PSBluesky Help[1]

https://jdhitsolutions.github.io

v1.2.0

[1]https://github.com/jdhitsolutions/PSBluesky

Contents

PSBluesky

This module is a set of PowerShell functions designed to let you interact with Bluesky API from PowerShell. Technically, the module commands are wrappers around the atproto protocols. The module is written for PowerShell 7, although it might work as written in Windows PowerShell with minimal changes. Commands *have not* been thoroughly tested for cross-platform compatibility.

Installation

You can install the module from the PowerShell Gallery.

```
Install-Module -Name PSBluesky
```

Or using `Install-PSResource` from the Microsoft.PowerShell.PSResourceGet module.

```
Install-PSResource -Name PSBluesky -Repository PSGallery -TrustRepository
```

You might want to also install the following modules or related modules to securely store your Bluesky credentials:

- Microsoft.PowerShell.SecretManagement
- Microsoft.PowerShell.SecretStore

After installing this module, you should end up with these PSBluesky commands:

- Add-BskyImage
- Get-BskyAccessToken

- Get-BskyFeed
- Get-BskyFollowers
- Get-BskyFollowing
- Get-BskyProfile
- Get-BskyNotification
- Get-BskySession
- Get-BskyTimeline
- New-BskyPost
- Open-BskyHelp
- Update-BskySession

Authentication

Tokens

In order to send data, you must authenticate. The `Get-BskyAccessToken` function will retrieve an access token. You shouldn't need to call this command directly. Other module commands will call it and pass the authentication token as needed. The access token has a limited lifetime unless it is refreshed. Beginning with version 1.2.0, the module will refresh the token every 15 minutes through a background runspace using a synchronized hashtable. If you remove the module, the runspace will be removed as well.

Run `Get-BskySession` to see your current session information.

```
PS C:\> Get-BskySession

   User: jdhitsolutions.com


Active AccessToken              RefreshToken              Age
------ -----------              ------------              ---
True   eyJ0eXAiOiJhdCtqd3Qi ... eyJ0eXAiOiJyZWZyZXNo ... 00:05:41.5115796
```

If you want to manually test the Bluesky API in your own code, you can use the access and refresh token properties from this object.

```
PS C:\> $bskySession = Get-BskySession
PS C:\> $access = $bskySession.AccessToken
PS C:\> $refresh = $bskySession.RefreshToken
```

However, it is still possible you will encounter an expired token error message.. If you do, you can run `Update-BskySession` to refresh the token.

```
Get-BSkySession | Update-BskySession
```

If this also fails, remove the module, re-import and start a new Bluesky session.

Credentials

You will need to create a PSCredential object with your Bluesky username and password. **The username is case-sensitive**. For automation purposes, you can use the Secrets management module to store your credential. Write your own code to retrieve the credential and pass it to the module commands.

You might want to use `PSDefaultParameterValues` to set the credential for all commands.

```
$PSDefaultParameterValues['*-Bsky*:Credential'] = $BlueskyCredential
```

Rate Limits

The commands in this module use the public Bluesky API which means there are rate limits. If you exceed the rate limit, you will get an error message. You will need to wait until the rate limit resets.

⚠️ There is a rate limit of 300 new sessions per day. This shouldn't be an issue for most people unless you are testing code or running some sort of high-volume automation.

Posting

Use `New-BskyPost`, or its alias `skeet`, to post a message to Bluesky. There are parameters to include an image. If you include an image, the `New-BskyPost` command will call `Add-BskyImage` to upload the image. It is strongly recommended that you included ALT text for the image.

```
$param = @{
    Message   = "Getting close to sharing my #PowerShell Bluesky code."
    ImagePath = "C:\work\MsPowerShell.jpg"
    ImageAlt  = "Ms. PowerShell"
    Verbose   = $true
}
New-BskyPost @param
```

The output is a URL to the post.

If your message contains a URL, it will be converted to a clickable link. Make sure your link is surrounded by white space. Beginning with v1.0.0, you can post Markdown style links.

```
PS C:\> $m = "Testing multiple Markdown style links from my [#PowerShell PSBluesky
module](https://github.com/jdhitsolutions/PSBluesky) which you can find on the
[PowerShell Gallery](https://www.powershellgallery.com/packages/PSBlueSky/0.6.0)"
PS C:\> skeet $m
```


Figure 1: Markdown links in a Bluesky post

This example is using the `skeet` alias for `New-BskyPost`.

Profiles

The module has a command to retrieve a Bluesky profile.

```
Get-BskyProfile jdhitsolutions.com
```

The module uses a custom format file.

The user's profile name should be a clickable link.

👫 Followers

You can retrieve a list of your followers. You can specify a number of followers between 1 and 100. The default is 50.

```
Get-BskyFollowers -Limit 2
```

The custom formatting includes a clickable link to the follower's profile if running in Windows Terminal or a console that supports hyperlinks.

You can pipe the follower object to `Get-BskyProfile` to retrieve more information.

```
PS C:\>$f= Get-BskyFollowers
PS C:\> $f[12] | Get-BskyProfile
```

```
PS C:\> Get-PSBlueskyProfile jdhitsolutions.com

Jeff Hicks [jdhitsolutions.com]

PowerShell Author/Teacher/MVP/Guide ✍️
Prof. PowerShell Emeritus 🎓
Grizzled and grumpy IT Pro – https://jdhitsolutions.github.io/
🎼Amateur composer – https://musescore.com/user/26698536
Wine drinker 🍷🐶 and dog lover


Created                 Posts Followers Following Lists
───────                 ───── ───────── ───────── ─────
5/21/2023 2:44:48 PM     533       317       172     0
```

Figure 2: A Bluesky user profile

```
PS C:\> Get-PSBlueskyFollowers –Limit 2

Display       Username           Description
───────       ────────           ───────────
Jonathan Lank jpluk.bsky.social
Libby Brown   trubludevil.social Product manager @ SpruceID. Mom. Digital Wallet and Passk
```

Figure 3: Bluesky followers

```
Jess Pomfret [jpomfret.bsky.social]

Database Engineer with a passion for automation, proper football and fitness.
She/Her.


Created              Posts Followers Following Lists
-------              ----- --------- --------- -----
8/14/2023 3:58:44 PM   125       236       157     1
```

The default behavior is to retrieve between 1 and 100 followers. Or you can use the –All parameter to retrieve all followers.

Following

Likewise, you can get a list of all the accounts you are following.

```
Get-BskyFollowing -limit 3
```

Figure 4: Bluesky following

As with followers, the default behavior is to retrieve between 1 and 100 accounts you are following. Or you can use the –All parameter to retrieve all accounts you are following.

Feed 📰

Use Get-BskyFeed to retrieve the latest posts from *your* feed. You can query for 1 to 100.

```
Get-BskyFeed -Limit 3
```

The default output uses a custom format file. The current behavior is to get posts and replies.

Figure 5: Getting your Bluesky feed

The formatted output includes clickable links to the the author, which might be different than you if reposting, and the post. For best results, run PowerShell 7 in Windows Terminal.

Timeline 📅

If you want to view items from your timeline, use the `Get-BskyTimeline` command. You can specify a limit of 1 to 100.The default is 50.

```
Get-BskyTimeline -Limit 25
```

The command uses a custom format file.

The default formatted output includes clickable links to the author and the post.
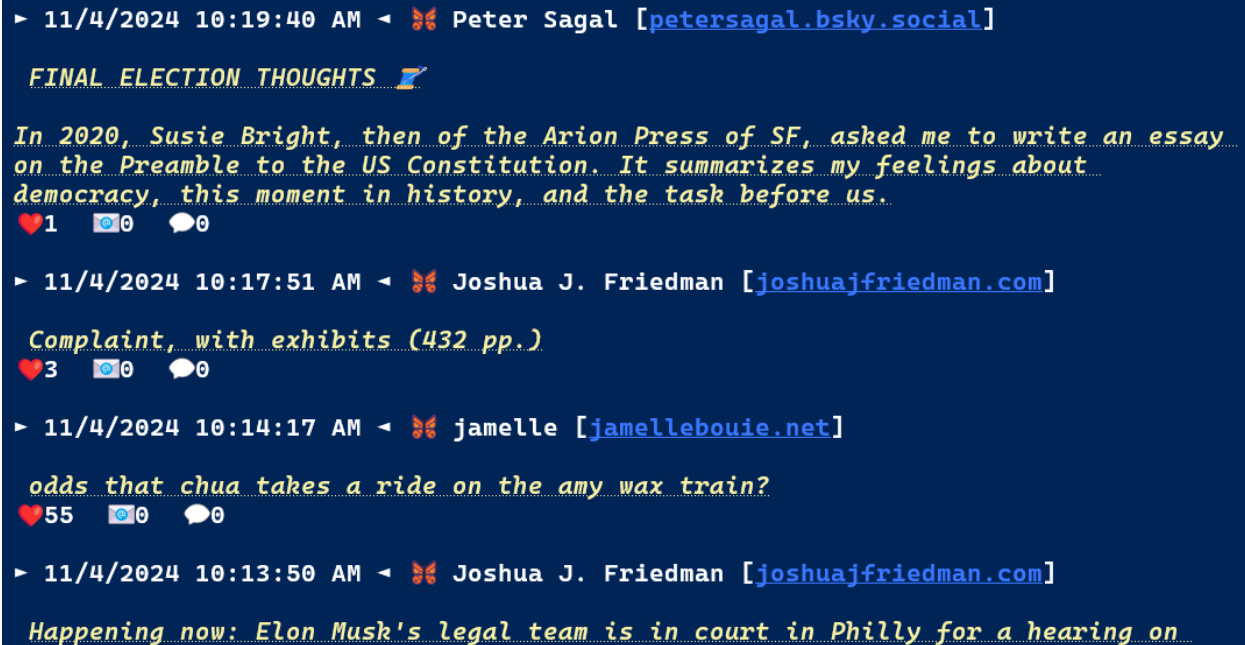
8

Figure 6: Getting your Bluesky timeline

Notifications 🔔

You can retrieve your notifications with `Get-BskyNotification`. You can specify a limit of 1 to 100. The default is 50.

```
Get-BskyNotification -limit 10
```

The default formatted output includes clickable links to the author and the liked post.

ℹ️ Information and Troubleshooting

The commands in this module should write the raw response from the API request to the Information stream. Some commands might include additional information.

```
![Information stream](images/bsky-information.png)
```

The output will be an object.

```powershell
PS C:\> $v.MessageData | Select-Object did,handle,*count
```

Figure 7: Getting your Bluesky notifications

```
did              : did:plc:ohgsqpfsbocaaxusxqlgfvd7
handle           : jdhitsolutions.com
followersCount   : 322
followsCount     : 177
postsCount       : 543
```

You might want to set a default parameter value.

```
$PSDefaultParameterValues['*-*Sky*:InformationVariable'] = "iv"
```

Roadmap 🗺️

I have a short list of items on my wish list:

- support posting multiple images
- localized verbose and other messaging
- commands to work with direct messages, aka chat
- maybe create a TUI-base reader for your timeline

If you are testing the module and think you've found a bug, please post an **Issue**. For all other topics and questions, including feature requests, please use the repository's **Discussions** section.