

# Atbash Octopus JWT Support

Rudy De Busscher

Version 0.5.0, 01/05/2018

# Table of Contents

Release Notes .....	1
0.5.0 .....	1
Introduction.....	1
Standard usage .....	1

User manual for Atbash configuration.

## Release Notes

### 0.5.0

1. First separate release from Octopus repository.

## Introduction

A first attempt for converting Java beans to and from a JWT. The code will be improved when Atbash Octopus has full support for MP JWT Auth spec and OAuth2/OpenId Connect.

In a second version, a better support for Cryptographic keys is planned. There is no support for JWE yet.

## Standard usage

Convert the Object *data* to JSON and Base64 encoded format.

```
@Inject
private JWTEncoder jwtEncoder;

JWTParameters parameters = JWTParametersBuilder.newBuilderFor(JWTEncoding.NONE)
    .build();

String encoded = jwtEncoder.encode(data, parameters);
```

Use the Object *data* as JWT payload, signed with a HMAC (Warning HMAC support will be changed in next release)

```
@Inject
private JWTEncoder jwtEncoder;

JWTParameters parameters = JWTParametersBuilder.newBuilderFor(JWTEncoding.JWS)
    .withHeader("Octopus Offline", "v0.2")
    .withSecretKeyForSigning(new HMACSecret(localSecret,
LOCAL_SECRET_KEY_ID, true))
    .build();

String encoded = encoder.encode(data, parameters);
```

In the above example, the JWT has a custom header.

Instead of injecting the Encoder, it is also possible to instantiate the encoder directly.

```
JWTEncoder jwtEncoder = new JWTEncoder();
```

Converting the JWT (or Base64 encoded JSON) to an Object instance.

The following example converts a signed JWT.

```
@Inject
private JWTDecoder jwtDecoder;

@Inject
private JWKManagerKeySelector keySelector;

@Inject
private MPBearerTokenVerifier verifier;

JWTData<MPJWTToken> data = jwtDecoder.decode(token, MPJWTToken.class, keySelector,
verifier);
MPJWTToken token = data.getData();
```

JWKManagerKeySelector → Selector of the key based on the id found in the header. In a next release, selector will work on multiple sources (JWK, PEM files and Java Key Stores)

MPBearerTokenVerifier → Optional verifier for validating the JWT.

TODO : Describe the default verifications.