**[Introduce myself]**

Thanks, pay yan. Let me share my screen.

Share screen.

Ok, so, Hello everyone. I’m Isaac Koh, and today we will be going through the document creator and how to set up the configuration files to suit your business needs.

Click slide.

**[CONTENT]**

So, let’s go through the content for today.

Ok, the first is the introduction to the document creator.

Next is the introduction to the configuration file.

Then, setting up of the config file for the transferable document.

Than after that we will set up, in the same config file, for the verifiable document.

After all is done, we will issue a document using the config file that was just created.

Then we will verify the issued document.

and lastly, we will have our Q & A.

Click slide.

So, before we dive in, for those of you who learn or understands better by reading, you can view the documentation online at docs.tradetrust.io/document-creator.

So, the links will be provided at the end of the webinar, ok?

So, let’s take a look.

Go to website to show them.

Ok, yup, this is the documentation.

So, you all can refer to down here if you want to, OK?

So, Ok, before we start, let me show you guys a quick demo of this document creator.

Go to demo tab.

**[DEMO of document creator]**

So, if we go to Creator.tradetrust.io.

You will be able to see this page.

So, this is the document creator, and here is where we drop our configuration file.

Select the config file.

So, once it is in, and everything is good, you will be directed to this log in page.

Over here, we will need to enter the wallet password.

So, just let me input my wallet password.

Enter password, and login.

Ok, next you will have to select the document that you want to create.

Then for this demo we will be using the cover letter.

Click the cover letter.

Then here is where you will have to fill up the form.

And attach any attachment you want.

Once you are done, you can click this button to issue the document.

Click on issue document button.

This process may take a while, so once it is done, you should be able to see this page.

You can download the issued document by just clicking this download button.

So, yes, this is document creator in a nutshell.

So, let’s go back to the slides.

Go back to the slides & Click slide.

**[Brief overview of Document Creator]**

So, a brief overview of the document creator.

What is document creator about?

It’s to provide a UI to the staffs that are involved with the daily operations who do not need to know anything about blockchain.

Why do we do this?

Mainly because we want to cater to small firms or companies that doesn't really have capabilities to build deep integration with their existing softwares or workflows.

So, for example, the tech department or someone tech savvy enough can set up the configuration file and clerks or other staffs can use this document creator to create the documents necessary for their business needs.

Click slide.

**[OUTCOME]**

Ok, outcome.

So, at the end of this session, you will know how to set up a configuration file for your non-tech staffs or businesses.

Click slide.

**[Prerequisite]**

Ok, so before we start, there are some prerequisite that we need.

Firstly, is the Ethereum wallet.

Secondly, we will need the token registry for the transferable record and the document store for the verifiable document.

Next is, we will need a configured DNS.

Lastly, we will need a custom renderer.

So, a little re-cap. In the previous webinar, webinar 2, we did the creation of the Ethereum wallet, document store, configured the DNS and touched on the custom renderer.

Then, in webinar 3, we did the token registry.

So, if any of you need any refresher, you can go on to re-visit those webinars at our website.

Links will be provided at the end of the presentation.

Click slide.

**[config file]**

Ok next, the configuration file.

Click slide.

Firstly, what is the configuration file?

It is a JSON file that contains information to configure the application to suit your business needs.

Then, what information does it contain?

As of now, it contains 3 important keys.

First is the network.

It can be the mainnet, or ropsten testnet, or rinkeby testnet.

Second is the Ethereum wallet.

Oh ya, one thing to note is that, this wallet has to be in a string.

Move to VSCODE to show wallet.

So, if you have a wallet.json, it will look something like this.

Then we can use this online converter to help get the wallet string.

So, all you need to do, is to copy the wallet and paste here.

Then convert it.

Then this is the wallet string that we want.

Go to slide with wallet string.

Then alternatively, if you have a private key, you also can convert it to the wallet string.

So, let me demonstrate it.

So, if you are using metamask, it is very easy to get your private key.

So, here in account details, export private key, then key in your password.

Then here you will get your private key.

OK, so let’s copy this and paste it somewhere.

Ok, next, we need to use the terminal.

Go to terminal.

Ok, then you will need to navigate to a repo that already has ether.js.

Then you will need to key in some commands.

First is to enable node.

Key in “node”.

Then, we have a variable, e , that require ethers.

So, I’ll copy this code.

Copy and paste it in from the slides.

Then, next we need to set the key.

I’ll just copy this key.

Copy and paste the “key=””” from the slides.

Then, we need to add the private key, that we just copied out from metamask.

Paste in the private key.

Then we will need to create a new wallet with the key.

Copy and paste the “wallet=…” from the slides.

Then next, we encrypt the wallet with a password.

So, lets copy this.

Copy and paste the “wallet.encrypt….” from the slides.

So now we enter in the password that we want for our wallet, inside the inverted commas.

So, for this demo, we will put password.

Then enter.

It will take a while.

Then yup, so, this is the wallet string that we want.

So, we will use this later.

OK, then now, let’s go back to the slides.

Go back to slides.

Then the last key that we want is forms.

This is where we put all the form structures inside and using these forms we generate our documents.

Click slide.

**[Set up for transferable document]**

OK, so, now we proceed to set up for the transferable document.

Move to VSCODE.

So, as you can see, over here, on the right side of the screen, I already have a template in place.

So, let’s go through what’s already inside.

The first thing you can see is the network, and for this demo, we will be using ropsten (rinkeby) testnet.

Then the next is the wallet.

So, here we are gonna paste in our string.

Go to terminal & copy the wallet string.

So, let me copy this.

And we will paste it inside here.

Go back to VSCODE & paste it inside.

Then next we have is the forms.

This field takes in an array of forms, and this is where we can have multiple forms.

OK, so let’s structure the form for our transferable document.

Click Slide.

**[Name]**

The first thing we need for the form is the name of the document.

This name will appear when you are selecting the document type.

Like over here.

Then ok let’s copy this.

And we paste it inside here.

Click slide.

**[Type]**

Then next is the type.

There can be only 2 types, either transferable record or verifiable document.

Since we are setting up for transferable document, we are gonna use transferable record.

So, let me copy this.

And paste it over here.

Click Slide.

**[Defaults]**

Next is the defaults.

So, what are these defaults for?

These defaults are settings that you have that are constant among the different documents of the same kind of file.

For example, you have eBL 1 and eBL 2, the contents of the forms will be different but there are some things that are the same, for instance, the document store address, company logo, signatures and the custom renderer, there can be more depending on your document.

So, these are just some examples.

But, they all will be the same throughout both the files.

So, in this case, they will be the defaults. Ok?

So, over in the code, we will need a defaults.

Enter “defaults:{}” in VSCODE.

Then the first thing we will be adding is the custom renderer.

You should already have this part as the prerequisite for this webinar.

OK, so, I’ll just copy it in.

Copy “$template” code chunk into VSCODE.

OK, next we need the issuers details.

This will be your data as you will be the one issuing the documents later on.

Let me copy the code.

Copy the “Issuers” chunk into VSCODE.

So just a quick look at what’s inside, we have the identity proof, which will point to your configured DNS.

Next is the name of your token registry and lastly, it’s your token registry address.

So, this makes up the issuer data.

Then, let me finish this defaults by adding in a name.

Copy “name” code into VSCODE.

With this, the defaults for this document is set.

Click slide.

**[Schema]**

Ok, next is the schema.

For the schema, we are using JSON schema.

For those who are not familiar with it, let me show u.

Open a new tab and enter “JSON SCHEMA”.

Select the first link.

This is JSON schema and if you are interested, you can read up more about it here.

OK, so let’s go back to the slides.

Click the slides.

For this schema, right, we will be using this chunk of code.­­

Let me copy it, then I’ll explain to you guys later.

Copy the chunk of code into VSCODE.

And paste it here.

OK, please take note, very important this part, the structure of this schema is derived from the structure of your document and must be in line with the custom renderer that you have already built for this document.

Click slide.

Meaning to say, if you take a look at this diagram over here.

This is the schema, whatever you have in the schema must be relevant in your document and your custom renderer.

So, if you take a look at the custom renderer, in this example, we are looking for the title and description.

So, if you have some other fields or missing one of these fields in the schema, it will not display at the custom renderer.

So, guys, please take note about this part ok.

So, for those of you who are wondering how to get this schema, it’s from when you are building the custom renderer.

You will know what fields are required, and from there you will get this schema.

Ok, a tool to help you in constructing this schema is this react JSON schema form playground.

Click the react JSON form playground tab

So, over here, this JSON schema section, this is the one that we require in our schema, ok?

So, let’s go back to the slides.

Ok, so with this, right, we are done with the schema, and let’s move on.

Go back to slide & Click slide.

**[Attachments]**

Ok, the last thing to be in the form is the attachments.

This attachments require 2 keys.

The first is an allow key, which takes in a Boolean, true or false.

This will allow or disallow the attachments in the form.

So, if you want to attach some documents, this should be set to true.

The next is the accept key, it will be a string for single file extension or an array or strings for multiple file extensions.

This will be the accepted file extensions, ok?

So, an example is, if you just want to allow a .pdf file, you just need to put .pdf as a string inside.

Like over here.

Then let’s say, you want to allow .png files also, you will then have to convert it to an array and just add it in like that.

OK, then, let’s say you want to allow all file extensions, you just have to remove this line from the attachments.

OK, so let’s copy this code inside.

Copy the “attachments” into VSCODE.

and, remember to save.

OK, so now we are done with the config file for the transferable document.

Let’s take a look at what we have created so far.

Go to Creator.tradetrust.io

So, on this page, I’ll select my file.

Select the config file.

Then key in my password.

Ah, remember this is the one that we just created, in the terminal.

Enter password.

Then as you can see, this is the name of the file that we just created.

Click “demo bill of lading” button.

Then, yes, this is the form, then once you fill it up you can issue it.

So, this is the set up for transferable document.

So, let’s go back to the slides.

Go back to the slide & click slide.

OK, so now let’s proceed to verifiable document.

**[Set up for verifiable document]**

Click slide.

Ok, for verifiable documents, its actually very much the same, the key differences are in the form type and the document store in the issuers section.

Ok, let’s take a look at how it will look like in the code.

Go to VSCODE.

Go to verifiable document file.

This is the structure of the verifiable document.

And if you take a closer look at the type.

Highlight the type.

We changed it to verifiable document.

Then, the other key difference, is the document store in the issuers.

Highlight the “documentStore”.

We replaced the token registry because we do not need it anymore.

Ok, so let’s go through one more time what are the fields we have.

First, we have the name of the document.

Highlight the “name”.

Then, the type as verifiable document.

Highlight the “type”.

Then, the defaults.

Highlight the “defaults”.

In the defaults, we have the custom renderer.

As you can see, its pointing to another custom renderer, because this is another document.

Then next, the issuers, with the updated document store.

Then, if you can see, this document has more defaults.

It has a name, logo, title and descriptions.

For this document, if you want to change the logo, you just have to change the URL.

Highlight the logo URL.

Then, for the title and description, these are the default values for the form.

If you look at the schema, you will see the same title and description, under the object properties.

Highlight the title and description in the schema.

This will tell the document creator that there are some default values to use for this document.

OK?

Then moving on, the next is the schema.

So, it’s the same as before, it will need to be in sync with your custom renderer and document.

Then lastly, the attachments.

Ok, so, I’ll copy this.

Copy the chunk of code.

Go back to the starting-config.json file.

So, in the same config file, you can have as many forms as you like, all you have to do is to add it in, like this.

Paste the code in the forms.

Ok, so with the code pasted in, our verifiable document is ready.

Click slide.

Now let’s move on to the next part, which is the issuing of the document.

[Issue document]

Ok, so now we head over to creator.tradetrust.io and let us load in our new config file.

Load the new config file.

Let me enter my password.

It will take some time.

Ok, but once it done, we will have to create a verifiable document and issue it.

Enter the password.

OK, for this part we will use the verifiable document.

Which is the demo cover letter.

Select the cover letter.

Then, over here, we will just need to fill up the form.

But as you can see there are some values already inside.

Like I mention before, these are the default values in the defaults section of the config file.

OK, let’s use them.

And, Ok, now let’s attach some attachments.

If you can see, on your right, the config file, we have accepted .pdf and .png files, so what happens if you attach something other than these 2 file types?

OK, let me show u.

Firstly, if you click this browse file.

Click browse file.

You will not be able to select anything other than .pdf and .png.

Then what if you drag and drop the file.

Drag and drop a file with wrong extension.

You will be prompted with this error, that shows you the accepted file types.

Ok, so, let’s attach some proper files.

Attach .pdf file.

Attach .png file.

So, now once everything is done, we are good to go.

Ok now let’s click this issue document button.

Click the issue document button.

Ok, this part will take some time.

Click to the successful tab.

So, once it is done, we will be able to see this page.

You can download it by clicking the button here.

So, we will be downloading it.

And we will be verifying it later on, ok?

Click the download button.

Ok, so let’s go back to the slides.

Click back to slide.

So that’s how we issue a document.

Click slide.

**[Verify document]**

Then ok, next we will see how we can verify the document.

So, this is mostly for someone who receives your document.

Click slide.

For this, if you are using mainnet, u can go to tradetrust.io to verify the document.

Then if you are using rinkeby testnet, you can go to rinkeby.tradetrust.io to verify the document.

For our case, we are using ropten testnet, so we will go to dev.tradetrust.io to verify our document.

Ok, so let’s head over there.

Go to dev.tradetrust.io.

Once you are here, you can just drop the file into the drop zone and it will verify the document.

If there are any errors, it will show the errors over here.

Then if everything is ok, we will proceed to the custom renderer page.

Drop the .tt file into the drop zone.

OK, so, yup, as you guys can see, everything is loading as it should be.

Over here we have a few indicators.

It shows that the document has not been tampered with, document has been issued, and document issuer has been identified.

Over here, in the custom renderer, we have the logo, then this is the title and description that we have.

It also has our attachments.

So, you can click the attachments.

You can even download them if you want to.

You can also preview the attachments if it is a .pdf file.

So ok, we are done with the verifying of the document.

And let’s go back to the slides.

Click slides.

**[Useful links]**

Ok, useful links.

Click slides.

Ok, so as mentioned previously.

These are the list of the useful links for what we have just done.

I’ll leave it here for your references.

And with this I end my presentation.

Thank you for your time and we will move on to the Q & A.