



# USING THE LTSS-DDA PROVIDER UPLOAD API

## Version History

Version	Date	Changes
Final V1.0	10/25/2019	
Final V2.0	01/08/2020	
Final V2.1	5/26/2020	
Final V4.0	7/23/2020	Updated to SSO v4
Final V4.1	10/26/2020	Changed the URL to health.maryland.gov Updated test commands for Windows to set InvokeRestMethod
Final V4.2	6/28/2022	Added NumberOfPeopleAuthorized field information
Final V5.0	2/1/2024	Updated Production URLs to match production (no action necessary for existing users) Added Appendix: Service Identifier Mapping table

## Overview

The provider upload API is the primary way for the provider system to create Billing Entries in Provider Portal platform. It's an HTTP based API that different provider apps and system can use it programmatically to POST Billing Entries.

## Using the Provider Upload API

### HTTP/1.1

All data transfer conform to HTTP/1.1, and all endpoints require HTTPS.

### Production URLs

- URL for authentication:  
[https://access.mymdthink.maryland.gov:443/mdtssso/oauth2/realms/root/realms/ltssselfservice/access\\_token](https://access.mymdthink.maryland.gov:443/mdtssso/oauth2/realms/root/realms/ltssselfservice/access_token)
- URL for sending data:  
<https://ltss.health.maryland.gov/ltssv2/Ltss.SelfServeApi.Web/api/providerupload/uploadbillingentry>

## Authentication

Authentication allows provider systems to create Billing Entry using Provider Upload API. Authentication also allow us to identify the provider system, the type of data being transmitted. Provider Upload API requires to have a JWT (JSON Web Token) each time you access the upload billing entry endpoint.

Authentication is the first request a system should make before begin their upload process.

Please request MDH to provide following provider specific credentials before using Provider Upload API

- AppId
- AppSecret
- SSO Username

**Note:** SSO accounts for accessing provider upload API has different role and provider cannot use the same account for accessing Provider Portal and vice-versa.

### Authentication Endpoint

	Key	Value	Comments
<b>Headers</b>			
	Authorization	Basic {Base64Encoding({appId}:{appSecret}) <b>**See Note Below</b>	appId and appSecret will be issued to each provider by MDH. The header value must be base64 encoded
	cache-control	no-cache	
	content-type	application/x-www-form-urlencoded	
<b>Post Body : x-www-form-urlencoded</b>			

	grant_type	password	Static value
	Username	{sso_username}	SSO Username will be provided by MDH
	password	{sso_password}	Providers will have the option reset their password.

**\*\*Basic {Base64Encoding({appId}:{appSecret})}**

Basic e2FwcElkfTp7YXBwU2VjcmV0fQ==

How to base64 encode the username and password:

**Windows:**

```
[Convert]::ToBase64String([System.Text.Encoding]::ASCII.GetBytes('{appId}:{appSecret}'))
```

Where the output of the command is

e2FwcElkfTp7YXBwU2VjcmV0fQ==

**Linux:**

```
echo -n '{appId}:{appSecret}' | base64 -w9999
```

Where the output of the command is

e2FwcElkfTp7YXBwU2VjcmV0fQ==

## Sample Requests:

**\*\*NOTE\*\*** - The screenshots below are examples that are calling a specific test environment. Please use the appropriate URL listed above based on the environment you are connecting.

### Linux

Passing all options in as a request body

```
curl -X POST \  
https://provtestssoltss.health.maryland.gov/connect/token \  
-H 'Authorization: Basic e2FwcElkfTp7YXBwU2VjcmV0fQ==' \  
-H 'Content-Type: application/x-www-form-urlencoded' \  
-H 'cache-control: no-cache' \  
-d 'grant_type=password&username={{sso_username}}&password={{sso_password}}'
```

Alternately, using the Forms options of curl:

```
curl -X POST \  
https://provtestssoltss.health.maryland.gov/connect/token \  
-H 'Authorization: Basic e2FwcElkfTp7YXBwU2VjcmV0fQ==' \  
-H 'cache-control: no-cache' \  
-F grant_type=password \  
-F username={{sso_username}} \  
-F password={{sso_password}}
```

*Note: The above sample request is for illustration purpose only. Every programming language will need to build the request in its native implementation.*

## Windows

```
$headers = @{
    "Cache-Control" = "no-cache"
    "Authorization" = "Basic e2FwcElkfTp7YXBwU2VjcmV0fQ=="
}
$form = @{
    "username" = "{{sso_username}}"
    "password" = "{{sso_password}}"
    "grant_type" = "password"
}
Invoke-RestMethod 'https://provtestssoltss.health.maryland.gov/connect/token' -Method 'POST' -
Headers $headers -Body $form | ConvertTo-Json
```

## Authentication Response

Success		
	HTTP Response Code	200
Response Body - Sample		
<pre>{   "access_token": "{TOKEN_STRING}",   "expires_in": {TOKEN_EXPIRATION_IN_SECONDS},   "token_type": "Bearer",   "scope": "{TOKEN_SCOPE}" }</pre>		
Fail		
Unauthorized	HTTP Response Code	401
{"error": "invalid_scope"}	Scopes Configured for Client do not allow token issuance	
{"error": "invalid_client"}	Authorization contents are invalid	

## Authentication Response Body Attributes

Response Body		
	access_token	<b>JWT token string. This token string needs to be used in every request for the upload process.</b>
	expires_in	Token expiration time in seconds. Default value is 3600 seconds
	token_type	Specifies which token type is being returned. Token_type does not need to be used in subsequent calls.

		Token type would always be <b>Bearer</b>
	scope	The scope in which this token is valid

## Upload Billing Entry

After successful authentication, provider system can access provider upload web API resource to upload billing entries one at a time. A single billing entry can be uploaded at a time. The Web API is designed to keep the small pay load for better throughput.

### Idempotent Design

Upload billing entry is idempotent by design. Each billing entry requires a unique transaction id. If for some reason the client experience an exception such as timeout, connection closed etc. Client system should use the same transaction Id for the retry call. This will avoid duplicate Billing Entry in provider portal system. You may reuse the same transaction Id when request fails due to validation error.

### Upload Billing Entry Endpoint

<https://ltss.health.maryland.gov/ltssv2/Ltss.SelfServeApi.Web/api/providerupload/uploadbillingentry>

### Upload Billing Entry Request Data Format

Field	JSON Type	Required	Description
TransactionId	String	Yes	A unique transactionId for every request. A 128 bit UUID/GUID to avoid collision. Example: 80a4889c-6f1b-453d-a306-eccfb9382355 (From <a href="https://www.guidgenerator.com/online-guid-generator.aspx">https://www.guidgenerator.com/online-guid-generator.aspx</a> )
ServiceIdentifier	string	Yes	Alpha-numeric Abbreviated service names/codes titles will be provided by DDA.
ServiceDate	string	Yes	Valid Date of Service Not greater than current date Not older than 365 days Should be greater than client's pilot date (pilot phase only) mm-dd-yyyy format
ClientLtssId	string	Yes	Alpha-numeric Valid Client's LTSS ID
ProviderMa	string	Yes	Alpha-numeric Valid Provider MA Number
Units	number	Yes	Required if Cost is not specified

Field	JSON Type	Required	Description
			Number of Units. A whole number No decimal is allowed
Cost	number	Yes	Required if Unit is not specified Cost of Activity up-to 2 decimal point
NumberOfPeopleAuthorized	number	Yes	Required when ServiceIdentifier is one of the following: <ul style="list-style-type: none"> <li>● CLES</li> <li>● CLESR</li> <li>● CLEST</li> <li>● CLGH</li> <li>● CLGHR</li> <li>● CLGHT</li> <li>● SUPLIV</li> </ul> Value should be null for other ServiceIdentifiers.  When a numeric value is given, it must be between 1 and 8, inclusive (i.e., $1 \leq x \leq 8$ ).

### Data Contract for Upload Request

	Key	Value	Comments
<b>Headers</b>			
	Authorization	Bearer {TOKEN_STRING}	Use <b>access_token</b> string from the authentication response. This header is required for every upload request.
	content-type	application/json	
<b>Post Body : application/json - Sample</b>			

```
{  
  "TransactionId": "80a4889c-6f1b-453d-a306-eccfb9382355"  
  "ServiceIdentifier": "{{UnitBasedServiceIdentifier}}",  
  "ServiceDate": "09-12-2020",  
  "ClientLtssId": "A1234567890",  
  "ProviderMa": "A12345678",  
  "Units": 2,  
  "NumberOfPeopleAuthorized": 8  
}
```

### Sample Billing Entry Request

#### Linux (With Unit-Based Service Identifier)

```
curl -X POST \  
https://provtstltss.health.maryland.gov/ltssv2/Ltss.SelfServeApi.Web/api/providerupload/uploadbillin  
gentry \  
-H 'Authorization: 'Bearer {{ api_token }}' \  
-H 'Content-Type: 'application/json' \  
-H 'cache-control: 'no-cache' \  
-data-raw '{ "TransactionId": "80a4889c-6f1b-453d-a306-eccfb9382355", "ServiceIdentifier":  
"{{UnitBasedServiceIdentifier}}", "ServiceDate": "09-12-2020", "ClientLtssId": "A1234567890",  
"ProviderMa": "A12345678", "Units": 2, "NumberOfPeopleAuthorized": 8 }'
```



### Windows (With Cost-Based Service Identifier)

```

$headers = @{
    "Authorization" = "Bearer {{access_token}}"
    "Cache-Control" = "no-cache"
}
$params = @{
    "TransactionId" = "80a4889c-6f1b-453d-a306-eccfb9382355"
    "ServiceIdentifier" = "{{CostBasedServiceIdentifier}}"
    "ServiceDate" = "09-12-2020"
    "ClientLtssId" = "A1234567890"
    "ProviderMa" = "A12345678"
    "Cost" = 5.4
}
Invoke-WebRequest -Uri
https://provtstltss.health.maryland.gov/ltssv2/Ltss.SelfServeApi.Web/api/providerupload/uploadbilli
ngentry -Headers $headers -Method POST -ContentType "application/json" -Body
($params|ConvertTo-Json)
    
```

*Note: The above sample request is for illustration purpose only. Every programming language will need to build the request in its native implementation.*

### Upload Billing Entry Response

Success			
	HTTP Response Code	200	Success
Response Header	TID	{TransactionId}	The same transaction Id from Request.
Fail			
Response Header	TID	{TransactionId}	The same transaction Id from Request.
	HTTP Response Code	401	Unauthorized This can be due to invalid or expired JWT.
		400	Bad request Data validation failed. 400 response will have ErrorCode and

			descriptive message for the validation that failed.
<b>Response Body for HTTP Response code 400 : application/json – Sample</b>			
<pre>{   "Message": "1000"   "ErrorCode": "Invalid SSO Username" }</pre>			

**Error Codes**

**HTTP Status 400**

The execution sequence of validations are in the order of Error codes. Provider upload api however will short circuit the validations and returns HTTP 400 error upon first failed validation.

Api at any point would always return a single error code and message.

<b>Explanation of all Error codes and messages for HTTP response code 400</b>		
<b>Error Code</b>	<b>Error Message</b>	<b>Validation Rule</b>
1000	Invalid SSO Username	No staff profile is associated with SSO username in LTSS
1100	TransactionId not present	Transaction Id is missing in request
1200	Invalid Date of Service	Date is not present in request. Date entered is invalid and cannot be parsed to a standard date. I.E. date is not in correct format. Date is less than dda-go-live (configured) date Difference between Service date and current date is more than 365 days.
1300	Invalid Unit or Cost	Any of the following:  Cost and Units are both absent in request. Or Both Cost and Units are specified and greater than 0 in request. Or Cost is defined and non-zero for a unit-based service Or Units is defined for a cost-based service  Unit should be a whole number Cost supports only up to 2 decimal point
1400	Duplicate TransactionId	Do not reuse TransactionId. An existing billing entry with the same transactionId will return error.

1500	Invalid Ltss Client Id	ClientId is blank in request ClientId does not exist in LTSS
1600	Invalid Provider MA#	Provider MA# is blank in request Provider MA# is not associated with any location
1700	Invalid Service Identifier	Service Identifier is blank in request Service Identifier is invalid
1800	Exceeds allowed cap limits	Cost specified is greater than max rate defined for the service group.
1900	Exceeds allowed cap limits	Units specified are greater than max units defined for the service group.
2000	Invalid Number of People Authorized	Any of the following:  NumberOfPeopleAuthorized was not provided for a ServiceIdentifier that requires a NumberOfPeopleAuthorized value (see <b>Upload Billing Entry Request Data Format</b> for relevant ServiceIdentifiers).  NumberOfPeopleAuthorized is not null and has a value that is either: less than or equal to 0, or greater than 8.

## HTTP Status 500

In the case of an HTTP 500 error, ensure that the header has a Content-Type set to 'application/x-www-form-urlencoded' for SSO calls and application/json for billing calls.

## Appendix – DDA Provider Upload API Requirements – Service Identifier Mapping

Service Name	Service Identifier for API Billing	# of People Authorized Required?	Max Units In a Day	Max Units In a Week	Max Units in a Month	Category of Service Code <i>Required to be assigned to the provider number to bill</i>
<b>Assistive Technology and Services</b>	ATS	N/A				2N
<b>BSS - Behavioral Assessment</b>	BSSBA	N/A	1			2G
<b>BSS - Behavioral Consultation</b>	BSSBC	N/A	32			2G
<b>BSS - Behavioral Plan</b>	BSSBP	N/A	1			2G
<b>BSS - Brief Support Implementation</b>	BSSBSI	N/A	32			2G
<b>Camp - Non-Respite (State Only Funded)</b>	SFCAMP	N/A				2N
<b>Career Exploration Services - Facility Based</b>	CEXFB	N/A	32	160		2E
<b>Career Exploration Services - Large Group</b>	CEXLG	N/A	32	160		Community Pathways/State Funded: 2E Community Supports: 2F
<b>Career Exploration Services - Small Group</b>	CEXSG	N/A	32	160		Community Pathways/State Funded: 2E Community Supports: 2F
<b>Community Development Services 1:1 Staffing Ratio</b>	CDS11	N/A	32	160		2H
<b>Community Development Services 2:1 Staffing Ratio</b>	CDS21	N/A	32	160		2H
<b>Community Development Services Group (1-4)</b>	CDSG	N/A	32	160		2H
<b>Community Living - Enhanced Supports</b>	CLES	Yes	1			2A
<b>Community Living - Enhanced Supports Retainer Fee</b>	CLESR	Yes	1			2A
<b>Community Living - Enhanced Supports Trial Experience</b>	CLEST	Yes	1			2A
<b>Community Living - Group Home</b>	CLGH	Yes	1			2B
<b>Community Living - Group Home Retainer Fee</b>	CLGHR	Yes	1			2B
<b>Community Living - Group Home Trial Experience</b>	CLGHT	Yes	1			2B
<b>Day Habilitation 1:1 Staffing Ratio</b>	DH11	N/A	32	160		Community Pathways/State Funded: 2C Community Supports: 2D
<b>Day Habilitation 2:1 Staffing Ratio</b>	DH21	N/A	32	160		Community Pathways/State Funded: 2C Community Supports: 2D

<b>Service Name</b>	<b>Service Identifier for API Billing</b>	<b># of People Authorized Required?</b>	<b>Max Units In a Day</b>	<b>Max Units In a Week</b>	<b>Max Units in a Month</b>	<b>Category of Service Code</b> <i>Required to be assigned to the provider number to bill</i>
<b>Day Habilitation Large Group (6-10)</b>	DHLG	N/A	32	160		Community Pathways/State Funded: 2C Community Supports: 2D
<b>Day Habilitation Small Group (2-5)</b>	DHSG	N/A	32	160		Community Pathways/State Funded: 2C Community Supports: 2D
<b>Dedicated Hours Community Living - Enhanced Supports (1:1)</b>	CLES11	N/A	96			2A
<b>Dedicated Hours Community Living - Enhanced Supports (2:1)</b>	CLES21	N/A	96			2A
<b>Dedicated Hours for Community Living - Group Home (1:1)</b>	CLGH11	N/A	96			2B
<b>Dedicated Hours for Community Living - Group Home (2:1)</b>	CLGH21	N/A	96			2B
<b>Dedicated Hours for Supported Living (1:1)</b>	SL11	N/A	96			2R
<b>Dedicated Hours for Supported Living (2:1)</b>	SL21	N/A	96			2R
<b>Employment Services - Co-worker Employment Supports</b>	ESCOW	N/A				2I
<b>Employment Services - Customized Self-Employment</b>	ESCSE	N/A	1			2I
<b>Employment Services - Discovery Milestone 1</b>	ESDM1	N/A	1			2I
<b>Employment Services - Discovery Milestone 2</b>	ESDM2	N/A	1			2I
<b>Employment Services - Discovery Milestone 3</b>	ESDM3	N/A	1			2I
<b>Employment Services - Follow Along Supports</b>	ESFAS	N/A			1	2I
<b>Employment Services - Job Development</b>	ESJD	N/A	32	160		2I
<b>Employment Services - On-going Job Supports</b>	ESOGJS	N/A	64 (40 pre 7/1/23)			2I
<b>Environmental Assessment</b>	EA	N/A	1			2N
<b>Environmental Modification</b>	EM	N/A				2N

<b>Service Name</b>	<b>Service Identifier for API Billing</b>	<b># of People Authorized Required?</b>	<b>Max Units In a Day</b>	<b>Max Units In a Week</b>	<b>Max Units in a Month</b>	<b>Category of Service Code</b> <i>Required to be assigned to the provider number to bill</i>
<b>Family and Peer Mentoring Supports</b>	FPMS	N/A		32		2J
<b>Family Caregiver Training and Empowerment Services</b>	FCTE	N/A				2J
<b>Housing Support Services</b>	HSS	N/A		32		2L
<b>Live-In Caregiver Supports</b>	LICS	N/A				2N
<b>Nursing - Skilled Nursing Services (State Only Funded)</b>	NSK	N/A		96		2M
<b>Nursing Support Services</b>	NSS	N/A		96		2M
<b>Other (State Only Funded)</b>	OTH	N/A				2N
<b>Participant Education, Training and Advocacy Supports</b>	PEDA	N/A				2J
<b>Remote Support Services</b>	RSS	N/A				2S
<b>Rent - Individual Support (State Only Funded)</b>	RIS	N/A				2N
<b>Respite Care Services - 15 Minute (Pre October 1, 2023)</b>	RESPH	N/A		96		2P
<b>Respite Care Services - Camp</b>	RESPC	N/A				2P
						Community Pathways/State Funded: 2B
						Family Supports/Community Supports:
<b>Respite Care Services - Day</b>	RESPD	N/A		1		2U
<b>Shared Living - Level 1</b>	SHARLIV1	N/A			1	2Q
<b>Shared Living - Level 2</b>	SHARLIV2	N/A			1	2Q
<b>Shared Living - Level 3</b>	SHARLIV3	N/A			1	2Q
<b>Supported Living</b>	SUPLIV	Yes		1		2R
<b>Transition Services</b>	TRANS	N/A				2N
<b>Transportation</b>	TRNSPT	N/A				2N
<b>Vehicle Modifications</b>	VM	N/A				2N