

ONLINE MARKETING WITH SOCNET SDK (PRACTICAL)

Gumuruh S.
IT Trainer & Practicioner

- 2 Type of SocNet SDK User
- How does SocNet SDK work?
- 3 Main Files
- Software Dev. Requirements
- Basic Simulation
- Challenging Simulation

2 TYPES OF SOCNET SDK USERS



- **SocNet SDK** is used by 2 users:

- **Non-Developers**

1. Called by Parameter Call within Command Prompt
(these slides)

- **Developers**

1. Called within Java Code as Library
(different slides)

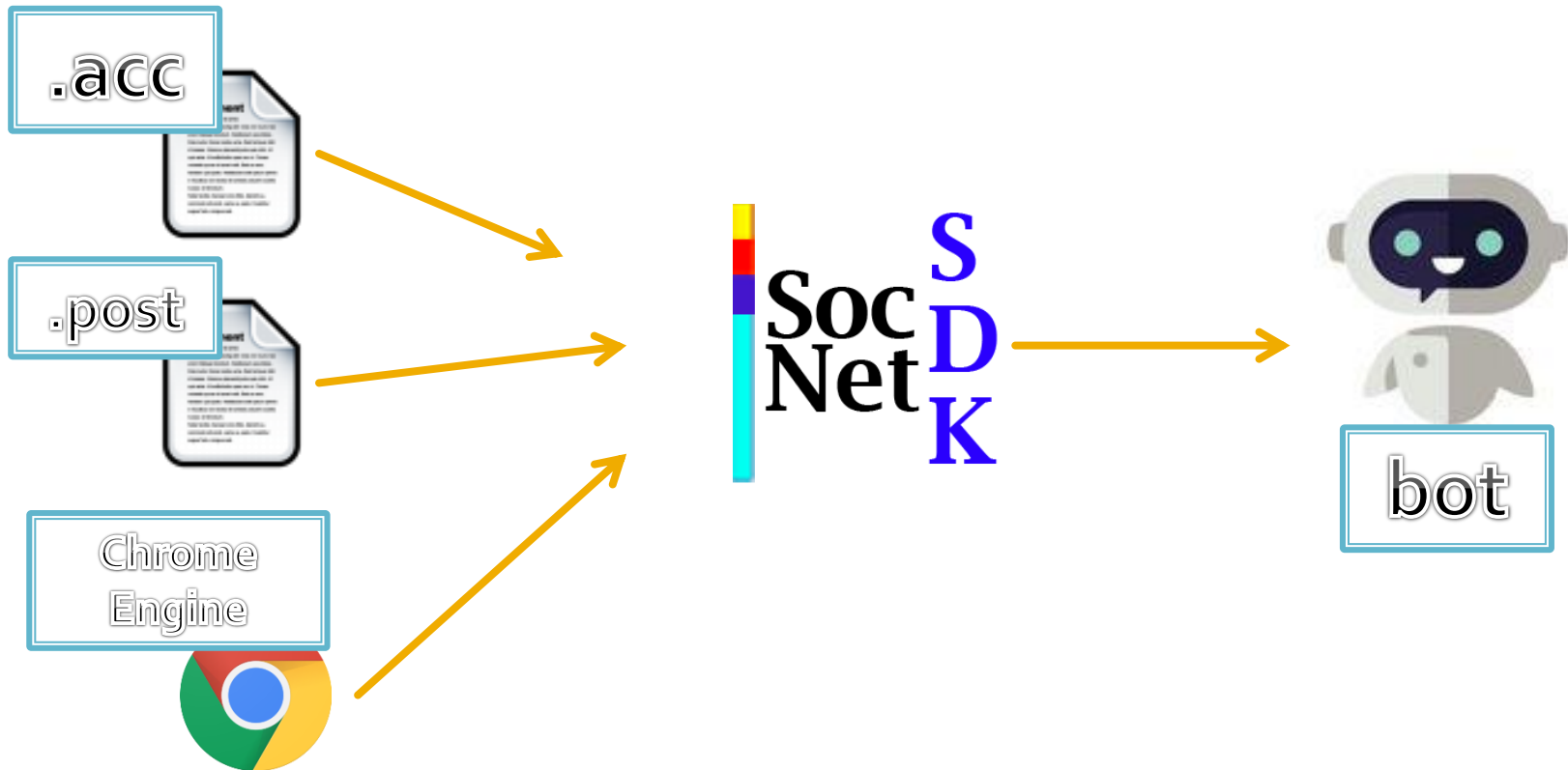
HOW DOES SOCNET SDK WORKS?



- **SocNet SDK** will start by executing several main files such as :
 - **Chromedriver** : main engine.
 - ***.acc** : user login details.
 - ***.post** : status to be published.



How does SocNet SDK work?



- **'bot'** means any program activities (executed by itself) without any user interaction (clicking / entering something).
- **SocNet SDK** implements **jxbrowser**, **Selenium**, and also **Chrome Engine** that will be executed as **bot**.

- Several common activities by **bot** such as:
 - clicking button,
 - Typing text,
 - Pressing enter,
 - Move page to another, dsb....

3 MAIN FILES



■ *.acc File:

- ✓ Consists of user login.
- ✓ The content format is fixed.
- ✓ This file will be used by **SocNet SDK** and encrypted *.acc.

■ *.acc Format:

✓ [username;pass;sosmed]

✓ Example:

✓ [brian28@yahoo.com;bukalangsung28;facebook](#)

✓ [ninag99@hotmail.com;abcd;twitter](#)

- ***.post File:**

- ✓ A file containing text to be published. Each post are separated by new line.

- ***.post Format:**

- ✓ [text-based]

- ✓ Example:

- ✓ Cool man!

- ✓ Now we're joining © YourBrand.com since 2013!

■ Chrome Engine

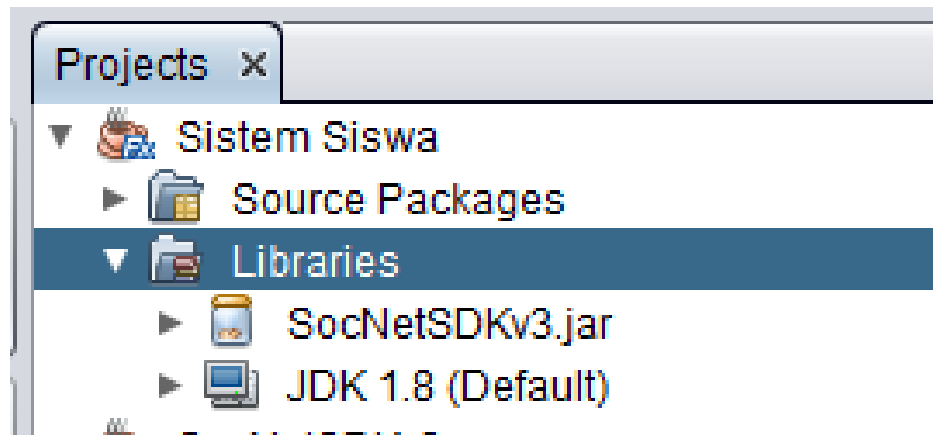
- ✓ *.exe File used in Windows OS, and *.sh file used in Linux, etc.
- ✓ Named as **chromedriver.exe** version of 2.36.
- ✓ Combined by **jxbrowser (java library)** version of 6.22.

SOFTWARE DEV. REQUIREMENTS

To use **SocNet SDK** as **Java Developers** you have to understand about :

- OOP Concept
- Method
- Thread

And also add the **SocNet SDK *.jar file** within **Java Project**.



BASIC SIMULATION



Basic Simulation

Let's try to do:

- 1) Logging into Facebook,
- 2) Post a status 1x.
- 3) Close

only **5 lines of Code!**



```
Engine myBot = new Engine();  
Account myUser = new Account("satu@gmail.com", "konci123", SocialMedia.FACEBOOK);  
myUser.addPost("try SocNetSDK now!");  
myBot.addUser(myUser);  
myBot.start(true);
```

Only **5 lines of Code!**

```
Engine myBot = new Engine();  
Account myUser = new Account("satu@gmail.com", "konci123", SocialMedia.FACEBOOK);  
myUser.addPost("try SocNetSDK.com!");  
myBot.addUser(myUser);  
myBot.start(true);
```

Give **true** value to make **Auto-Closing** for an active **Frame** after all works done!

Or leave it **empty**, so **Frame** will not be Closed.



CHALLENGING SIMULATION



Challenging Simulation

What if the Username & Password used by **2 different of Social Medias?**

Let's give a time interval for avoiding **security blocking**.

Let's see the differences...

Source Code is still **5 lines!**



```
Engine myBot = new Engine();  
Account myUsers []= AccountBuilder.create("one@gmail.com", "key", "horray socnetSDK!",  
SocialMedia.FACEBOOK, SocialMedia.TWITTER);  
myBot.addUsers(myUsers);  
myBot.start(true, 3, 7);
```

Source Code is still **5 lines!**

```
Engine myBot = new Engine();  
Account myUsers []= AccountBuilder.create("one@gmail.com", "key", "horray socnetSDK!",  
SocialMedia.FACEBOOK, SocialMedia.TWITTER);  
myBot.addUsers(myUsers);  
myBot.start(true, 3, 7);
```

Give an **interval number**, for posting with a minute gap (defined as random value).
It means, start Engine with a gap of 3-7 minutes.



Twitter : @fggroupindonesia

FB : <http://fbl.me/fggroupindonesia>

Tel / SMS / WhatsApp :

+62-857-9556-9337

+62-857-2126-1437