system. Accessing information relating to the registration of customers is a separate feature.

Finally there is the post-sales process where the Publisher may decide to change the license conditions applicable to either documents or customers. They are summarized as follows:



4. Protocols available for calling the Locklizard servlet

You may interact with the Locklizard Administration system using either the HTTP or HTTPS protocols. The choice is yours. If you are working internally then you may decide that there is no additional security value to be obtained by using the more secure HTTPS protocol. On the other hand, if you are interacting with the generally accessible Locklizard Administration servers you may prefer to implement the HTTPS protocol in order to prevent others from being able to read the contents of your messages.

There are therefore two methods by which you may initiate a call to the Locklizard Administration system:

<http://www.website.com/Interop.php?> or

<https://www.website.com/Interop.php?>

The Locklizard Administration system uses port 80 for non-secured HTTP messages and port 443 if secure HTTP is to be used.



NOTE: Server URL

The server URL you enter depends on the URL you use to access the Locklizard administration system. The <http://www.website.com> URL will need to be replaced by the appropriate server URL.

For example, if the URL to access your admin system is <https://www.mywebsite.com/safeguard3/> then the URL you would use is <https://www.mywebsite.com/safeguard3/Interop.php?>

The Interop.php? command **MUST** always be used.

Please note that if you invoke the HTTPS URL then your own server side application must be programmed to know how to make and interpret HTTPS calls. This must be added by yourselves and cannot be added by us. Also, you MUST have an appropriate SSL key pair that are to be used for the application.

Locklizard take no responsibility for protecting information (whether personal or subject to privacy regulation) and make it a specific requirement of the Publisher that they take all such steps as are necessary to apply such controls over information they use when sending commands to the Locklizard Administration system as is necessary to comply with such regulations as they are subject to.

Locklizard specifically disclaim any warranty that their system meets any particular requirements laid down by any national regulator or international convention for the security or privacy of information being exchanged using the servlet commands.



IMPORTANT NOTE - Web server issues with Cold Fusion 5.x

Cold Fusion 5.x has a bug that incorrectly issues HTTP PUT and GET commands causing the Locklizard administration server to generate incorrect licenses that do not contain a port number.

If you are running Cold Fusion 5.x then you can purchase CFX_HTTP5 from <http://www.adiabata.com/Docs/Products.cfm> that fixes this bug and others.

Alternatively you can upgrade to Cold Fusion 6 or above.

5. Format of Server Responses

Responses to commands made to the Locklizard Administration system are in plain text (following receipt at the server in the case of HTTPS formatted commands) for immediate interpretation by the calling module.

The server replies use the following format:

- all fields are separated by a single blank space.
- all fields with the exception of the status messages (OK and Failed), the customer/document/publication counts, and the license file information (displayed as a result of using the `get_customer_license_file` command), are formatted with quotes.
- the data lines separator (linefeed) returned for ecommerce commands is formed by the characters 13 and 10. CHAR(13) - CHAR (10) or /r/n (13-10) is the default carriage return for web services, XML and SOAP.

For example, if the server responds with the three fields (field 1 is "customer id", field 2 is "first and last name", field 3 is "email address", then the response looks like this:

```
"1254" "First Last Name" "address@domain.com"
```

You can therefore parse the server response and determine each field, given the above field structure.

Example server response:

This is an example of a query sent to the Locklizard Administration server and the response received. The command is to list all customers.

Command

```
https://www.website.com/Interop.php?un=<username>&pw=<password>&action=list_customers
```

Server response

OK

```
"26728" "John Doe" "jdoe@locklizard.com" "SOAS" "08-10-2008" "never" "1" "false"
"true" "true"
"26772" "J Smith" "john@smith.co.uk" "John company" "08-11-2009" "never" "3"
"false" "true" "false"
"7219" "Simon Peters" "simonp@compuserve.com" "Bramleys" "01-15-2010" "never"
"0" "true" "false" "true"
```

The server has responded by listing all the customers (three in this case). The fields can be interpreted as follows:

- OK = server response is valid
- 26728 = customer id
- John Doe = customer name
- jdoe@locklizard.com = customer email
- SOAS = customer company name
- 08-10-2008 = date from when the customer account is valid
- never = customer account never expires (otherwise a date would be shown)
- 1 = number of licenses available
- false = the account is active
- true = the customer has registered
- true = Web Viewer is enabled on the account

- 26772 = customer id
- John Smith = customer name
- john@smith.co.uk = customer email
- John Company = customer company name
- 08-11-2009 = date from when the customer account is valid
- never = customer account never expires (otherwise a date would be shown)
- 3 = number of licenses available

- false = the account is active
- true = the customer has registered
- false = Web Viewer is NOT enabled on the account

- 7219 = customer id
- Simon Peters = customer name
- simonp@compuserve.com = customer email
- Bramleys = customer company name
- 01-15-2010 = date from when the customer account is valid
- never = customer account never expires (otherwise a date would be shown)
- 0 = number of licenses available
- true = the account is suspended
- false = the customer has NOT registered
- true = Web Viewer is enabled on the account

6. Quoted strings

Most requests will return quoted strings (enclosed within quotes).

The following fields are NOT enclosed within quoted strings:

- status messages (OK and Failed);
- customers/documents/publications counts (displayed as a result of using the `get_XXXXXXXXXX_count` command).;
- license file information (displayed as a result of using the `get_customer_license_file` command).

The following 'escape' sequences may also be encountered in a quoted string:

`\` – for a quote in the original string

`\\` - for a backslash in the original string

It is the responsibility of the Publisher's application to properly interpret these sequences. They are considered to be a standard way of presenting information for programming purposes.

7. Definition of common parameters and field encoding

The Locklizard Administration system uses GET or POST requests, and there are a number of common parameters that apply to all GET and POST commands used in the servlet.

There are two data elements that must be present in all GET and POST commands to be sent to the Locklizard Administration system, whether it is hosted by Locklizard or by you.

- un – publisher username
- pw – publisher password

The values for these fields are the same as the logon information used to logon to the Locklizard administration system. This information is contained in the Writer Registration email that is sent to you when your Publisher account is created.

For example:

username: LocklizardSupport

password: vmkbxyj437ctg

Using the variables above, an example command to list customers would therefore be:

https://www.website.com/Interop.php?un=LocklizardSupport&pw=vmkbxyj437ctg&action=list_customers

7.1 Use of non alpha-numeric characters in username / password

If you are using non alpha-numeric characters in your username and/or password, then these must be encoded in your HTML string. You can use a URL encoder/decoder tool such as <http://meyerweb.com/eric/tools/dencoder/>

For example if you had the following username and password:

username: Support@Locklizard.com

password: vmkb*\$xyj437ctg£#

Then this would have to be encoded as:

username: Support%40Locklizard.com

password: vmkb*%24xyj437ctg%C2%A3%23

Using the variables above, an example command to list customers would therefore be:

[https://www.website.com/Interop.php?un=Support%40Locklizard.com
&pw=vmkb*%24xyj437ctg%C2%A3%23 &action=list_customers](https://www.website.com/Interop.php?un=Support%40Locklizard.com&pw=vmkb*%24xyj437ctg%C2%A3%23&action=list_customers)

8. Date and time

The time zone used for date and time is GMT. (This is almost identical to UTC time – the internationally coordinated time line used by the atomic clocks around the world, but the difference is not significant in this application).

Therefore, independent of the local time at the Publisher or their customer, all references to date and time will be converted into GMT for action by the Locklizard Administration system. This means that if a Publisher in New York sets the end date for a document as 06-23-2006 it will expire in all time zones at 23:59:59 GMT. In effect this will be 18:59:59 EST and 15:59:59 PST.

The format of the Date is mm-dd-yyyy,

The format of the Time is hh:mm:ss with hours in range 0-23, minutes and seconds in the range 0 – 59.

9. Definition of available commands

9.1 Request: List customers

This command is used to list existing customer records with their customer ids.

Parameters:

action=list_customers

- webonly=1 – list only customers with Web Viewer access.
- OR**
- pdconly=1 – list only customers without Web Viewer access.

Output:

If the request succeeds then the first line of the output will be OK, followed by the list of customers, one per line. For each customer the following information (separated by spaces) will be returned:

- customer id. The Administration system returns to you its internal customer id value, which is unique, and which you MUST use in other commands rather than the name and email. Please be sure to collect this response if you are going to issue other commands for this customer such as granting access to publications and documents. The customer ID is also displayed in the Administration system under each individual customers account.
- customer name
- customer email address
- customer company name
- date from when the customer account is valid
- date when the customer account will expire or never if doesn't expire
- number of licenses
- if the account is suspended (suspended=true, not suspended=false)
- account registration status (registered=true, not registered=false)

- whether the Web Viewer is enabled on the account (enabled=true, not enabled=false)

If the request fails the first line of the output will be Failed and on the next line a description of the error as a character string.

Example request 1:

To request the customer id list held by the Locklizard administration system use a request like this:

```
https://www.website.com/Interop.php?un=<replace with publisher
username>&pw=<replace with publisher password>&action=list_customers
```

Example output:

```
OK
"5167" "Arnie" "info@locklizard.com" "" "07-27-2007" "never" "0" "false" "true"
"true"
"3840" "Chris" "support-team@locklizard.com" "" "06-15-2007" "never" "0"
"false" "true" "true"
"2077" "John Adams" "john.adams@barnacles.com" "Barnacles, Inc." "05-02-
2007" "never" "1" "false" "true" "false"
"13498" "John Doe" "johndoe@yahoo.com" "" "04-01-2008" "never" "2" "false"
"false" "false"
"1164" "Locklizard Demo User" "demo@locklizard.com" "" "03-19-2007"
"never" "2958" "false" "true" "true"
"6352" "Paul Simmons" "sales@locklizard.com" "Locklizard" "02-15-2005" "03-
16-2012" "0" "false" "false" "false"
"1982" "Robert Howard" "support@locklizard.com" "" "02-15-2005" "never"
"0" "false" "true" "true"
```

Example request 2:

To request the customer id list held by the Locklizard administration system for customers with Web Viewer access only use a request like this:

```
https://www.website.com/Interop.php?un=<replace with publisher
username>&pw=<replace with publisher
password>&action=list_customers&webonly=1
```

9.2 Request: List customer

This command is used to list a single customer with details of the documents and publications they have access to and whether they have registered or not. You can list a customer using either their id or their email address.

Parameters:

action=list_customer

- custid – customer ID
- OR**
- email – customer email
 - nodocs – specifying the parameter nodocs=1 will prevent the listing of document IDs.

Output:

If the request succeeds then the first line of the output will be OK. The next line will contain the following information:

- customer id
- customer name
- customer email address
- customer company
- date from when the customer account is valid
- date when the customer account will expire or never if doesn't expire
- number of licenses
- if the account is suspended (suspended=true, not suspended=false)
- account registration status (registered=true, not registered=false)
- documents the customer has been granted access to
- publications the customer has been granted access to
- whether the Web Viewer is enabled on the account (enabled=true, not enabled=false)

If the request fails the first line of the output will be Failed and on the next line a description of the error as a character string.

Example request using customer id:

To list a single customer with details of the documents and publications they have access to and whether they have registered or not:

```
https://www.website.com/Interop.php?un=<replace with publisher
username>&pw=<replace with publisher
password>&action=list_customer&custid=id<replace with actual customer id>
```

Example request using customer email:

To list a single customer with details of the documents and publications they have access to and whether they have registered or not:

```
https://www.website.com/Interop.php?un=<replace with publisher
username>&pw=<replace with publisher
password>&action=list_customer&email=email<replace with customer email
address>
```

Example output:

OK

```
"238" "Brendon Silvers" "brendon.silvers@locklizard.com" "Locklizard" "02-15-
2010" "02-15-2012" "0" "false" "true"
```

```
"72,73,74,75,113,114,115,116,118,119,165 ," "14,26" "true"
```

In the example above, "72,73,74,75,113,114,115,116,118,119,165 ," are the document ids the customer has been granted access to and "14,26" are the publication ids the customer has been granted access to. The "false" field shows that the customer account is active (not suspended), and the customer has registered since their account registration status is showing as "true". The final "true" field displayed after the publication IDs shows that the Web Viewer is enabled on that customer's account.

9.3 Request: Get customers count

This command is used to list the number of customers you have.

Parameters:

action=get_customers_count

- weonly=1 – counts only those customers with Web Viewer enabled on their accounts.

Output:

If the request succeeds the output will be OK.

If the response is OK then the following line will contain the number of customers.

Example request 1:

To get the number of customers the request will be:

```
https://www.website.com/Interop.php?un=<replace with publisher
username>&pw=<replace with publisher
password>&action=get_customers_count
```

Example output:

```
OK
9
```

Example request 2:

To get the number of customers who have Web Viewer enabled on their accounts the request will be:

```
https://www.website.com/Interop.php?un=<replace with publisher
username>&pw=<replace with publisher
password>&action=get_customer_count&weonly=1
```

9.4 Request: Add customer

This command is used to add a new customer to the Administration system.

Parameters:

action=add_customer

- name – customer name (either %20 or a space can be used to separate the first and surname. If using CURL you need to use %%20).
- email – customer email address. The license information generated by Locklizard administration will be automatically sent to this email address
- company – customer company (Optional)
- start_date – date when the account becomes valid. Must be in the format mm-dd-yyyy
- end_type – possible values: text “date” or the text “unlimited” (without the parentheses). If unlimited is used, then the customer’s account will not expire. If a date is used then an additional parameter specifying the date until the account is valid must be provided (see below)
- end_date – date until the account is valid. This parameter is required only if end_type is set to date. Must be in format mm-dd-yyyy
- licenses – number of licenses. Value must be a number greater than 0
- publication – grants access to the stated publication(s). The value must be a valid publication identifier (see [List publications](#) on how to get the publication id). If you want to grant access to more than one publication then you need to separate each publication id by a comma (e.g. 100,101). This parameter is optional. If it is missing no error will be reported because you may simply wish to add a customer and sell a publication at a later date, or sell a document using the [Grant Document access](#) command.
- noregemail – stops the sending of the registration email. If the administration system is set to send registration emails by default then you can use the value 1 to prevent sending of the registration email. For example, *&noregemail=1*.

- webviewer – enables Web Viewer access. The username and password will be automatically generated by the system. The username will be the user’s email address. The generated username and password will be displayed in the output (as per example request 3 below).

Output:

If the request succeeds the output will be OK.

If the response is OK then the following line will contain the unique customer id that has been allocated in the Administration system. This will be an integer number.

If request fails the first line of the output will be Failed followed on the next line by the relevant error description as a character string.



NOTE: If there is already an existing Administration record for a customer having the same **email address** then the customer record will be **updated** with the new end date, allocated licenses (the value you enter will **OVERWRITE** the license count value set on the administration system), and additional publication access. If you enter a new start date it will be ignored – you have to use the `update_customer_account_validity` command if you want to change the customer start date.

If the request succeeds the output will be:

OK.

Existing customer account successfully updated.

Example request 1:

To create a customer account valid from 1st April 2008 to 1st May 2008 and grant them access to 2 publications (let’s assume there are 2 publications already created defined with ids 100 and 101) use a request like this:

https://www.website.com/Interop.php?un=<replace with publisher username>&pw=<replace with publisher password>&action=add_customer&name=John%20Doe&email=johndoe@yahoo

o.com&start_date=04-01-2008&end_type=date&end_date=05-01-2008&publication=100,101&licenses=1

Example request 2:

To create a customer account valid from 04-01-2008 with unlimited validity period the request will be:

`https://www.website.com/Interop.php?un=<replace with publisher username>&pw=<replace with publisher password>&action=add_customer&name=John%20Doe&email=johndoe@yahoo.com&start_date=04-01-2008&end_type=unlimited&publication=100,101&licenses=1`

The customer first name and surname, in this example John Doe, can either be separated by a space or %20. Both are accepted by the administration system. If using CURL you need to use %%20.

Example output:

OK
"13498"

Example request 3:

To create a customer account valid from 04-01-2014 with unlimited validity period and Web Viewer access the request will be:

`https://www.website.com/Interop.php?un=<replace with publisher username>&pw=<replace with publisher password>&action=add_customer&name=John%20Doe&email=johndoe@yahoo.com&start_date=04-01-2014&end_type=unlimited&licenses=1&webviewer=1`

Example output:

OK
"13498"
"johndoe@yahoo.com"
"uxT@@U9#F"

9.5 Request: Update customer account validity

This command is used to update the start and expiry dates of a customer's account.

It is used to enable you to extend a customer's account if they decide to subscribe to your services for another year, and/or to change their start date.

Parameters:

action=update_customer_account_validity

- custid – id of the customer
- start_date – date when the account becomes valid. Must be in the format mm-dd-yyyy. This field is OPTIONAL and only needs to be used if you want to change the start date for the account.
- end_type – possible values: the text "date" or the text "unlimited" (without the parentheses). If unlimited is used, then the customer's account will not expire. If a date is used then an additional parameter, end_date specifying the date until the account is valid must be provided
- end_date – date until the account is valid. This parameter is required only if end_type is set to date. Must be in format mm-dd-yyyy

Output:

If the request succeeds then the output will be the text 'OK'.

If the request fails the first line of the output will be Failed followed on the next line by the error description.

Example request 1:

To specify a new end date for a customer's account (let's assume the customer ID is 19):

```
https://www.website.com/Interop.php?un=<replace with publisher  
username>&pw=<replace with publisher  
password>&action=update_customer_account_validity&custid=19>&end_type  
=date &end_date=07-12-2009
```

Example request 2:

To specify a new start and end date for a customer's account (id is 19):

```
https://www.website.com/Interop.php?un=<replace with publisher  
username>&pw=<replace with publisher  
password>&action=update_customer_account_validity&custid=19&start_date  
=07-12-2008&end_type=date &end_date=07-12-2009
```

Example request 3:

To change a customer's account (id is 19) so that it does not expire:

```
https://www.website.com/Interop.php?un=<replace with publisher  
username>&pw=<replace with publisher  
password>&action=update_customer_account_validity&custid=19&end_type=  
unlimited
```

9.6 Request: Set customer license count

This command is used to set the license count for a given customer account to the number you specify.

The value you enter will **OVERWRITE** the license count value set on the administration system.

Parameters:

action=set_customer_license_count

- custid – id of the customer
- licenses – number of licenses you want to grant.

Output:

If the request succeeds then the output will be the text 'OK'.

If the request fails the first line of the output will be Failed followed on the next line by the error description.

Example request:

To set the number of licenses on a customer's account to 2 (let's assume the customer ID is 19):

```
https://www.website.com/Interop.php?un=<replace with publisher  
username>&pw=<replace with publisher password>&action=  
set_customer_license_count&custid=19&licenses=2
```

In this example, if the customer account already had 5 licenses assigned to it then the customer account will now have 2 licenses assigned to it after executing this command.

9.7 Request: Update customer license count

This command is used to update the license count for a given customer account to the number you specify.

The value you enter will INCREMENT the license count value set on the administration system.

Parameters:

action=update_customer_license_count

- custid – id of the customer
- licenses – number of licenses you want to add

Output:

If the request succeeds then the output will be the text 'OK'.

If the request fails the first line of the output will be Failed followed on the next line by the error description.

Example request:

To increase the number of licenses on a customer's account by 2 (let's assume the customer ID is 19):

```
https://www.website.com/Interop.php?un=<replace with publisher  
username>&pw=<replace with publisher password>&action=  
update_customer_license_count&custid=19&licenses=2
```

In this example, if the customer account already had 5 licenses assigned to it then the customer account will now have 7 licenses assigned to it after executing this command.

9.8 Request: Get customer license

This request enables you to display the license file that would be sent to a customer on a web page so that it is immediately available to your customer, rather than them having to wait for an email to arrive before they can use your protected documents.

How this information is presented to your customer depends upon your own house style, but you could put a button or link to your web page, like Save File, so your customers can save the license file to disk. The information starting with -----BEGIN PDC Viewer LICENSE----- and ending with -----END PDC Viewer LICENSE----- should be saved to a file ending with a .llv file extension, which the user can then download to their hard disk to register.

The Locklizard Administration system (whether hosted by Locklizard or by you) will still email out license codes to users, but this command helps ensure that the license file is immediately accessible to your customer. Note however that you can stop license files being emailed to your customers from the Administration System by setting the 'Change license file delivery' to disabled.

It is recommended that you also use the text from the license file email that tells your customers how to download, install and register the software and start viewing protected documents so that they can begin using your documents as soon as possible.



IMPORTANT NOTE: Customer license files

If you use this feature to provide downloads of customer license files from your web site we recommend you put in place the necessary measures to ensure that these files are not available to hackers to intercept, and are deleted as soon as they are no longer required.

Parameters:

action=get_customer_license

- custid – id of the customer
- link=1 – displays the license file as a clickable link

Output:

If the request succeeds the output will be the license content.

If the request fails the first line of the output will be Failed followed on the next line by the error description.

Example request 1:

To get the license content for a customer and **display it as a clickable link** use a request like this (let's assume the customer id is 145):

```
https://www.website.com/Interop.php?un=<replace with publisher
username>&pw=<replace with publisher
password>&action=get_customer_license&custid=145&link=1
```

Example output:

```
https://www.website.com/safeguard/?action=getvlic&un=eb5cd826a22e17ce4291307
df0vb5deb&pw=e293f0223dd9b54612586440b40f1022
```

Example request 2:

To get the license content for a customer use a request like this (let's assume the customer id is 145):

```
https://www.website.com/Interop.php?un=<replace with publisher
username>&pw=<replace with publisher
password>&action=get_customer_license&custid=145
```

Example output:

```
-----BEGIN PDC Viewer LICENSE FOR: Rob (support@locklizard.com)-  
a2e2d0ea3c26a0608c4c2278b78c7047d716c0b9f25339fe46ed16d992ecbf11153484b6  
9cab9ffcb3f6b0872286acffc5f55e716a33ae15d17018ace47172107b7e5263810798838  
3afc24e73892920e20e3a829dfbd8ba837f2125d8ca2ac570d8f3c68e0a4cc90d09e8cb7  
63f67d7efceb53786f0c0f2f87fc98d8573dfdde70f5e63fc355b05405c927483f43cda42b  
a388244bc7ff9675af8ad2c5bd159bb89027a0ee48fa6a337e02a1d4313d1c63edbf1df8  
9f13eca5914d6232b71420a429616de8fd2429733716ca785530c  
-----END PDC Viewer LICENSE FOR: Rob (support@locklizard.com)---
```

Example use (for converting to a link):

```
http://www.server-name.com/ll-genlicense.php?pn=username&ur=a2e2d0ea3c26a0608c4c2278b78c7047d716c0b9f25339fe46ed16d992ecbf11153484b69cab9ffcb3f6b0872286acffc5f55e716a33ae15d17018ace47172107b7e52638107988383afc24e73892920e20e3a829dfbd8ba837f2125d8ca2ac570d8f3c68e0a4cc90d09e8cb763f67d7efceb53786f0c0f2f87fc98d8573dfdde70f5e63fc355b05405c927483f43cda42ba388244bc7ff9675af8ad2c5bd159bb89027a0ee48fa6a337e02a1d4313d1c63edbf1df89f13eca5914d6232b71420a429616de8fd2429733716ca785530c&fname=safeguard\_v\_license.llv
```

Where:

- server-name.com is replaced by the name of the Locklizard administration server;
- username is replaced by the user's email address;
- safeguard_v_license.llv is the name you want to call the license file. For example, you may want to replace this with the users name in the format firstname_surname.llv or just leave it as safeguard_v_license.llv for every user.

9.9 Request: Set customer Web Viewer access

This command is used to grant or deny access to the Web Viewer for an existing user or to change their Web Viewer login credentials.

Parameters:

action=set_customer_webviewer_access

- custid – id of the customer
- webviewer – set this to 1 to enable Web Viewer access or 0 to disable Web Viewer access.
- username – login username. If the user account had previously been granted Web Viewer access then this parameter is optional – if it is not entered then the previous login username will be used.
- password – login password. If the account had previously been granted Web Viewer access then this parameter is optional – if it is not entered then the previous login password will be used.
- noregemail – stops the sending of the registration email. If the administration system is set to send registration emails by default then you can use the value 1 to prevent sending of the registration email. For example, *&noregemail=1*.

If you want to change the login username and/or password for a Web Viewer account then you can use the set_customer_webviewer_access command to change those values accordingly.

Output:

If the request succeeds, then the first line of the output will be OK, followed by the username and password.

If the request fails the first line of the output will be Failed and on the next line a description of the error as a character string.

Example request 1:

To grant Web Viewer access to an existing customer use a request like this (let's assume the customer id is 145):

```
https://www.website.com/Interop.php?un=<replace with publisher
username>&pw=<replace with publisher password>&action=
set_customer_webviewer_access&custid=145&webviewer=1
&username=johndoe&password=abcdefg
```

Example output:

```
OK
"johndoe"
"abcdefg"
```

Example request 2:

To change the Web Viewer login password for an existing customer that has Web Viewer access, use a request like this (let's assume the customer id is 145):

```
https://www.website.com/Interop.php?un=<replace with publisher
username>&pw=<replace with publisher password>&action=
set_customer_webviewer_access&custid=145&webviewer=1
&username=johndoe&password=hijklmnop
```

Example output:

```
OK
"johndoe"
"hijklmnop"
```

Example request 3:

To disable Web Viewer access for an existing customer use a request like this (let's assume the customer id is 145):

```
https://www.website.com/Interop.php?un=<replace with publisher
username>&pw=<replace with publisher password>&action=
set_customer_webviewer_access&custid=145&webviewer=0
```

Example output:

OK

Example request 4:

To re-enable Web Viewer access for an existing customer and prevent re-sending of their login info use a request like this (let's assume the customer id is 145):

https://www.website.com/Interop.php?un=<replace with publisher username>&pw=<replace with publisher password>&action=set_customer_webviewer_access&custid=145&webviewer=1&noregemail=1

Example output:

OK
"johndoe"
"hijklmnop"

9.10 Request: Get customer Web Viewer access

This command is used to get the Web Viewer access status for an existing user along with their Web Viewer login credentials.

Parameters:

action=get_customer_webviewer_access

- custid – id of the customer

Output:

If the request succeeds, then the first line of the output will be OK, followed by the number 1 if access is enabled and the username and password, or 0 if access is disabled.

If the request fails the first line of the output will be Failed and on the next line a description of the error as a character string.

Example request 1:

To get the Web Viewer access status for an existing customer use a request like this (let's assume the customer id is 145):

```
https://www.website.com/Interop.php?un=<replace with publisher  
username>&pw=<replace with publisher password>&action=  
get_customer_webviewer_access&custid=145
```

Example output 1 (user who has Web Viewer access):

```
OK  
1  
"johndoe"  
"hjklmnop"
```

Example output 2 (user who does NOT have Web Viewer access):

```
OK  
0  
""  
""
```

9.11 Request: Update customer views

This command is used to update the number of views the customer has available. The number you enter adds to the existing value, so for example, if a customer has 10 views left on their account and you use this command with the value '5' then the customer account will now have 15 views available.

Parameters:

action=update_views

- custid – id of the customer
- docid – id of the documents
- views – number of views available

Output:

If the request succeeds, then the output will be OK.

If the request fails the first line of the output will be Failed and on the next line a description of the error as a character string.

Example request:

To add 10 more views to the customer account, use a request like this (let's assume the customer id is 145 and the document id is 6335):

```
https://www.website.com/Interop.php?un=<replace with publisher  
username>&pw=<replace with publisher  
password>&action=update_views&custid=145&docid=6335&views=10
```

9.12 Request: Update customer prints

This command is used to update the number of prints the customer has available. The number you enter adds to the existing value, so for example, if a customer has 10 prints left on their account and you use this command with the value '5' then the customer account will now have 15 prints available. If you use a negative number then prints will be removed from the existing value.

Parameters:

action=update_prints

- custid – id of the customer
- docid – id of the documents
- prints – number of prints available

Output:

If the request succeeds, then the output will be OK.

If the request fails the first line of the output will be Failed and on the next line a description of the error as a character string.

Example request (adding prints):

To ADD 10 more prints to a customer account, use a request like this (let's assume the customer id is 145 and the document id is 6335):

```
https://www.website.com/Interop.php?un=<replace with publisher  
username>&pw=<replace with publisher  
password>&action=update_prints&custid=145&docid=6335&prints=10
```

Example request (removing prints):

To REMOVE 10 prints from a customer account, use a request like this (let's assume the customer id is 145 and the document id is 6335):

```
https://www.website.com/Interop.php?un=<replace with publisher  
username>&pw=<replace with publisher  
password>&action=update_prints&custid=145&docid=6335&prints=-10
```

9.13 Request: List publications

This command is used to obtain the publication ids for all your publications.

If you are assigning customers to publications then you will need to know the publication ids.

Parameters:

action=list_publications

Output:

If the request succeeds, then the first line of the output will be OK, followed by the list of publications, one per line on each subsequent line. Publications are displayed in publication id order. For each publication the first part of the line is its id (unique to each publication) followed by one space, and then the name of the publication as a character string.

If the request fails the first line of the output will be Failed and on the next line a description of the error as a character string.

Example request:

```
https://www.website.com/Interop.php?un=<replace with publisher  
username>&pw=<replace with publisher password>&action=list_publications
```

Example output:

```
OK  
"702" "Forex"  
"943" "Finance Monthly"  
"944" "Hard up Times"  
"1139" "Cash News"  
"1832" "Special Promotions"
```

9.14 Request: List publication customers

This command is used to obtain a list of all the customers who have been given access to a specific publication.

Parameters:

action=list_publications_customers

Output:

If the request succeeds then the first line of the output will be OK, followed by the list of publication ids and the customer ids that can access them:

- publication id
- customer id

Results are displayed in publication id order. The publication id is displayed on the left and the customer id is displayed on the right. If a publication does not have any customers who can access it then the publication id is not displayed.

If the request fails the first line of the output will be Failed followed on the next line by the error description.

Example request:

```
https://www.website.com/Interop.php?un=<replace with publisher  
username>&pw=<replace with publisher  
password>&action=list_publications_customers
```

Example output:

```
OK  
"14" "375"  
"14" "376"  
"14" "238"  
"15" "376"  
"15" "391"  
"15" "393"
```

In the example output above, publication 14 can be accessed by customer ids 275, 376 and 238, and publication 15 can be accessed by customer ids 376, 391 and 393.

9.15 Request: Get publications count

This command is used to list the number of publications you have.

Parameters:

action=get_publications_count

Output:

If the request succeeds the output will be OK.

If the response is OK then the following line will contain the number of publications.

Example request 1:

To get the number of publications the request will be:

```
https://www.website.com/Interop.php?un=<replace with publisher  
username>&pw=<replace with publisher  
password>&action=get_publications_count
```

Example output:

```
OK  
9
```

9.16 Request: Add publication

This command is used to add a publication to the Administration system.

Parameters:

action=add_publication

- name – name of the publication
- description – description of the publication (e.g. monthly finance magazine)
- obeypubdate – whether the customer account start date is obeyed or not. Enter yes or no.

The publication name is mandatory, the description and the obeypubdate is optional. If you don't specify the obeypubdate parameter then the customer account start date will not be obeyed.

Output:

If the request succeeds the output will be OK followed by the publication ID.

If the request fails the first line of the output will be Failed followed on the next line by the error description.

Example request:

To add a publication with the publication name 'Finance Monthly' with the description 'monthly finance magazine', which obeys the customer account start date, use a request like this:

```
https://www.website.com/Interop.php?un=<replace with publisher  
username>&pw=<replace with publisher  
password>&action=add_publication&name=Finance Monthly  
&description=monthly finance magazine&obeypubdate=yes
```

Example output:

```
OK  
"5197"
```

9.17 Request: Grant publication access

This command is used to grant customer access to a publication. You can grant access for multiple customers to multiple publications.

You can also use this command to change the start and end date access to a publication which a customer already has access to – you just issue the `grant_publication_access` with the new start and end dates.

Parameters:

`action=grant_publication_access`

- `custid` - id(s) of the customer to which the access is granted. The parameter can be included multiple times when you want to grant more than one customer access.
- `publication` - id(s) of the publication to grant access to. The parameter can be included multiple times when you want to grant access to more than one publication
- `start_date` – the date from which you want to grant access to the publication (optional)
- `end_date` – the date from which you want to stop access to the publication (optional)

Output:

If the request succeeds the output will be OK.

If the request fails the first line of the output will be Failed followed on the next line by the error description.

Example request 1:

To grant a single customer access to two publications (let's assume there are 2 publications defined with ids 100 and 101) use a request like this:

```
https://www.website.com/Interop.php?un=<replace with publisher
username>&pw=<replace with publisher
password>&action=grant_publication_access&custid=id<replace with actual
customer id>&publication=100,101
```

Example request 2:

To grant two customers access to a single publication (let's assume there are two customers defined with ids of 145 and 153, and a publication defined with id 100) use a request like this:

```
https://www.website.com/Interop.php?un=<replace with publisher
username>&pw=<replace with publisher
password>&action=grant_publication_access&custid=145,153&publication=10
0,101
```

Example request 3:

To grant two customers access to two publications (let's assume there are two customers defined with ids of 145 and 153, and two publications defined with ids 100 and 101) use a request like this:

```
https://www.website.com/Interop.php?un=<replace with publisher
username>&pw=<replace with publisher
password>&action=grant_publication_access&custid=145,153&publication=10
0,101
```

Example request 4:

To grant a single customer limited access (1st May 2010 to 1st May 2011) to a publication (let's assume there is a customer defined with id 145, and a publication defined with id 100) use a request like this:

```
https://www.website.com/Interop.php?un=<replace with publisher
username>&pw=<replace with publisher
password>&action=grant_publication_access&custid=145&publication=100&st
art_date=05-01-2010&end_date=05-01-2011
```

Example request 5:

To change the publication access period for a customer that already has access to a publication (let's assume there is a customer defined with id 145, and a publication defined with id 100 that they already have access to, and you want to change the access period to 1st May 2010 - 1st May 2012), use a request like this:

```
https://www.website.com/Interop.php?un=<replace with publisher
username>&pw=<replace with publisher
password>&action=grant_publication_access&custid=145&publication=100&st
art_date=05-01-2010&end_date=05-01-2012
```

9.18 Request: List documents

This command is used to obtain the list of document ids matched to document titles.

The document id is used when allocating documents to customers and is essential for distinguishing documents since more than one document can exist with the same name on the administration system. Listing documents is also useful where several administrators are adding documents and it is necessary to automate the availability of these documents and their presentation on a web site.

Parameters:

action=list_documents

- weonly=1 – list only documents published for use with the Web Viewer.
- OR**
- pdonly=1 – list only documents not published for Web Viewer use.

Output:

If the request succeeds then the first line of the output will be OK, followed by the list of documents, one per line. For each document the following information (separated by spaces) will be returned:

- document id
- document title
- date and time when the document was published
- date and time when the document expires or never if it does not expire
- how the document was protected
 - for all customers = all
 - individually = none
 - to a publication = publication id

- whether the document was published for use with the Web (yes=true, no=false)

If request fails the first line of the output will be Failed and on the next line a description of the error as a character string.

Example request 1:

https://www.website.com/Interop.php?un=<replace with publisher username>&pw=<replace with publisher password>&action=list_documents

Example output:

OK

"158" "PDF Security" "04-06-2010 19:40:06" "never" "19" "true"

"135" "Flash Protection" "01-18-2010 20:42:05" "never" "all" "false"

"165" "Copyright Example" "04-08-2010 18:04:54" "never" "none" "true"

In the example output above, the document 'PDF Security' was protected to a publication (id=19) and for use with the Web Viewer, the document 'Flash Protection' was protected for all customers, and the document 'Copyright Example' was protected as a standalone document (i.e. for selected customers - must be individually allocated) and for use with the Web Viewer.

Example request 2:

To request the document list held by the Locklizard administration system for documents published for use with the Web Viewer only use a request like this:

https://www.website.com/Interop.php?un=<replace with publisher username>&pw=<replace with publisher password>&action=list_documents&webonly=1

9.19 Request: List publication documents

This command is used to obtain a list of all the documents in a publication.

It is used in order to determine all the documents available for use by people who have access to that publication. You should note that you may need to cross reference the user's start and stop dates against document start and stop dates in order to obtain a complete picture.

Parameters:

action=list_publication_documents

- pubid – id of the publication

Output:

If the request succeeds then the first line of the output will be OK, followed by the list of documents, one per line. For each document the following information (separated by spaces) will be returned:

- document id
- document title
- date and time when the document was published
- date and time when the document expires or never if it does not expire

Results are displayed in publication id order.

If the request fails the first line of the output will be Failed followed on the next line by the error description.

Example request:

```
https://www.website.com/Interop.php?un=<replace with publisher
username>&pw=<replace with publisher
password>&action=list_publication_documents&pubid=<replace with
publication id>
```

Example output:

OK

"16979" "IPR_and_the_Digital_Revolution" "07-26-2007 16:39:01" "never"

"16987" "OOF_Product_Data_Sheet" "07-26-2007 18:00:23" "never"

"16988" "flap" "07-26-2007 18:01:44" "never"

"17524" "Redundant_Guiding" "07-31-2007 18:13:56" "never"

"17525" "A_History_of_Rights_Management" "07-31-2007 18:15:45" "08-24-2007 18:15:45"

9.20 Request: List documents direct access

This command is used to obtain a list of all the documents that are not part of a publication or available automatically to all customers. It is used to provide you with the list of all documents that must be allocated individually to customers for them to have access.

Parameters:

action=list_documents_direct_access

Output:

If the request succeeds then the first line of the output will be OK, followed by the list of documents with each customer that can access them directly:

- document id
- customer id

Results are displayed in customer id order. The document id is displayed on the left and the customer id is displayed on the right.

If the request fails the first line of the output will be Failed followed on the next line by the error description.

Example request:

```
https://www.website.com/Interop.php?un=<replace with publisher
username>&pw=<replace with publisher
password>&action=list_documents_direct_access
```

Example output:

```
OK
"113" "238"
"162" "238"
"118" "238"
"119" "238"
"162" "242"
```

In the example output above, customer id 238 has access to documents 113, 162, 118, and 119. Customer id 242 has access to document 162.

9.21 Request: Get documents count

This command is used to list the number of documents you have.

Parameters:

action=get_documents_count

- weonly=1 – counts only those documents that have been published for use with the Web Viewer.

Output:

If the request succeeds the output will be OK.

If the response is OK then the following line will contain the number of customers.

Example request 1:

To get the number of documents the request will be:

```
https://www.website.com/Interop.php?un=<replace with publisher
username>&pw=<replace with publisher
password>&action=get_documents_count
```

Example output:

```
OK
9
```

Example request 1:

To get the number of documents who have been published for use with the Web Viewer the request will be:

```
https://www.website.com/Interop.php?un=<replace with publisher
username>&pw=<replace with publisher
password>&action=get_documents_count&weonly=1
```

9.22 Request: Grant document access

This command is used to grant customer access to a document. You can grant access for multiple customers to multiple document.

You can also use this command to change the start and end date access to a document which a customer already has access to – you just issue the `grant_document_access` with the new start and end dates.

Documents that have been published for all customers do not need to be added to customer records with this command. Documents published for access by all customers become available immediately without any intervention from the Publisher at all.

Parameters:

`action=grant_document_access`

- `custid` – id of the customer to which the access is granted. The customer id will have been returned to you as a result of using the Add customer command or the List customers command or you can look this information up on the administration system
- `docid` – id of the document to grant access to
- `access_type` – indicates if the access is limited or not. The possible values are unlimited (uses the expiry settings you chose when you protected the document) and limited. If the access is limited the following parameters are required too:
 - `start_date` – date from when the access is granted
 - `end_date` – date until the access is granted

Output:

If the request succeeds the output will be OK.

If the request fails the first line of the output will be Failed followed on the next line by the error description.

Example request 1:

To grant a customer access to a specific document for an unlimited period using the expiry settings you chose when you protected the document (let's assume the customer id is 145 and the document id is 5431), use a request like this:

```
https://www.website.com/Interop.php?un=<replace with publisher  
username>&pw=<replace with publisher  
password>&action=grant_document_access&custid=145&docid=5431&access_  
type=unlimited
```

Example request 2:

To grant two customers access to a specific document for an unlimited period: (let's assume the customer ids are 145 and 167 and the document id is 5431) use a request like this:

```
https://www.website.com/Interop.php?un=<replace with publisher  
username>&pw=<replace with publisher password>&action=grant_  
document_access&custid=145,167&docid=5431&access_type=unlimited
```

Example request 3:

To grant two customers access to two documents for a limited period from the 1st April 2010 to the 1st May 2010 (let's assume the customer ids are 145 and 167 and the document ids are 5431 and 7632), use a request like this:

```
https://www.website.com/Interop.php?un=<replace with publisher  
username>&pw=<replace with publisher password>&action=grant_  
document_access&custid=145,167&docid=5431,7632&access_type=limited&start_date=  
04-01-2010&end_date=05-01-2010
```

Example request 4:

To change the document access period for a customer that already has access to a document (let's assume there is a customer defined with id 145, and a document defined with id 5431 that they already have access to, and you want to change the access period to 1st May 2010 - 1st May 2012), use a request like this:

```
https://www.website.com/Interop.php?un=<replace with publisher  
username>&pw=<replace with publisher password>&action=grant_  
document  
_access&custid=145&docid=5431&access_type=limited&start_date=05-01-  
2010&end_date=05-01-2012
```

10. FAQs

This section covers frequently asked questions on using the ecommerce system.

10.1 What are the steps involved when a customer purchases a protected PDF document from my web site?

Your online ordering system would have a product field that matches a document id (number) and/or a publication id (number). A customer would:

- 1) Come to your site.
- 2) Go through your shopping cart system, place their order.
 - a. During the checkout process, after capturing billing information, you would send a request in the background to the Locklizard Administration server for a list of customers and check the email address to see if it is already registered - if so, you would just capture the customer id.
 - b. If the customer record does not already exist then you would use the `add_customer` command to add it.
 - c. After the credit card is successfully processed, you would send a request to the Locklizard Administration server to grant access to the document(s) for the customer.
 - d. In return, you would place the results in your database (i.e. the Customer ID assigned by the administration system would need to be part of your customer record in the event there was an issue).
- 3) Your customer would receive a receipt from you for their purchase, and an e-mail confirmation with their license key for the document/publication.

Because some customers may enter bogus or erroneous email addresses, and because the license email could get flagged as spam by some portion of your customers' servers, you may want to add a step to the above, wherein after purchasing, the web site makes a call to Locklizard Administration server and gets the license key(s), then display's them for download along with the

protected documents. This can be achieved using the `get_customer_license` command. You can tell your customers that "this license will also be emailed to you, but you may wish to download it now, and here's how to use it..." or some such.

4) You would provide the download link to your document, so that your customer could download it immediately. Once they have that, they need their licence key from you in order to open it. This might be a good place to also link to a download of the Locklizard viewer software.

5) Once the customer has the protected document and the license key they can then unlock and view the document.

In addition to the above steps, you will need to be able to handle errors which may occur if the administration server fails to return information for any reason (i.e. temporarily offline).

Repeat customers who have previously registered using the same email address, will not have to register with you again, so you will want to bear this in mind when displaying license information for a second time. For example you might want to say "As an existing customer you can view the documents you have just purchased without having to register again. Just download and double-click on the documents to open them".

Repeat customers who have used a different email address will be treated as a new customer by the administration system and they will need to register again with you using their new license file.

10.2 How can I add a customer and give them access to a document?

This is a two-step process. Firstly you must add a customer account using the command `add_customer` and then you must grant them access to a document. The document must already exist on the administration system and you must know its id (this information is displayed on the administration system and can also be retrieved using the ecommerce command `list_documents`).

1) STEP 1

Create a customer account. In the following example we shall create a customer account for the user John Doe with an email address of johndoe@yahoo.com, valid from 1st April 2008 with 1 license assigned to it:

```
https://www.website.com/Interop.php?un=<replace with publisher
username>&pw=<replace with publisher
password>&action=add_customer&name=John%20Doe&email=johndoe@yaho
o.com&start_date=04-01-2008&end_type=unlimited &licenses=1
```

If the customer account is created successfully then the system will respond with the text 'OK'. On the following line the unique customer id (an integer number) that has been allocated in the Locklizard administration system will be displayed. You will need to record this response for use in step 2.

2) STEP 2

Assign a document to the customer account. In the following example John Doe (whose customer id is 1234) is given access to a document which has the id 76842:

```
https://www.website.com/Interop.php?un=<replace with publisher
username>&pw=<replace with publisher
password>&action=grant_document_access&custid=1234&docid=76842&acce
ss_type=unlimited
```

10.3 How can I add a customer and give them access to a publication?

This action can be performed in a single step. If you do not know the publication id then you can find this out using the [List publications](#) command.

In the following example we shall create a customer account for the user John Doe with an email address of johndoe@yahoo.com, valid from 1st April 2008 with 1 license assigned to it and who has been granted access to a publication whose id is 100:

```
https://www.website.com/Interop.php?un=<replace with publisher
username>&pw=<replace with publisher
password>&action=add_customer&name=John%20Doe&email=johndoe@yaho
```

o.com&start_date=04-01-2008&end_type=unlimited&publication=100
&licenses=1

10.4 Do you provide sample code for integration into our shopping cart system?

No. We do however work with a third party, [Proexe](#), from who you can purchase the integration code, or you can let them do the integration for you.

10.5 How can I extend a customer's account for another year?

You need to use the [update customer account validity](#) command to change the customer's account end date.

10.6 How can I extend a customer's subscription (publication) access?

Assuming your customers are subscribing to publications, you need to use the [grant publication access](#) command to change the customer's access date for that publication.

Example request:

To change the publication access period for a customer that already has access to a publication (let's assume there is a customer defined with id 145, and a publication defined with id 100 that they already have access to, and you want to change the access period to 1st May 2010 - 1st May 2012), use a request like this:

```
https://www.website.com/Interop.php?un=<replace with publisher  
username>&pw=<replace with publisher  
password>&action=grant_publication_access&custid=145&publication=100&st  
art_date=05-01-2010&end_date=05-01-2012
```

11. Upgrading from Safeguard v2 to Safeguard v3

This section is relevant to Publishers who are upgrading from Safeguard version 2.x to Safeguard v3.x. It provides a summary of the differences in the formats of existing ecommerce commands between Safeguard v2 and Safeguard v3, as well as detailing new commands.

11.1 General changes

1. The Interop command used to initiate a call to the Locklizard Admin system has been replaced by Interop.php
2. Multiple publications are now separated by a comma - e.g. &publication=100,101
3. All fields returned in response to a command, apart from the status messages and the number count for customers, documents and publications, are now enclosed in quotes (").
4. The customer email address is now the unique identifier and not a combination of the name and email address.
5. When adding a customer, if the email address is the same as one on an existing customer account, then the response returned is "Failed. Account already exists. Customer id", where 'Customer id' is replaced by the id number of the customer account that already exists.
6. The sample shopping cart integration scripts use both the Carriage Return CHR(13) and Line Feed CHR(10) characters as a separation between lines. However, in Enterprise, the response for all commands is just a Line Feed CHR(10) character. For all commands, you should therefore change:

```
$resp = explode(chr(13).chr(10), $resp);
```

to

```
$resp = explode(chr(10), $resp);
```

11.2 New commands

- [list customer](#)
- [update customer views](#)
- [update customer prints](#)
- [get customers count](#)
- [get customer webviewer access](#)
- [set customer webviewer access](#)
- [get publications count](#)
- [get documents count](#)

11.3 Modified commands

1. [add customer](#) - if there is already an existing Administration record for a customer having the same **email address** then the customer record will be **updated** with the new start and end dates, allocated licenses (the value you enter will OVERWRITE the license count value set on the administration system), and additional publication access. A new optional parameter 'noregmail' is now supported that will prevent sending of the registration email regardless of the setting in your administration system.
2. [get customer license](#) - returns a different license file structure, and the OK response is no longer returned
3. [list customer](#) - additionally returns whether the account is enabled for use with the Web Viewer
4. [list customers](#) - additionally returns the registration status and whether the account is enabled for use with the Web Viewer
5. [grant publication access](#) - enables you to grant publication(s) access to multiple customers and specify the customer access period to the publication(s)
6. [grant document access](#) - enables you to grant multiple document access to a single or multiple customers

7. [add_publication](#) - enables you to state whether the customer account start date is obeyed or not
8. [list_publications](#) - shows results in publication id order
9. [list_publications_customers](#) - shows results in publication id order
10. [list_documents](#) - additionally returns how the document was protected (for all customers, individual access, or to a publication) and whether it was published for use with the Web Viewer