Documentatie rift

**Wat is rift**

Rift zorg ervoor dat banken kunnen communiceren zonder dat ze elkaars gegevens hoeven hebben en zonder dat ze elkaars encryptie keys moeten hebben. Het enige wat de bank hoeft te weten zijn de gegevens van rift zelf.

**Welk in welk formaat moet ik naar rift sturen**

rift communiceert met xml, een voorbeeld van xml tekst dat naar rift gestuurd word ziet er zo uit:

```
<ATM_Request><actionType>withdraw</actionType><pasnumber>30192283</pasnumber><afzende
r>070001</afzender><banknumber>070002</banknumber><pincode>1234</pincode><amount>100<
/amount><automaatNr>1</automaatNr><errorNote> </errorNote></ATM_Request>
```

het base element van de xml tekst is altijd ATM_Request, daarin zit altijd een een actionType, de actionType bepaalt waar het bericht voor is en wat er verder in zal staan.
Er zijn momenteel:

- Withdraw, hierop antwoord een server met het actionType withdrawconfirm
- getSaldo, hierop antwoord een server met het actionType saldoReply
- login, hierop antwoord een server met het actionType loginReply

in elk van deze types zit ook:

- banknumber: het ruban nummer van de ontvangende bank
- errorNote: de afgesproken errors

in login,getSaldo en withdraw zitten ook:

- pasnumber: het uid van de pas
- pincode
- afzender: het ruban nummer van de afzender
- automaatNr: we hebben niet meerdere apparaten dus het is altijd 1

verder heeft withdraw een amount: het bedrag
in saldoReply betekent amount het bedrag dat op de rekening staat
bij withdrawConfirm, saldoReply, loginReply het element succes die true of false kan zijn

de xml tekst mag elementen bevatten die er niet in horen, dat maakt het maken ervan eenvoudiger.

Nog wat meer voorbeelden:

```
<ATM_Request><actionType>loginReply</actionType><succes>true</succes><pasnumber>aed20
bc9</pasnumber><banknumber>070001</banknumber>errorNote> </errorNote></ATM_Request>

<actionType>loginReply</actionType><succes>false</succes><pasnumber>aed20bc9</pasnumb
er><banknumber>070001</banknumber><errorNote>BAD_PIN</errorNote></ATM_Request>

<ATM_Request><actionType>withdrawConfirm</actionType><succes>true</succes><pasnumber>
aed20bc9</pasnumber><banknumber>070001</banknumber<errorNote>
</errorNote></ATM_Request>
```

<ATM_Request><actionType>getSaldo</actionType><pasnumber>aed20bc9</pasnumber><afzender>070001</afzender><banknumber>070001</banknumber><pincode>1234<pincode><errorNote></errorNote></ATM_Request>
<ATM_Request><actionType>saldoReply</actionType><succes>true</succes><pasnumber>aed20bc9</pasnumber><banknumber>070001</banknumber><amount>37474.00</amount><errorNote/></ATM_Request>

**Welke encryptie gebruikt rift**

rift maakt momenteel gebruik van AES in CBC modus, de code van de AES klasse zullen we delen. Waarschijnlijk komt er later een nieuwe versie die beveiligd is tegen replay attacks.

**Wat doet rift**

Vanbinnen werkt rift zo:

1. Als rift een request binnenkrijgt word deze verwerkt in een nieuwe thread
2. Het afzender ip adress word afgelezen, vervolgens word er in de database gekeken welke bank bij dat ip hoort en hoe het bericht gedecrypt moet worden.(er word altijd gecheckt of het inkomende ip wel klopt met het ip dat bij de bank hoort);
3. Als het om een withdraw gaat word alles opgeslagen in de database
4. De inhoud van het bericht word geencrypt met de sleutel van de andere bank
5. Het bericht word naar de andere bank gestuurd
6. Als er een antwoord komt word dit antwoord gedecrypt en weer geencrypt voor de andere bank. (als het een withdrawConfirm is word gelogt in de database dat het gelukt is)
7. Het bericht word doorgestuurd naar de bank waarvan het request afkomstig was

De code van rift word gedeeld

**Wat moet rift van mijn bank af weten**

In de tabel banken van rift staat:

- Banknaam
- Banknummer
- Ip adress
- Poortnummer
- Aes key(16 characters lang)

Deze gegevens moeten eerst naar ons gestuurd worden voor rift te gebruiken is.

**hoe stuur ik het naar rift**

dit is voorbeeld code om een request door te sturen naar een ander bank. De klasses riftConnector en AES krijgen jullie.

```
AES aes =new AES();
        RiftConnector rift = new RiftConnector("145.24.222.69",8501);
        String
message="<ATM_Request><actionType>login</actionType><pasnumber>30192283</pasnumber><afzender>10</afzender><banknumber>01</banknumber><pincode>hrZAupOCJ597BYtPOxj03Q==</pincode><automaatNr>1</automaatNr><errorNote>leave this area for error messages or notes</errorNote></ATM_Request>";
```

```
        String encryptedmessage = new AES().encrypt(message,"testtesttesttest");
        System.out.println("server said: " +
aes.decrypt(rift.sendToRift(encryptedmessage),"yourkey16chars16"));
```

Voor het ontvangen van berichten kan je het beste dezelfde poort gebruiken als die je gebruikt om je berichten van je bankautomaat te ontvangen.

Het rift ip is 145.24.222.69 poort 8501