USER DOCUMENTATION(ALEPHINO 5.0)

Encryption with Alephino

ExLibris The bridge to knowledge

Author: Frank Bieber

Contents

1) Encryption of the Alephino C/S protocol 2
2) Exceptions 3
3) Encryption of the OPAC and the service module 3

Encryption in Alephino

The coming into force of the General Data Protection Regulation (GDPR) in the European Union raised the need for comprehensive measures to protect personal data in Alephino. The essential measure here is the encryption of the communication both between GUI clients and the Alephino server as well as the browser-based services OPAC and Web Services. If the Alephino database (server) and its applications are not operated exclusively in the local network, ie the services can be reached on the public Internet, encryption is essential.

1) Encryption of the Alephino C/S protocol

The Encryption Service Pack for Alephino provides two methods for encrypting communication between GUI clients and servers. The corresponding capabilities are also integrated into the installation packages available since the end of 2018. It should be noted that from the perspective of the clients, Alephino Server and Z39.50 Gateway are servers serving the same protocol. So that clients can use both services at the same time, the encryption must always be configured identically.

a) **„Ex Libris“**
This method was developed by Ex Libris and is already being used in Aleph and Alephino to encrypt staff account passwords. Now, the same algorithm is also available for encrypting the communication.

a. **Server side configuration**
Files : etc/alephino.cfg und etc/zgate.cfg
Parameter section : (Communication)
Parameter : Encryption = 1

b. **Client side configuration (GUI)**
File : alephcom/tab/alephcom.ini
Parameter section : [Main]
Parameter : Encryption = 1

b) **DES**
This method takes advantage of the Data Encryption Standard and is using a symmetric key that must be known to both server and client.

a. **Server side configuration**
Files : etc/alephino.cfg und etc/zgate.cfg
Parameter section : (Communication)
Parameter 1 : Encryption = 2
Parameter 2 : DESKey = ../etc/des_key.dat (Pathname of the key file)

File : etc/des_key.dat
Contents : Any string (numbers, letters), with only 7 characters (56 bits) being relevant.

b. **Client side configuration (GUI)**

File : alephcom/tab/alephcom.ini
Parameter section : [Main]
Parameter : Encryption = 2

File : alephcom/tab/des_key.dat
Contents being relevant. : Any string (numbers, letters), with only 7 characters (56 bits)

2) Exceptions

All components of Alephino communicate using the same client / server protocol. An Alephino database server not only serves the GUI clients, but also the web-based components of the system, such as the OPAC and the service module, as well as the servers for Z39.50 and Self-Check (SIP2). With the exception of the GUI, communication between the server and clients takes place exclusively at the local machine (localhost) or the intranet or VPN level. A Z39.50 or self-check server is unlikely to run on any machine other than its associated Alephino server, and if it does, it may not be connected to public IP addresses.

Thus, the internal server-server communication links require no encryption and have the ability to do so yet. In order for an encrypted Alephino database server to be able to continue to service such internal components as well, exceptions ***must*** be defined for these. This is done by setting IP addresses or address ranges.

Server side configuration

File	: etc/alephino.cfg	
Parameter section	: (IpfilterUNENCRYPTED)	
Parameter 1	: Allowed = 127.0.0.1	(localhost)
Parameter 2	: Allowed = 192.168.*.*	(local network)
…		
Parameter n	: Allowed = 10.180.*.*	(VPN)

Of course, this principle can also be applied to GUI clients, provided that they only communicate with their server in the local network, thus there is no danger of eavesdropping on the message exchange.

Note: Incorrect configurations of server and client encryption methods will always cause programs to fail. Unfortunately, there is no alternative, since neither the method used nor the key (understandably) may be exchanged openly. If the server was addressed with an inappropriate method, it will not respond any longer and must be restarted!

3) Encryption of the OPAC and the service module

To protect against eavesdropping on the browser-based message exchange, the HTTPS protocol is used instead of the unencrypted HTTP. The required configuration of asymmetric encryption with SSL / TLS is specific to the web server used, so strictly speaking it has nothing to do with Alephino,

so control tables or web pages in Alephino do not need to be changed.

Here is an example configuration of how this applies to the most used Apache web server. The prerequisite is that the web server supports SSL and there are no conflicts with services operated on the same server with regard to the ports used. Thus, the example assumes that a dedicated server is being used for Alephino.

```
#
# Basic SSL configuration
#
LoadModule ssl_module /usr/lib/apache2/modules/mod_ssl.so

<IfModule mod_ssl.c>

SSLRandomSeed startup builtin
SSLRandomSeed startup file:/dev/urandom 512
SSLRandomSeed connect builtin
SSLRandomSeed connect file:/dev/urandom 512
AddType application/x-x509-ca-cert .crt
AddType application/x-pkcs7-crl    .crl

SSLPassPhraseDialog  builtin

SSLSessionCache        shmcb:/var/run/apache2/ssl_scache(512000)
SSLSessionCacheTimeout  300

SSLMutex  file:/var/run/apache2/ssl_mutex

ALL:!ADH:!EXPORT56:RC4+RSA:+HIGH:+MEDIUM:+LOW:+SSLv2:+EXP:+eNULL
SSLProtocol all -SSLv2

</IfModule>
```

```
#
# ExLibris(D) GmbH
# Settings for Alephino 5.0
#

# Ports for Alephino services
Listen 80
<IfModule mod_ssl.c>
  Listen 443
</IfModule>

# Allow access to Alephino directories
<Directory  "/home/exlibris/alephino_50">
Require all granted
Options -Indexes
</Directory>

# Alephino Administration
<IfModule mod_ssl.c>
<VirtualHost *:443>
SSLEngine on
SSLCertificateFile "/home/exlibris/alephino_50/etc/server.cer"
SSLCertificateKeyFile "/home/exlibris/alephino_50/etc/server.key"
SSLCertificateChainFile  "/home/exlibris/alephino_50/etc/trustchain.cer"

AddDefaultCharset UTF-8
AddType application/x-Research-Info-Systems .ris
DocumentRoot "/home/exlibris/alephino_50"
Alias /german "/home/exlibris/alephino_50/htdoc/aliadm_ger"
Alias /english "/home/exlibris/alephino_50/htdoc/aliadm_eng"
Alias /wjhk.jupload.jar "/home/exlibris/alephino_50/bin/wjhk.jupload.jar"
Alias /download "/home/exlibris/alephino_50/temp"
Alias /pix "/home/exlibris/alephino_50/htdoc"
Alias /repository "/home/exlibris/alephino_50/data/objects"
ScriptAlias /alipac "/home/exlibris/alephino_50/bin/alipac"
ScriptAlias /alix "/home/exlibris/alephino_50/bin/alix"
ScriptAlias /aliadm "/home/exlibris/alephino_50/bin/aliadm"
ScriptAlias /aliadz "/home/exlibris/alephino_50/bin/aliadz"
ScriptAlias /upload "/home/exlibris/alephino_50/bin/upload"
ScriptAlias /cgistart "/home/exlibris/alephino_50/bin/cgistart"
ErrorLog "|/usr/sbin/rotatelogs /home/exlibris/alephino_50/weblog/https_error.%Y-%m-%d-%H_%M 10M"
TransferLog "|/usr/sbin/rotatelogs /home/exlibris/alephino_50/weblog/https_access.%Y-%m-%d-%H_%M 10M"
RedirectMatch ^/$ /alipac

# Authentication
<Location "/aliadm">
AuthType Basic
AuthName "Alephino Administration"
AuthUserFile "/home/exlibris/alephino_50/etc/passwords"
Require user alephino
</Location>

</VirtualHost>
</IfModule>

# Alephino OPAC
<VirtualHost *:80>
ServerName alephino.mylibrary.org
DocumentRoot "/home/exlibris/alephino_50"
Redirect permanent / https://alephino.mylibrary.org/
</VirtualHost>
```

For explanation:

● Private key and SSL certificate. Such may either be issued by a CA or self-signed. In the former case, the so-called chain of trust, ie the list of certificates issued by the CA, must be indicated.

● To call the OPAC, it is sufficient to enter the server name in the address bar of the browser.

● Since the service module allows unrestricted access to the Alephino database, this should be advantageously protected by an additional password query. In the example, the password of the user "alephino" is to be created or changed by means of "htpasswd".

● Accesses to the OPAC via unencrypted HTTP are thereby automatically redirected to HTTPS.