

EMS-Meldung - Beispielcode in PHP

Inhalt

1. Einleitung	1
2. Ablauf einer Meldung.....	1
3. Voraussetzungen.....	1
4. Inhalt des Pakets	2
5. Sourcecode.....	2
6. Beispiele für Antwort des Servers	4
6.1. OK.....	4
6.2. Fehler.....	4

1. Einleitung

Dieses Dokument beschreibt die Installation und Nutzung des Beispielcodes zum Absetzen von (Labor- bzw. Arzt-) Meldungen an das Epidemiologische Meldesystem „EMS“.

Rechtliche Grundlagen sind abrufbar unter

<https://www.sozialministerium.at/Themen/Gesundheit/Uebertragbare-Krankheiten/Rechtliches.html>

Technische Informationen finden sich unter

<https://www.sozialministerium.at/Themen/Gesundheit/Uebertragbare-Krankheiten/Rechtliches/COVID-19-Technisches-f%C3%BCr-Labor-und-%C3%84rzte.html>

2. Ablauf einer Meldung

- Erstellen der Meldung in XML-Format unter Verwendung von HL7
- Übermittlung an den Schnittstellenserver per https POST, die Authentifizierung erfolgt mittels Client-Zertifikat
- Auswertung der Antwort des Servers (Daten im Response-Header)

3. Voraussetzungen

PHP mit curl, Der Beispielcode wurde getestet mit

- „PHP 7.4.1“ und "curl 7.55.1 (Windows) libcurl/7.55.1 WinSSL"
- "PHP 7.3.27-1~deb10u1 (cli)" und "curl 7.64.0 (arm-unknown-linux-gnueabi) libcurl/7.64.0 OpenSSL/1.1.1d"

Anmerkung: Abhängig von der vorhandenen OpenSSL Version kann es in seltenen Fällen nötig sein, den Parameter „CipherString = DEFAULT@SECLEVEL“ in openssl.conf anzupassen.

4. Inhalt des Pakets

Das .zip File beinhaltet folgende Dateien:

sendToEMS.php	PHP Script zum Versenden der Meldung
EMSLabor999_npass.p12	PKCS#12 Clientzertifikat für Zugriff auf das EMS Testsystem
Beispielmeldung01.xml	Meldung in HL7

5. Sourcecode

```
<?php
/*
Beispielcode erstellt von Christian Baumann (f. BMSGPK)
5.3.2021

Umgebung:
* PHP mit curl:
* getestet mit "PHP 7.4.1" und "curl 7.55.1 (Windows) libcurl/7.55.1 WinSSL"
* getestet mit "PHP 7.3.27-1~deb10u1 (cli)" und "curl 7.64.0 (arm-unknown-linux-
gnewabi64) libcurl/7.64.0 OpenSSL/1.1.1d"
*/

$sampleXML = "Beispielmeldung01.xml";
$url = "https://ems-labor-test.ehealth.gv.at/hl7if/";
$clientCert = "EMSLabor999_npass.p12"; // Zertifikat fuer EMS Testumgebung
$responseHeaders = array();

// initialize curl & set options
$ch = curl_init();
curl_setopt($ch, CURLOPT_URL, $url);
curl_setopt($ch, CURLOPT_PORT, 443);
curl_setopt($ch, CURLOPT_VERBOSE, 0);
curl_setopt($ch, CURLOPT_HEADER, 0);
curl_setopt($ch, CURLOPT_SSLCERT, $clientCert);
curl_setopt($ch, CURLOPT_SSLCERTTYPE, 'P12');
curl_setopt($ch, CURLOPT_SSLCERTPASSWD, '');
curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, 1);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);

$postData = file_get_contents($sampleXML);
$headers = array();
$headers[] = 'Content-type: text/xml; charset=utf-8;';
$headers[] = 'Content-Length: ' . strlen($postData);
curl_setopt($ch, CURLOPT_POST, 1);
curl_setopt($ch, CURLOPT_POSTFIELDS, $postData);
curl_setopt($ch, CURLOPT_HTTPHEADER, $headers);
curl_setopt($ch, CURLOPT_HEADERFUNCTION, 'headerCallback');

$response = curl_exec($ch);

echo("---\r\n");
if (curl_errno($ch) > 0) {
    echo("curl error: " . curl_errno($ch) . ': ' . curl_error($ch) . "\r\n");
    die('');
} else {
    // echo("curl info: " . print_r(curl_getinfo($ch), true) . "\r\n");
}

curl_close($ch);

$responseCode = $responseHeaders['LIF_ResponseCode'];
```

```

$responseText = cleanText($responseHeaders['LIF_ResponseText']);
echo("responseCode: $responseCode - " . getResponseInfo($responseCode) . "\r\n");
echo("responseText: $responseText\r\n");

if (isSuccess($responseCode)) {
    $fallID = $responseHeaders['LIF_FallID'];
    $isNewFall = $responseHeaders['LIF_isNewFall'];
    echo("fallID: $fallID\r\n");
    echo("isNewFall: $isNewFall\r\n");
} else {

}

////////// helper functions

function headerCallback($ch, $headerLine)
{
    global $responseHeaders;
    $temp = trim($headerLine);
    $temp = explode(':', $temp, 2);
    $responseHeaders[$temp[0]] = @$temp[1];
    return strlen($headerLine);
}

function isSuccess($responseCode) {
    $responseCode = (integer) $responseCode;
    return(in_array($responseCode, array(0, 11, 14, 15, 16, 17, 18)));
}

function getResponseInfo($responseCode) {
    $responseCode = (integer) $responseCode;
    $responseInfo = array(
        0 => "SUCCESS", # Datenübermittlung erfolgreich
        1 => "OTHER", # Sonstiger Serverfehler (+Fehlerbeschreibung)
        2 => "XMLERROR", # Fehler in XML-Struktur (+Fehlerbeschreibung)
        3 => "NODATA", # Es wurden keine Daten übermittelt
        4 => "NOCERTIFICATE", # Kein Authentifizierungstoken vorhanden
        5 => "CERTNOTVALID", # Authentifizierungstoken ungültig (+Fehlerbeschreibung)
        8 => "CDAVALIDATION", # Fehler in CDA-Dokument/Schema (+Fehlerbeschreibung)
        9 => "NODISEASE", # Fehler in CDA-Dokument/Schema (+Fehlerbeschreibung)
        10 => "LIF_INTERNAL", # Fehler in CDA-Dokument/Schema (+Fehlerbeschreibung)
        11 => "NOTIFICATION_OVERWRITTEN", # Eine bestehende Arztmeldung wurde überschrieben
        12 => "NOGDAFOUND", # Es konnte kein entsprechender GDA in BAGDAD gefunden werden
        13 => "WRONGGDA", # Falscher GDA angegeben
        14 => "CASECREATED", # (Obsolet, Code wird nicht mehr verwendet)
        15 => "CASEMATCHED", # Meldung an bestehenden Fall angehängt
        16 => "CASECREATEDZMRFOUND", # Fall wurde angelegt, Patient in ZMR gefunden
        17 => "CASECREATEDERNPFOUND", # Fall wurde angelegt, Patient in ERNP gefunden
        18 => "CASECREATEDERNPINSERT", # Fall wurde angelegt, Patient in ERNP angelegt
    );
    if (isset($responseInfo[$responseCode])) {
        return($responseInfo[$responseCode]);
    } else {
        return('ERROR decoding ResponseInfo');
    }
}

function cleanText($text) {
    $res = str_replace(array('%0a', '&#252;', '&#228;'), array('', 'ü', 'ä'), $text);
    return($res);
}
?>

```

6. Beispiele für Antwort des Servers

6.1.OK

```
---  
responseCode: 15 - CASEMATCHED  
responseText: Datenübermittlung erfolgreich, Meldung an bestehenden Fall  
angehängt: Labormeldung54993_20210305-1138_Fall1486709.xml  
fallID: 486709  
isNewFall: False
```

6.2.Fehler

```
---  
responseCode: 5 - CERTNOTVALID  
responseText: Zertifikat ungültig! (Dem Zertifikat, mit dem Sie einsteigen  
(SerialNr=04e5), ist kein GDA zugeordnet, oder das Zertifikat wurde  
deaktiviert.)
```