


Bearer Token Authentication

The Bearer Token Authentication

Apstrata provides the users and devices a bearer token authentication method that enables them to access protected resources without sending their credentials or tokens as parameters. Instead, users and devices can set in the **Authorization** header of any Apstrata HTTP request a bearer token that will be computed based on the application authentication key, their unique identifier and their Apstrata token.

 Once the bearer token is set in the Authorization header, no signature, token, application key or identifier should be passed to the request.

The bearer token is computed by encoding the following string **authkey:Identifier:token** with the Base64 encoding scheme and then be set in the Authorization header of the request as follows:

```
Authorization: Bearer <Base64_encode(authKey:Identifier:token)>
```

The Bearer Token Authentication Rules

The Bearer Token Authentication on Apstrata follows the rules below:

- User or Device sends the request header Authorization with the bearer token to `https://<serviceName>/rest/<action>`.
- If the signature parameter, the token parameter or a cookie containing a valid Apstrata token is sent while using the bearer token authentication method then an `INVALID_REQUEST` exception will be returned.
- The bearer token authentication method can be used by anonymous requests simply by Base64 encoding the application key (**Base64_encode(authKey)**).
- The bearer token authentication follows the same rules as the [token-based authentication](#), therefore it can only be used with the same list of Apstrata APIs.
- When using the bearer token authentication with a custom API, the application key should be removed from the service URL as follows: `https://<serviceName>/rest/r/<scriptname>`.
- If the Authorization request header is sent using http, and it contains a valid token, the token will be invalidated.

The example below describes how to send a request to Apstrata using the bearer token authentication method.

Device `R2D2` belongs to the application `X735F0C3PO`, it needs to execute the Apstrata script `deliverMessage` using the bearer token authentication method. It will request a token by calling the [GenerateToken](#) API and signing with its credentials.

It will then use the returned token (`1FFB2081F4E4A0680D72E469AEDB79AC`) to compute its bearer token by Base64 encoding the following string:

```
X735F0C3PO:R2D2:1FFB2081F4E4A0680D72E469AEDB79AC
```

Finally, device `R2D2` sends the `RunScript` request to the service URL and adds the bearer token (`WDczNUYwQzNQZpSMkQyOjFGRkIyMDgxRjRFNEEwNjgwRDcyRTQ2OUFFREI3OUFD`) to the Authorization header.

Service URL

```
https://<serviceName>/rest/RunScript?apsws.time=[timestamp]&apsdb.scriptName=deliverMessage
```

Request Headers

```
Authorization: Bearer WDczNUYwQzNQZpSMkQyOjFGRkIyMDgxRjRFNEEwNjgwRDcyRTQ2OUFFREI3OUFD
```