



A2019 JSON Web Token Package

Readme

Version 1.0.1

19/03/2021

Table of Contents

Readme	1
Table of Contents	2
1. Introduction	3
1.1 Overview	3
1.2 Use cases	3
2. Requirements & Prerequisites	4
2.1 System Requirements	4
2.2 Prerequisites	4
3. Getting Started	5
3.1 Quick Start	5
3.1.1 Setup	5
3.1.2 Configuration and Use	5
3.1.3 Key formats	7
3.2 Support	8
3.3 FAQs	8

1. Introduction

This document contains all essential information for the user to make full use of this **A2019 JSON Web Token Package**. It includes a description of the functions and capabilities and step-by-step procedures for setup & configuration of the **A2019 JSON Web Token Package**.

1.1 Overview

With the package actions you can build bots that generate a JSON Web Token for authentication and authorization when you call REST Services from DocuSign, Microsoft and many more

1.2 Use cases

The key use cases include:

- Generate a JSON Web Token for authentication
- Validate a received JSON Web Token

2. Requirements & Prerequisites

2.1 System Requirements

- Enterprise A2019.19+

2.2 Prerequisites

- Secrets or keys in PKCS#1 or PKCS#8 format to sign or verify JSON Web Tokens

3. Getting Started

3.1 Quick Start

3.1.1 Setup

Import the A2019 JSON Web Token Package in the Control Room under Bots->Packages

3.1.2 Configuration and Use

The package contains the following command actions



Get Signed Token



Parse JWT

Name	Description	Parameter
Get Signed Token	Generate and sign a Token	Private or Secret Key in PKCS#1, PKCS#8 or Secret String format
		String that contains the key for signing
		Issuer (optional)
		Standard claim "iss"
		Subject (optional)
		Standard claim "sub"
		Audience (optional)
		Standard claim "aud"
		Id (optional)
		Standard claim "jti"
		Issued At, if not set = now (optional)
		Standard claim "iat"
		Expires after (sec) (optional)
		Standard claim "exp"
		Not Before (sec) (optional)
		Standard claim "nbf"
		Private Claims (optional)
		Dictionary with custom claims

		Signature Algorithm	Signature algorithm to sign the token, must match with the key
		JSON Web Token	Generated JSON Web Token
Parse JWT	Parse and validate Token	JSON Web Token	Token to verify
		Public or Secret Key in PKCS#1, PKCS#8 or Secret String format	Key to verify the token like a public key or secret
		Signature Algorithm	Algorithm which was used to sign the token
		A dictionary with the JWT claims based on RFC 7519	If a valid token the dictionary contains the standard and custom claims

3.1.3 Key formats

Keys must be in a certain format if not a secret

- a line consisting of 5 hyphens, the word BEGIN, one or a few (space-separated) words defining the type of data, and 5 hyphens
- base64 of the data,
- a line consisting of 5 hyphens, the word END, one or a few (space-separated) words defining the type of data, and 5 hyphens

The actions do some reformatting to avoid formatting issues

PKCS#1 sample

```
-----BEGIN RSA PRIVATE KEY-----
MIICXAIBAAKBgQCjcGqTkOq0CR3rTx0ZSQSIdTrDrFAYI29611xN8aVgMQIWtDB/
ID0W5TpKPuU9iaiG/sSn/VYt6EzN7Sr332jj7cyl2WrrHI6ujRswNy4HojMuqtfa
b5FFDpRmCuvl35fge18OvoQTJELhhJ1EvJ5KUeZiuJ3u3YyMnxxXzLuKbQIDAQAB
AoGAPrNDz7TKtaLBvaluMaMXgBopHyQd3jFKbT/tg2Fu5kYm3PrnmCoQfZYXFKCo
ZUFIS/G1FBVWWGpD/MQ9tbYZkKpwuH+t2rGndMnLXiTC296/s9uix7gsjnT4Naci
5N6EN9pVUBWqMGrYUTHFc58ThtelSiPARX7LSU2ibtJSv8ECQQDWBRRrAYmbCUN7
ra0DFT6SppaDtvvuKtb+mUeKbg0B8U4y4wCIK5GH8EyQSwUWcXnNBO05rlUPbifs
DLv/u82IAkEaw39sTJ0KmJJyaChqvqAJ8guulKlgucQJ0Et9ppZyet9iVwNKX/aW
9UlwGBMQdafQ36nd1QMEA8AbAw4D+hw/KQJBANJbHDUGQtK2hrSmZNoV5HXB9Uiq
7v4N71k5ER8XwgM5yVGs2tX8dMM3RhnBEtQXXs9LW1uJZSOQcv7JGXNnhN0CQBZe
nZrJAWxh3XtznHtBfsHWelyCYRIAj4rpCHCmaGUM6ljCVKFUawOYKp5mmAyObkUZ
f8ue87emJLEdynC1CLKCQHduNjP1hemAGWrd6v8BHhE3kKtcK6KHsPvJR5dOfzbd
HAqVePERHISfN6cwZt5p8B3/JUwSR8el66DF7Jm57BM=
-----END RSA PRIVATE KEY-----
```

PKCS#8 sample

```
-----BEGIN PRIVATE KEY-----
MIICdgIBADANBgkqhkiG9w0BAQEFAASCAmAwggJcAgEAAoGBAKNwapOQ6rQJHetP
HRIJBIh1OsOsUBiXb3rXXE3xpWAXAha0MH+UPRbIOko+5T2JqIb+xKf9Vi3oTM3t
KvffaOPtXKZauscj6NGzA3LgeiMy6q19pvkUUOlGYK6+Xfl+B7Xw6+hBMkQuGE
nUS8nkpR5mK4ne7djlyHffMu4ptAgMBAAECgYA+sOPPtMq1osG9oi4xoxeAGikf
JB3eMUptP+2DYW7mRibc+ueYKhB9IhcUoKhIQUhL8bUUFVZYakP8xD21thmQqnC4
f63asad0ycteJMLb3r+z26LHuCyOdPg1pyLk3oQ32IVQHBCYathRMcVznxOG16VK
I8BFfstJTaJu0IK/wQJBANYFGusBiZsJQ3utrQMVPPkmlO2++4q1v6ZR4puDQHx
TjLjAlgrkYfwtJBLBRZxec0E7TmuVQ9uJ+wMu/+7zaUCQQDDf2xMnQqYknJoKGq+
oAnyC66UqWC5xAnQS32mInJ632JXA0pf9pb1SXAYExB1p9Dfqd3VAwQDwBsDDgP6
HD8pAkEA0lscNQZC2TaGtKZk2hXkdcH1SKru/g3vWtKRHxfCAznJUaza1fx0wzdG
GcES1Bdez0tbW4lII5By/skZ2ceE3QJAFI6fOskBbGHde3Oce0F+wdZ6XIJhEgCP
iuklCkZoZQzoiMJUoVRrA5gqnmaYDI5uRRl/y57zt6YksR3KcLUlUQJAd242M/WF
6YAZat3q/wEeTeQq1wroew+8IHI05/Nt0cCpV48RGEHJ83pzBm3mnwHf8ITBJH
x6XromXsmbnsEw==
-----END PRIVATE KEY-----
```

3.2 Support

Free bots are not officially supported. You can get access to Community Support through the following channels:

- You can get access to Community Support, connecting with other Automation Anywhere customers and developers on [APeople](#) – the [Bot Building Forum](#), the [Bot Store Support Forum](#), or the [Developers Everywhere Group](#).
- Automation Anywhere also provides a [Product Documentation portal](#) which can be accessed for more information about our products and guidance on [Enterprise A2019](#).

3.3 FAQs

You find more information about JSON Web Tokens here

<https://tools.ietf.org/html/rfc7519>

<https://jwt.io/> (Includes a JWT Debugger)

For questions relating to Enterprise A2019: See the [Enterprise A2019 FAQs](#).

Appendix A: Record of Changes

Instructions: Provide information on the version number, the date of the version, the author/owner of the version, and a brief description of the reason for creating the revised version.

No.	Version Number	Date of Change	Author	Notes
1	1.0.1	19/03/2021	Stefan Karsten	Initial version

Appendix B: References

No.	Topic	Reference Link
1	Overview of Enterprise A2019	Click here
2	Guidance: Building basic A2019 bots	Click here
3	Guidance: Building A2019 action packages	Click here
4	APeople Community Forum	Click here
5	Automation Anywhere University	Click here