



e-Tresor API

Datenerfassungsprotokoll

Inhaltsverzeichnis

1	Voraussetzungen für die Verwendung.....	4
1.1	Funktion aktivieren.....	4
1.1.1	e-Tresor Upgrade kaufen.....	4
1.1.2	Upgrade konfigurieren	5
2	Methoden.....	6
	Rückgabewerte des State Objekts.....	6
2.1	Add	7
2.1.1	Beschreibung.....	7
2.1.2	Parameter	7
2.1.3	Rückgabewerte.....	7
2.2	DeleteAllDEPEntriesForRegisterId.....	7
2.2.1	Beschreibung.....	7
2.2.2	Parameter	7
2.2.3	Rückgabewerte.....	7
2.3	DownloadDEPFile	8
2.3.1	Beschreibung.....	8
2.3.2	Parameter	8
2.3.3	Rückgabewerte.....	8
2.4	Export	9
2.4.1	Beschreibung.....	9
2.4.2	Parameter	9
2.4.3	Rückgabewerte.....	9
2.5	ReadRKSVMMetaData	10
2.5.1	Beschreibung.....	10
2.5.2	Parameter	10
2.5.3	Rückgabewerte.....	10
2.6	SaveRKSVMMetaData.....	10
2.6.1	Beschreibung.....	10
2.6.2	Parameter	11
2.6.3	Rückgabewerte.....	11
2.7	UploadDEPFile	12
2.7.1	Beschreibung.....	12

2.7.2	Parameter	12
2.7.3	Rückgabewerte	12
3	Webservice Beschreibung (SOAP)	13
3.1.1	SOAP 1.1	13
3.1.2	SOAP 1.2	14
4	Codebeispiele	15
4.1	Beispielaufruf in C#	15
5	Implementierung	16
5.1	Echtsystem	16
5.2	Testsystem	16
5.3	Aktivieren der Funktion e-Tresor DEP	16
5.3.1	Aktivieren im Testsystem	16
5.3.2	Status der Erweiterung	16
5.4	Anlegen eines DEP API Benutzers	17

1 Voraussetzungen für die Verwendung

Für die signierten Belegdaten Ihrer Registrierkasse benötigen Sie einen externen Speicher für das Datenerfassungsprotokoll (DEP). Nach der RKS-V müssen die Daten vierteljährlich auf einem elektronischen externen Medium unveränderbar gesichert werden. Diese Sicherung ist gemäß § 132 BAO aufzubewahren.

Der e-Tresor von A-Trust erfüllt alle gesetzlichen Anforderungen an eine revisionssichere Archivierung.

1.1 Funktion aktivieren

1.1.1 e-Tresor Upgrade kaufen

Um das Datenerfassungsprotokoll im e-Tresor ablegen zu können, muss die entsprechende Funktion erst aktiviert werden. Das geht ganz einfach, indem Sie sich für ein gratis e-Tresor Konto registrieren und nach dem Login auf den Ordner-Menüpunkt 'Datenerfassungsprotokoll' klicken. Dort kann die Funktion durch Angabe eines zuvor erhaltenen Aktivierungscodes aktiviert werden.

Aktivierungscodes können über den A-Trust Webshop gekauft werden:

<https://www.a-trust.at/webshop/Detail.aspx?ProdId=3030>

Die Funktion e-Tresor DEP ist grundsätzlich für ein Jahr aktiv und muss danach wieder neu aktiviert werden. Sollten Sie eine gültige E-Mail Adresse unter ‚Mehr‘ - ‚Einstellungen‘ - ‚Kontaktadressen‘ hinterlegt haben, so werden Sie ein Monat vor Ablauf automatisch verständigt.

Tips für Partner:

Wenn Sie direkt über den A-Trust Webshop und Ihr Partner Login die Funktion e-Tresor DEP erwerben, erhalten Sie Aktivierungscodes, die Sie an Ihre Endkunden weitergeben können, oder die Funktion komplett für diese einrichten können.

Bitte beachten Sie, dass pro Aktivierungscode nur das Datenerfassungsprotokoll einer Kasse abgelegt werden kann. Pro aktiviertem Upgrade kann also das DEP einer KassenId abgelegt werden. Sollte sich die KassenId ändern, muss zuerst das DEP der alten KassenId gelöscht werden, um ohne zusätzliches Upgrade ein neues DEP (neue KassenId) abzulegen.

1.1.2 Upgrade konfigurieren

In weiterer Folge muss über die Einstellungen ein DEP Webservice Benutzer angelegt werden, mit dem Ihre Kassenlösung die Daten des Datenerfassungsprotokolls im e-Tresor ablegen kann.

Um diesen Benutzer zu erstellen, melden Sie sich an einem e-Tresor Konto an. Nach dem Login kann über den Menüpunkt 'Mehr', 'Einstellungen', 'Zugänge/Logins' ein Datenerfassungsprotokoll Webservice Benutzer Login angelegt werden. Diese Zugangsdaten benötigt Ihre Registrierkasse, um die Daten des DEPs in Ihrem e-Tresor abzulegen.

Um die einzelnen Belegzeilen des Datenerfassungsprotokolles Ihrer Registrierkasse im e-Tresor abzulegen, müssen diese per Webservice - Schnittstelle in Ihr e-Tresor Konto hochgeladen werden. Sollte Ihre Registrierkasse die Anbindung an die e-Tresor-Datenerfassungsprotokoll-API bereits unterstützen, benötigen Sie die Zugangsdaten des zuvor angelegten e-Tresor DEP Webservice Benutzers, welcher in Ihrer Registrierkasse hinterlegt werden muss.

Sollten Sie selbst Registrierkassen Hersteller sein, finden Sie in diesem Dokument alle nötigen Informationen für die Verwendung der API Schnittstelle.

2 Methoden

Die e-Tresor DEP Webservice Schnittstelle bietet die folgenden Funktionen an:

- Add: Hinzufügen von Belegzeilen zum Datenerfassungsprotokoll
- DeleteAllDEPEntriesForRegisterId: Alle Einträge einer KassenId löschen
- DownloadDEPFile: zuvor hochgeladenes DEP File herunterladen
- Export: Abruf von vorhandenen Daten des Datenerfassungsprotokolls
- ReadRKSVMetaData: RKSV Metadaten der Kasse auslesen
- (Register: FÜR INTERNEN GEBRAUCH)
- SaveRKSVMetaData: RKSV Metadaten der Kasse ablegen
- UploadDEPFile: Ein DEP File hochladen

Bitte beachten Sie, dass Funktionen zur Ablage neuer Daten nur solange zur Verfügung steht, wie die DEP Funktion gebucht wurde.

Die Export/Download Funktionen stehen dabei uneingeschränkt zur Verfügung.

Rückgabewerte des State Objekts

State kann die folgenden Werte haben (gilt für alle Webservice Calls):

- **AuthenticationError:** Tritt bei ungültigen Credentials auf
- **FileAlreadyExists:** Eine Datei mit dem Dateinamen existiert bereits
- **FunctionError:** Ein funktionsspezifischer Fehler ist aufgetreten
- **InternalError:** Tritt bei serverinternen Fehlern auf
- **InvalidDEPFileFormat:** ungültiges DEP Export Format der Datei oder keine DEP Einträge in der Datei gefunden
- **MaintenanceActive:** Das Service ist im Wartungsmodus
- **NotFound:** Objekt konnte nicht gefunden werden (zB Datei)
- **ReceiptNumberAlreadyExists:** Ein DEP Eintrag mit der KassenId/Belegnummer existiert bereits
- **RegisterIdMismatch:** Es wurde versucht mehrere DEPs abzulegen als DEP Upgrades aktiviert wurden (ein DEP Upgrade = eine KassenId)
- **RequiredParameterNotSet:** Ein erforderlicher Parameter wurde nicht angegeben
- **Success:** Verarbeitung erfolgreich
- **Unprocessed:** Verarbeitung wurde nicht durchgeführt
- **UpgradeInactive:** Das DEP Upgrade ist für dieses Konto nicht aktiv

2.1 Add

2.1.1 Beschreibung

Mit diesem Aufruf wird ein neuer Eintrag des Datenerfassungsprotokolls im zugehörigen Konto des e-Tresor Benutzers abgelegt.

2.1.2 Parameter

`Result Add(NetworkCredential credential, string entry, string signedEntry)`

Parameter	Beschreibung
credential	Die Textlogin-Zugangsdaten (Domain wird nicht verwendet)
entry	Der Eintrag des Datenerfassungsprotokolls
signedEntry	Der signierte Eintrag

2.1.3 Rückgabewerte

```
Result
{
    Data // wird nicht verwendet
    Description // Beschreibung des Ergebnisses
    State // Status, ob die Verarbeitung erfolgreich war etc.
    DataType // wird nicht verwendet
}
```

2.2 DeleteAllDEPEntriesForRegisterId

2.2.1 Beschreibung

Mit diesem Aufruf werden alle Einträge des Datenerfassungsprotokolls für eine bestimmte KassenId gelöscht.

2.2.2 Parameter

`Result Add(NetworkCredential credential, string registerId)`

Parameter	Beschreibung
credential	Die Textlogin-Zugangsdaten (Domain wird nicht verwendet)
registerId	Die KassenId des Datenerfassungsprotokolls

2.2.3 Rückgabewerte

```
Result
{
    Data // wird nicht verwendet
    Description // Beschreibung des Ergebnisses
    State // Status, ob die Verarbeitung erfolgreich war etc.
    DataType // wird nicht verwendet
}
```

2.3 DownloadDEPFile

2.3.1 Beschreibung

Mit dieser Funktion können zuvor über UploadDEPFile hochgeladene Dateien des Datenerfassungsprotokolls heruntergeladen (exportiert) werden.

zu beachten:

- Der Dateiname muss mit dem des Uploads übereinstimmen.
- Es kann nicht nach Datumsgrenzen exportiert werden.

2.3.2 Parameter

`Result DownloadDEPFile(NetworkCredential credential, string fileName)`

Parameter	Beschreibung
credential	Die Textlogin-Zugangsdaten (Domain wird nicht verwendet)
fileName	Name der DEP Datei

2.3.3 Rückgabewerte

```
Result
{
    Data // Dateidaten
    Description // Beschreibung des Ergebnisses
    State // Status, ob die Verarbeitung erfolgreich war etc.
    DataType // byte[]
}
```


2.4 Export

2.4.1 Beschreibung

Mit dieser Funktion können im e-Tresor abgelegte DEP Einträge für einen bestimmten Zeitraum im json-Export Format abgeholt werden.

2.4.2 Parameter

`Result` Export(`NetworkCredential` credential, `DateTime` startDate, `DateTime` endDate)

Parameter	Beschreibung
credential	Die Textlogin-Zugangsdaten (Domain wird nicht verwendet)
startDate	Startdatum Export Datenerfassungsprotokolls
endDate	Enddatum Export Datenerfassungsprotokolls

2.4.3 Rückgabewerte

```
Result
{
    Data // DEP Export als JSON String
    Description // Beschreibung des Ergebnisses
    State // Status, ob die Verarbeitung erfolgreich war etc.
    DataType // string
}
```

2.5 ReadRKSVMetaData

2.5.1 Beschreibung

Mit dieser Funktion können über SaveRKSVMetaData hinterlegte Metadaten ausgelesen werden.

2.5.2 Parameter

`Result ReadRKSVMetaData(NetworkCredential credential, string registerId)`

Parameter	Beschreibung
<code>credential</code>	Die Textlogin-Zugangsdaten (Domain wird nicht verwendet)
<code>registerId</code>	KassenId der zu lesenden MetaDaten

2.5.3 Rückgabewerte

```
Result
{
    Data // RKSVMetaData Object
    Description // Beschreibung des Ergebnisses
    State // Status, ob die Verarbeitung erfolgreich war etc.
    DataType // RKSVMetaData
}
```

2.6 SaveRKSVMetaData

2.6.1 Beschreibung

Mit dieser Funktion können die Kassenspezifischen Daten im e-Tresor abgelegt werden. Diese umfassen folgende Werte:

- | | |
|--|-----------------------------------|
| - KassenId | <i>RegisterId</i> |
| - AES Schlüssel | <i>Base64AESKey</i> |
| - Signatur Algorithmus | <i>Algorithmus</i> |
| - Zertifikat | <i>Certificate</i> |
| - Zertifikatsseriennummer im HEX Format | <i>CertificateSerialNumberHEX</i> |
| - ATU oder Steuernummer des Unternehmens | <i>CompanyIdentifizier</i> |
| - Zertifizierungsdiensteanbieterkennung | <i>ZDAID</i> |

zu beachten:

- bei einem erneuten Upload der Metadaten mit der selben KassenId werden bereits vorhandene Metadaten überschrieben
- Umsatzzähler der einzeln nochgeladenen Belegzeilen kann nur angezeigt werden, wenn diese Metadaten vorhanden sind.

2.6.2 Parameter

Result SaveRKSVMetaData(**NetworkCredential** credential, **RKSVMetaData** metaData)

Parameter	Beschreibung
credential	Die Textlogin-Zugangsdaten (Domain wird nicht verwendet)
metaData	MetaData Objekt

```
<metaData>
  <RegisterId>string</RegisterId>
  <Base64AESKey>string</Base64AESKey>
  <Algorithmus>string</Algorithmus>
  <Certificate>string</Certificate>
  <CertificateSerialNumberHEX>string</CertificateSerialNumberHEX>
  <CompanyIdentifier>string</CompanyIdentifier>
  <ZDAID>string</ZDAID>
</metaData>
```

2.6.3 Rückgabewerte

```
Result
{
    Data wird nicht verwendet
    Description // Beschreibung des Ergebnisses
    State // Status, ob die Verarbeitung erfolgreich war etc.
    DataType wird nicht verwendet
}
```

2.7 UploadDEPFile

2.7.1 Beschreibung

Mit dieser Funktion kann das Datenerfassungsprotokoll als Datei hochgeladen werden. Es müssen also nicht einzelne Belegzeilen wie bei der **Add** Funktion hochgeladen werden, sondern es können gesammelte Einträge des Journals asynchron (zb vierteljährlich) hochgeladen werden.

2.7.2 Parameter

`Result UploadDEPFile(NetworkCredential credential, string fileName, byte[] fileData)`

Parameter	Beschreibung
credential	Die Textlogin-Zugangsdaten (Domain wird nicht verwendet)
fileName	Name der DEP Datei
fileData	Dateidaten

2.7.3 Rückgabewerte

```
Result
{
    Data wird nicht verwendet
    Description // Beschreibung des Ergebnisses
    State // Status, ob die Verarbeitung erfolgreich war etc.
    DataType wird nicht verwendet
}
```

3 Webservice Beschreibung (SOAP)

3.1.1 SOAP 1.1

Es folgt ein Beispiel für eine SOAP 1.1-Anforderung und -Antwort. Die angezeigten Platzhalter müssen durch tatsächliche Werte ersetzt werden.

```
POST /rksv_dep.asmx HTTP/1.1
Host: localhost
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "https://www.e-tresor.at/Add"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <Add xmlns="https://www.e-tresor.at/">
      <credential>
        <UserName>string</UserName>
        <Password>string</Password>
        <SecurePassword />
        <Domain>string</Domain>
      </credential>
      <entry>string</entry>
      <signedEntry>string</signedEntry>
    </Add>
  </soap:Body>
</soap:Envelope>
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <AddResponse xmlns="https://www.e-tresor.at/">
      <AddResult>
        <Data />
        <DataType>string</DataType>
        <Description>string</Description>
        <ResultState>Unprocessed or AuthenticationError or FunctionError or
InternalError or MaintenanceActive or Success or NotFound</ResultState>
      </AddResult>
    </AddResponse>
  </soap:Body>
</soap:Envelope>
```

3.1.2 SOAP 1.2

Es folgt ein Beispiel für eine SOAP 1.2-Anforderung und -Antwort. Die angezeigten Platzhalter müssen durch tatsächliche Werte ersetzt werden.

```
POST /rksv_dep.asmx HTTP/1.1
Host: localhost
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <Add xmlns="https://www.e-tresor.at/">
      <credential>
        <UserName>string</UserName>
        <Password>string</Password>
        <SecurePassword />
        <Domain>string</Domain>
      </credential>
      <entry>string</entry>
      <signedEntry>string</signedEntry>
    </Add>
  </soap12:Body>
</soap12:Envelope>
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <AddResponse xmlns="https://www.e-tresor.at/">
      <AddResult>
        <Data />
        <DataType>string</DataType>
        <Description>string</Description>
        <ResultState>Unprocessed or AuthenticationError or FunctionError or
InternalError or MaintenanceActive or Success or NotFound</ResultState>
      </AddResult>
    </AddResponse>
  </soap12:Body>
</soap12:Envelope>
```

4 Codebeispiele

4.1 Beispielaufruf in C#

```
string rksvDepEntry =
    "_R1-AT1_RK01_BE00001_2016-03-18T12:11:56_0,00_0,00_0,00_0,00_0,00_kz4BTVXev9PbZvETwKrG0g==_2030F1FE_13DHEvvdXY8=";

string signedRksvDepEntry =
    "eyJhbGciOiJFUzI1NiJ9.X1IxLUFUMV9SSzAxX0JFMdAwMDFfMjAxNi0wMy0xOFQxMjoxMTo1Nl8wLDAwXzAsMdBfMCMwMF8wLDAwXzAsMDBfa3o0QlRWWGV2OVBiWnZlVHdLckdPZz09XzIwMzBGMUZFX2wzREhFdnZkWFk4PQ.jAzpTZ_hZqWwLJ7k8o1iAIgn5vULiYEtP48VfrlgEM_9w_LeYnYYwvuVbS_d2pt4YYDnpBjTNkUvz2q8nG4CtA";

RKSVAPl.rksv_depSoapClient rksvApiClient = new RKSVAPl.rksv_depSoapClient();

RKSVAPl.NetworkCredential c = new RKSVAPl.NetworkCredential();
c.UserName = "demo";
c.Password = "demo";

RKSVAPl.Result result = rksvApiClient.Add(c, rksvDepEntry, signedRksvDepEntry);

Console.WriteLine(result.ResultState.ToString());
```

5 Implementierung

5.1 Echtsystem

Die e-Tresor Anwendungadresse lautet:

<https://www.e-tresor.at>

Die e-Tresor DEP API Schnittstellenadresse lautet:

https://www.e-tresor.at/api.et/rksv_dep.aspx

5.2 Testsystem

A-Trust stellt Registrierkassen Herstellern ein Testsystem zur Implementierung der e-Tresor DEP Schnittstelle zur Verfügung.

Die e-Tresor Anwendungadresse im Testsystem lautet:

<https://test.e-tresor.at/web/>

Die e-Tresor DEP API Schnittstellenadresse des Testsystems lautet:

https://test.e-tresor.at/api/rksv_dep.aspx

5.3 Aktivieren der Funktion e-Tresor DEP

Um die Funktion zu aktivieren, benötigen Sie zuerst ein gratis e-Tresor Konto, welches über die Startseite angelegt werden kann.

Unter dem Menüpunkt ‚Datenerfassungsprotokoll‘ kann die Funktion dann auf zwei Arten aktiviert werden:

- Mehr -> Store -> **e-Tresor CheckoutDEPApi** kaufen
- Angabe eine Aktivierungscode (erhältlich im A-Trust Webshop)

5.3.1 Aktivieren im Testsystem

Um die Funktion im Testsystem zu aktivieren geben Sie bitte den Aktivierungscode **12345** ein.

5.3.2 Status der Erweiterung

Der aktuelle Status der e-Tresor DEP Funktion wird unter ‚Mehr‘ – ‚Meine Daten‘ unter ‚Konto Erweiterung‘ angezeigt.

5.4 Anlegen eines DEP API Benutzers

Um die e-Tresor DEP API Schnittstelle zu nutzen, muss ein DEP API Webservice Benutzer über die e-Tresor Oberfläche angelegt werden.

- Auf der e-Tresor Startseite ein neues e-Tresor Konto registrieren. An die angegebene Emailadresse werden die Textlogin Zugangsdaten des neuen e-Tresor Kontos geschickt. ACHTUNG Diese Zugangsdaten sind nicht für die API Schnittstelle.
- Mit den Textlogin Zugangsdaten am e-Tresor Konto anmelden
- Unter dem Menüpunkt „**Mehr->Einstellungen->Zugänge/Logins**“ auf „**Loginmöglichkeit hinzufügen**“ und ein neues „**DEP-API-Login**“ anlegen.
- Dieses Login kann nun für die Verwendung der Schnittstelle (Credentials Username/Passord) verwendet werden