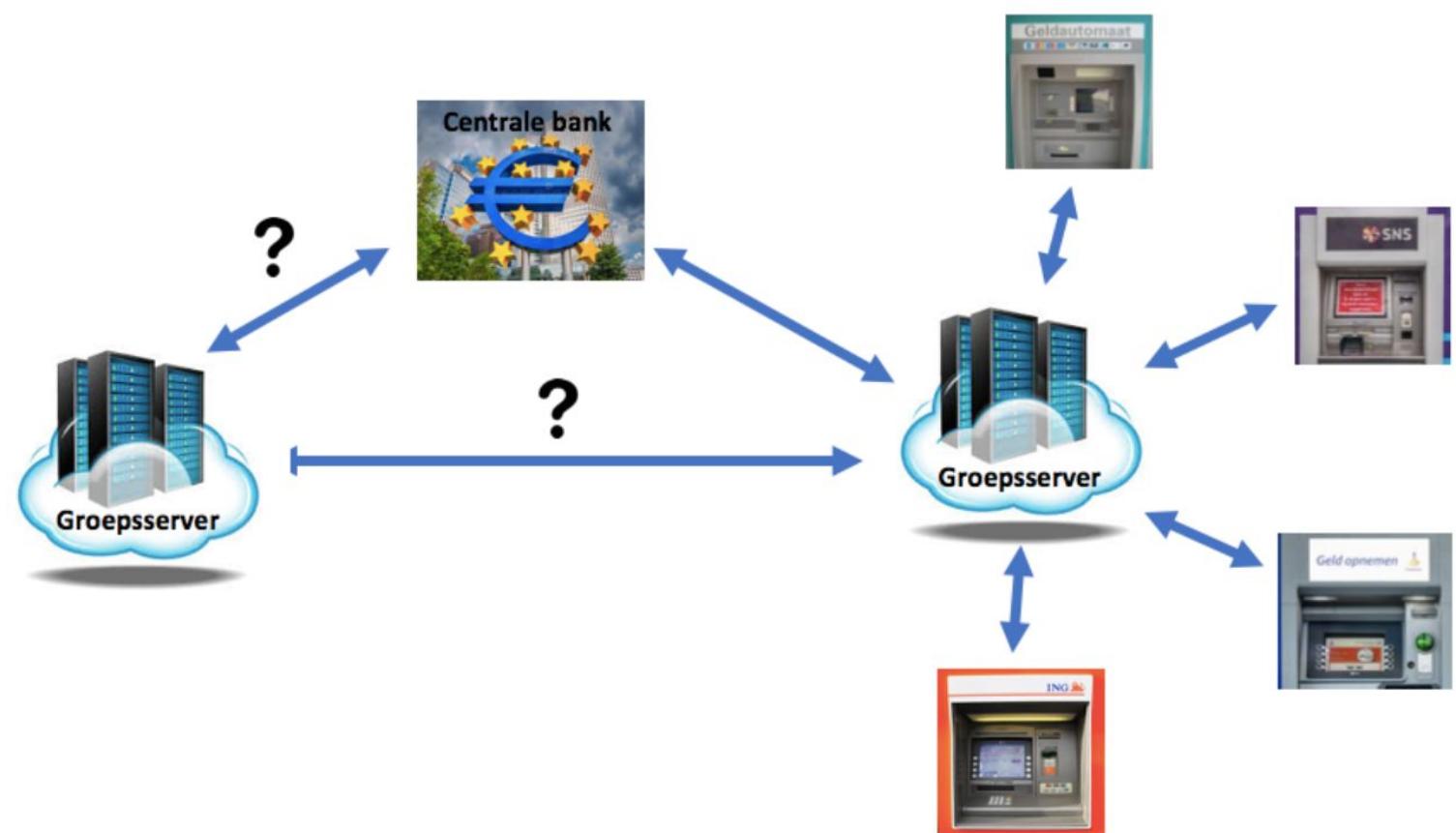


rapport project 4: Bankalicious!



door: Owen Braber
studentnummer: 0946167
byte groep: 8
klas: TI1C

inleiding	3
analyse	3
eisen	3
man in the middle attack	4
manieren beveiligen	5
manier 1: certificaten en SSL	5
manier 2: IP whitelist	5
servers verbinden	5
manier 1: een rij van servers	5
manier 2: een web van servers	6
manier 3: tussenpersoon	7
voordelen en nadelen	7
oplossing	8
servers verbinding oplossing	8
beveiligen of versleutelen oplossing	9
conclusie	9
ontwerp	10
netwerkdiagram	10
data flow diagram	11
toepassen van beveiliging	12
eindconclusie	12
riscolog	13
issue tracker	14

inleiding

Tijdens project 3 waren er servers waar de groep mee kon communiceren om informatie te halen uit hun database. Hierdoor kon een automaat een kaart van die database lezen. Maar een andere server kon niet die kaarten lezen. Hierdoor kan een kaart alleen worden gebruikt bij een automaat waar die server aan zit verbonden. Met de centrale bank wordt dit anders. Met de centrale bank kan ieder groep, die eraan zit gekoppeld, informatie halen uit de andere groeps servers. Hierdoor kan een kaart bij meerdere automaten worden gebruikt.

De centrale bank kan op verschillende manieren worden ingericht. Dit rapport is er om te kijken wat de beste manier is om de centrale bank in te richten. Hiervoor stellen we eerst een paar eisen waar de centrale bank aan moet voldoen, daarna kan gekeken worden naar verschillende oplossingen voor de centrale bank. Als er éénmaal een oplossing is, kan er gekeken worden naar hoe die oplossing eruit komt te zien. Eerst zal dus een analyse gemaakt moeten worden van de verschillende oplossingen.

analyse

Voor de beste oplossing voor het inrichten van een centrale bank, moeten we natuurlijke eisen opstellen. Ook moeten we natuurlijk verschillende opties zoeken. Ik zal door middel van het opstellen van eisen kijken naar de beste manieren om deze eisen te behalen.

eisen

De inrichting van de centrale bank moet natuurlijk aan verschillende eisen voldoen. De inrichting moet natuurlijk aan eisen voldoen die gaan over het eindresultaat van de centrale bank en over het gebruik ervan.

De eisen die zijn opgesteld zijn geprioriteerd met behulp van de MoSCoW methode:

- M: must
- S: should
- C: could
- W: would

De eisen voor het eindresultaat luiden als volgt:

- de verbinding tussen de verschillende servers moet beveiligd zijn. Dit is om zogenaamde "Men in the middle" attacks te voorkomen.(M)
- Bij de oplossing moet worden gedacht aan uitbreiding of verkleining. Het toevoegen of verwijderen van servers moet makkelijk zijn.(M)

De eisen voor het gebruik luiden als volgt:

- De centrale bank moet vertrouwelijk zijn, klanten moeten de centrale bank kunnen vertrouwen met hun gegevens.(M)
- De centrale bank moet effectief werken.(M)

Verder zijn er nog een paar niet-functionele eisen, deze eisen zijn allemaal optioneel:

-Er is een logbestand wat bijna al het verkeer tussen de servers logt (C).

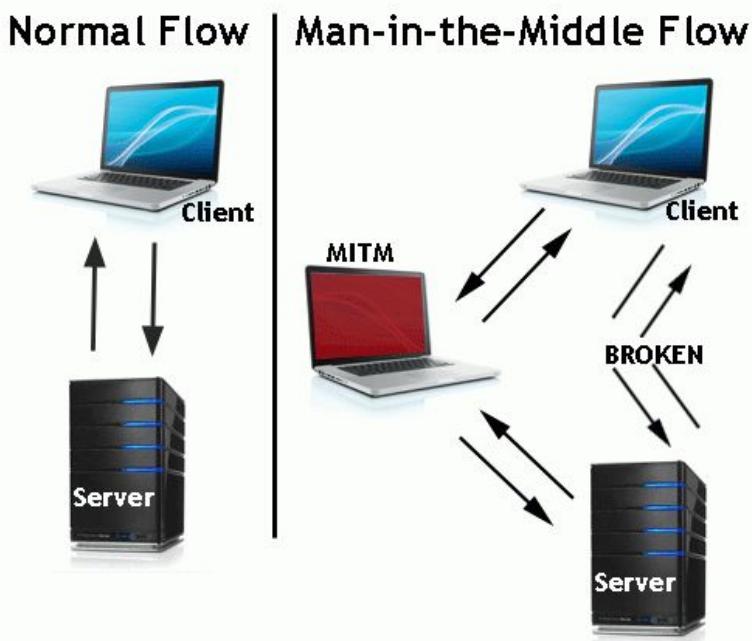
-Alle servers die zich bij de centrale bank aansluiten, moeten laten zien dat ze bij de centrale bank horen op hun pinautomaten (W).

Op deze eisen zal ik de beste oplossing vinden, door middel te kijken naar verschillende manieren. Ook zal ik uitleggen wat sommige onderwerpen nu precies inhouden.

Bijvoorbeeld, wat is een "Man in the middle" attack?

man in the middle attack

een man in the middle (of MITM afgekort) attack is een manier van het krijgen van data. Een hacker zal tussen 2 verbonden apparaten gaan zitten. Dit doet de hacker door middel van voor te doen als een soort relay. Hierdoor zal de data altijd eerst langs de hacker gaan (zie afbeelding hieronder).



Dit zal ervoor zorgen dat de data, die bijvoorbeeld de groeps servers willen versturen in handen komt van de verkeerde mensen. Dit is zeker gevaarlijk als we praten over bankgegevens. Het voorkomen van een MITM attack is door de identiteit van de ontvanger/zender te controleren. Dit kan op verschillende manieren en elke manier heeft zijn voordeelen en nadelen.¹

Hieronder zullen we kijken naar verschillende manieren en ook kijken wat ze allemaal inhouden.

¹ bron: <https://www.veracode.com/security/man-middle-attack>

manieren beveiligen

Er zijn verschillende manieren om de connectie tussen servers te beveiligen en ze te versleutelen. Hiermee zorg je dat de data die je stuurt naar een ander server veilig blijft en niet door iemand anders gelezen kan worden. We zullen kijken naar twee verschillende manieren om de connectie te kunnen beveiligen en/of ze te kunnen versleutelen.

manier 1: certificaten en SSL

Wat is SSL en wat zijn SSL certificaten, en wat heeft het voor invloed op de verbinding tussen servers? SSL is een protocol wat een versleutelde verbinding maakt met een andere server, zodat het verkeer tussen de twee servers niet gelezen/gebruikt kan worden door mensen van buitenaf. Alleen een SSL verbinding heeft een SSL certificaat nodig (van de server waar je mee wilt connecten). Dit certificaat bevat een public key en een private key. Deze keys zorgen ervoor dat alleen servers met hetzelfde certificaat (wat van de server afkomt waar een connectie mee wordt gemaakt) kunnen verbinden. Dit zorgt ervoor dat je controle hebt in wie er verbinding maakt en dat je weet dat de data wordt versleuteld.²

manier 2: IP whitelist

Een IP whitelist is een lijst met IP adressen. De IP adressen die op deze lijst staan zijn de enige IP adressen die met de server een connectie kunnen maken. Hierdoor heb je controle over wie er met je server kan connecten en wie de data uit de server mag gebruiken. Dit zorgt ervoor dat een MITM attack wordt tegengehouden.³

Nu we gekeken hebben naar de manieren om connecties te beveiligen en/of te versleutelen, is het nu tijd om te kijken naar manieren om servers met elkaar te verbinden.

servers verbinden

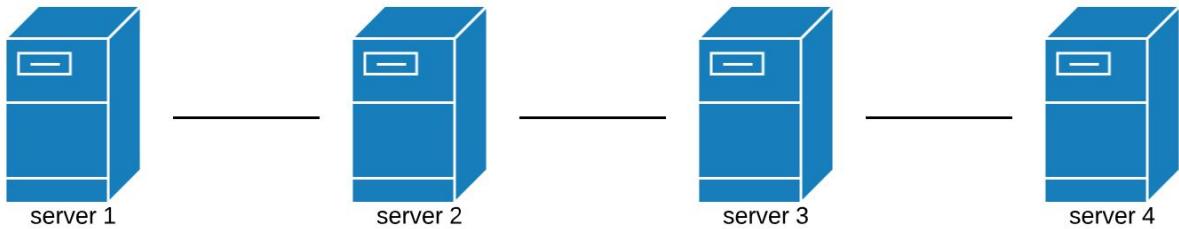
Er zijn verschillende manieren om de servers met elkaar te verbinden. Sommige manieren zijn beter dan anderen. We zullen kijken naar drie verschillende soorten manieren om servers met elkaar te verbinden.

manier 1: een rij van servers

Een manier om alle groeps servers met elkaar te connecten is als volgt. Een groeps server is verbonden met maar 1 andere groeps server en die server is dan weer verbonden met een andere server, zodat er een soort van lijn ontstaat (zie afbeelding).

² bron: https://www.sslcertificaten.nl/support/Terminologie/SSL_Certificaten

³ bron: <https://security.stackexchange.com/a/124716>

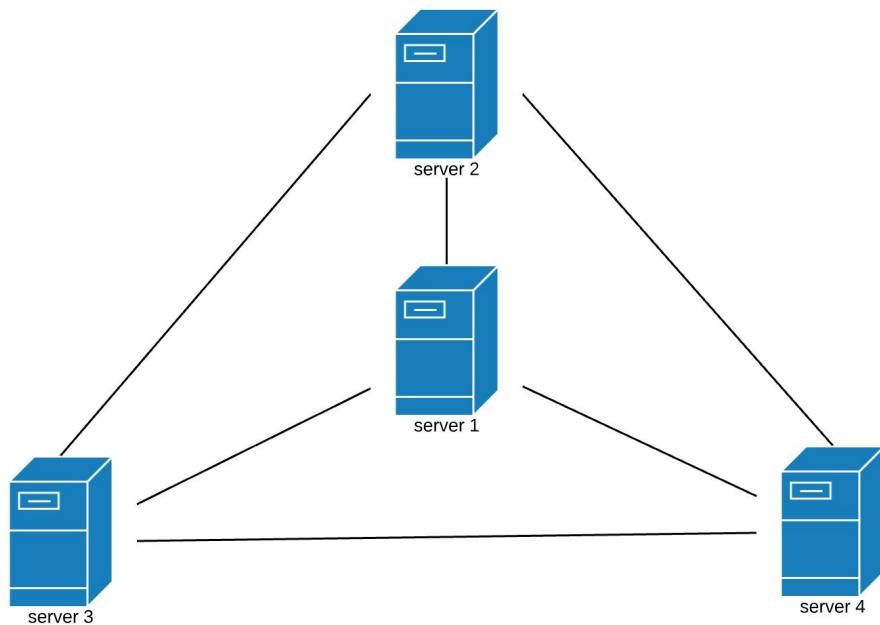


Bijvoorbeeld, als een gebruiker probeert in te loggen met een kaart zal de server een verzoek sturen naar de twee aanliggende servers.

Als een server dit verzoek krijgt, zal de server in zijn database kijken of de doorgegeven kaart erin staat. Als de kaart niet in de database staat, zal de server het verzoek doorsturen naar de volgende server. Als de kaart wel in de database staat, zal de server iets terugsturen naar de server (waar het verzoek origineel vandaan kwam) waarmee de server weet in welke database de kaart staat. Als de server namelijk weet in welke database de kaart zal, zal dit de communicatie makkelijker maken.

manier 2: een web van servers

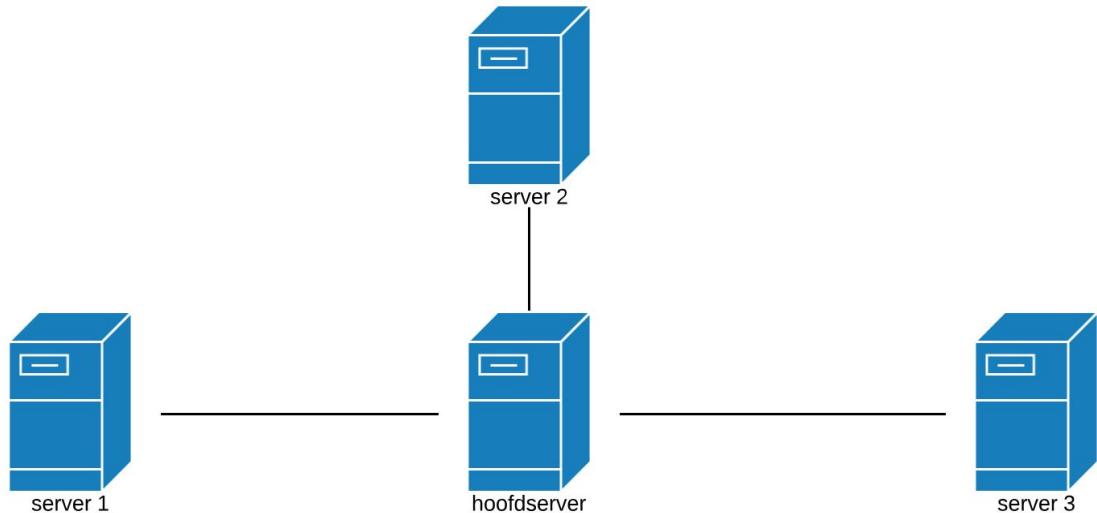
Een andere manier om de groeps servers met elkaar te laten communiceren, is om ze allemaal met elkaar te verbinden. Hierdoor ontstaan er een soort web van servers (zie afbeelding hieronder).



Hierdoor kan elke groeps server makkelijk communiceren met alle andere groeps servers. Als, bijvoorbeeld, iemand probeert in te loggen met een kaart. Zal de server aan alle servers vragen of de kaart in hun database staat. Als dat zo is, zal hij die server gebruiken om mee te communiceren deze inlogsessie.

manier 3: tussenpersoon

Een andere manier om de groeps servers met elkaar te laten communiceren, is om een “tussenpersoon”, oftewel een hoofdserver, er tussen te zetten (zie afbeelding hieronder).



Communicatie tussen de servers werkt dus met een hoofdserver, deze hoofdserver zal al het verkeer tussen de servers regelen. Bijvoorbeeld, als een gebruiker probeert in te loggen met een kaart. Zal de server een verzoek sturen naar de hoofdserver om te kijken in welke database de kaart staat. De hoofdserver zal dan alle servers langsgaan en vragen of de kaart in hun database staat. Als de hoofdserver de juiste server heeft gevonden, zal de hoofdserver iets terugsturen, naar de server die het verzoek stuurde, waarmee de server weet met welke andere server hij moet communiceren. Als de server een ander verzoek wilt sturen naar de andere server, stuurt hij bijvoorbeeld het ID van de server naar de hoofdserver.

voordelen en nadelen

Hieronder ziet u een tabel van alle manieren van beveiligen en manieren om servers te verbinden en daarnaast ziet u alle voor- en nadelen.

manier	voordelen	nadelen
certificaten en SSL	-makkelijk instellen, alleen het certificaat van de server nodig	-met meerdere servers connecten betekent meerdere certificaten

	-verbinding versleuteld	bemachtigen
IP whitelist	-makkelijk instellen, alleen de IP-adressen nodig	-IP kan worden veranderd, zodat men data naar de server kan sturen -verbinding wordt niet versleuteld
een rij van servers	-geen tussenpersoon/hoofdserver	-er kunnen twee groepen ontstaan als er een server uitvalt -toevoegen of verwijderen van een server kan lastig zijn
een web van servers	-geen tussenpersoon/hoofdserver -communiceren is makkelijker, want elke server staat verbonden met alle andere	-toevoegen of verwijderen van een server kan lastig zijn, want iedere server moet worden aangepast
tussenpersoon	-elke server (behalve de hoofdserver) is maar verbonden met één server -verkeer kan makkelijk vast gelegd worden -makkelijk om servers toe te voegen of te verwijderen, alleen de hoofdserver moet worden aangepast	-als de hoofdserver offline gaat, kunnen de servers niet meer communiceren

Nu we manieren hebben bekeken voor het beveiligen en/of het versleutelen van de verbindingen en manieren hebben gezien om servers met elkaar te verbinden. Is het nu tijd om te kijken welke manier het beste is en welke manier het beste past bij de eisen die zijn opgesteld.

oplossing

De manieren die zijn beschreven kunnen helpen bij het vinden van de beste oplossing. Deze oplossing moet passen bij de eisen die zijn opgesteld eerder in dit rapport.

servers verbinding oplossing

Er zijn drie manieren beschreven voor het verbinden van de groeps servers. We zullen kijken welke manier het beste is kijkend naar de eisen die zijn opgesteld en ook naar hun voor- en nadelen.

De manier: een web van servers, voldoet niet aan de eisen, want de uitbreidbaarheid is niet makkelijk. Je moet hiervoor de nieuwe server aan elke andere server koppelen en servers die niet meer meedoen verwijderen van alle servers. Ook kan de hoeveelheid connecties problemen zorgen voor de snelheid van de server.

De manier: een rij van servers, voldoet aan een deel van de eisen. Maar als er een bank bijkomt, moet je de aanliggende server(s) aanpassen. Ook als er een bank weg gaat moet je meerdere servers aanpassen. Dit heeft impact op de uitbreidbaarheid.

De enige manier die nog overblijft is dat er een hoofdserver tussen de communicatie gaat zitten. Dit zal ervoor zorgen dat alle servers zullen praten met alleen de hoofdserver en als een server met een andere server wilt praten, dit via de hoofdserver gaat. Doordat alleen de hoofdserver praat met alle andere servers, is uitbreiding makkelijk. Je hoeft hiervoor alleen de hoofdserver aan te passen en niet alle andere servers. Ook kun je gemakkelijk al het verkeer vast leggen op de hoofdserver, want het gaat allemaal via de hoofdserver.

beveiligen of versleutelen oplossing

Er zijn twee manieren beschreven voor het beveiligen of versleutelen van het verkeer tussen servers. We zullen kijken welke manier het beste is kijkend naar de eisen die zijn opgesteld.

De twee manieren die benoemd waren zijn, certificaten en SSL en een IP whitelist. IP whitelist is makkelijk in te stellen maar het zorgt er niet voor om een versleutelde connectie. Het gebruik van certificaten en SSL zorgt ervoor dat de connectie wordt beveiligd en het zorgt ervoor dat het verkeer wordt versleuteld. Ook is, doordat de verbinding versleuteld en beveiligd is, de centrale bank vertrouwelijk. Zo hoeven de klanten, die zich bij de centrale bank aansluiten, geen zorgen te maken over dat hun data kan worden gestolen.

conclusie

De beste manier om servers met elkaar te verbinden, is om er een hoofdserver tussen te zetten. Dit zorgt ervoor dat al het verkeer via de hoofdserver gaat. Ook is het zo dat het makkelijk is om uit te breiden, want je hoeft alleen de hoofdserver aan te passen. Lettend hierop is het beveiligen en versleutelen van het verkeer met behulp van SSL en certificaten een goede oplossing. Dit zorgt er namelijk voor dat het verkeer wordt versleuteld en wordt beveiligd met certificaten. En uitbreiding is nog steeds makkelijk hierdoor, want het enige certificaat wat nodig is, is degene van de hoofdserver zelf.

Maar waarom zouden we maar één oplossing hebben voor het beveiligen? Waarom niet twee? Het is beter om ook een IP whitelist in te stellen op de hoofdserver, hierdoor is er extra beveiliging voor de data die op de servers staan. Hierdoor is het vertrouwen van de centrale bank verbeterd, want klanten, die zich aansluiten bij de centrale bank, hoeven dan geen zorgen te maken dat hun klantgegevens kunnen worden gestolen.

ontwerp

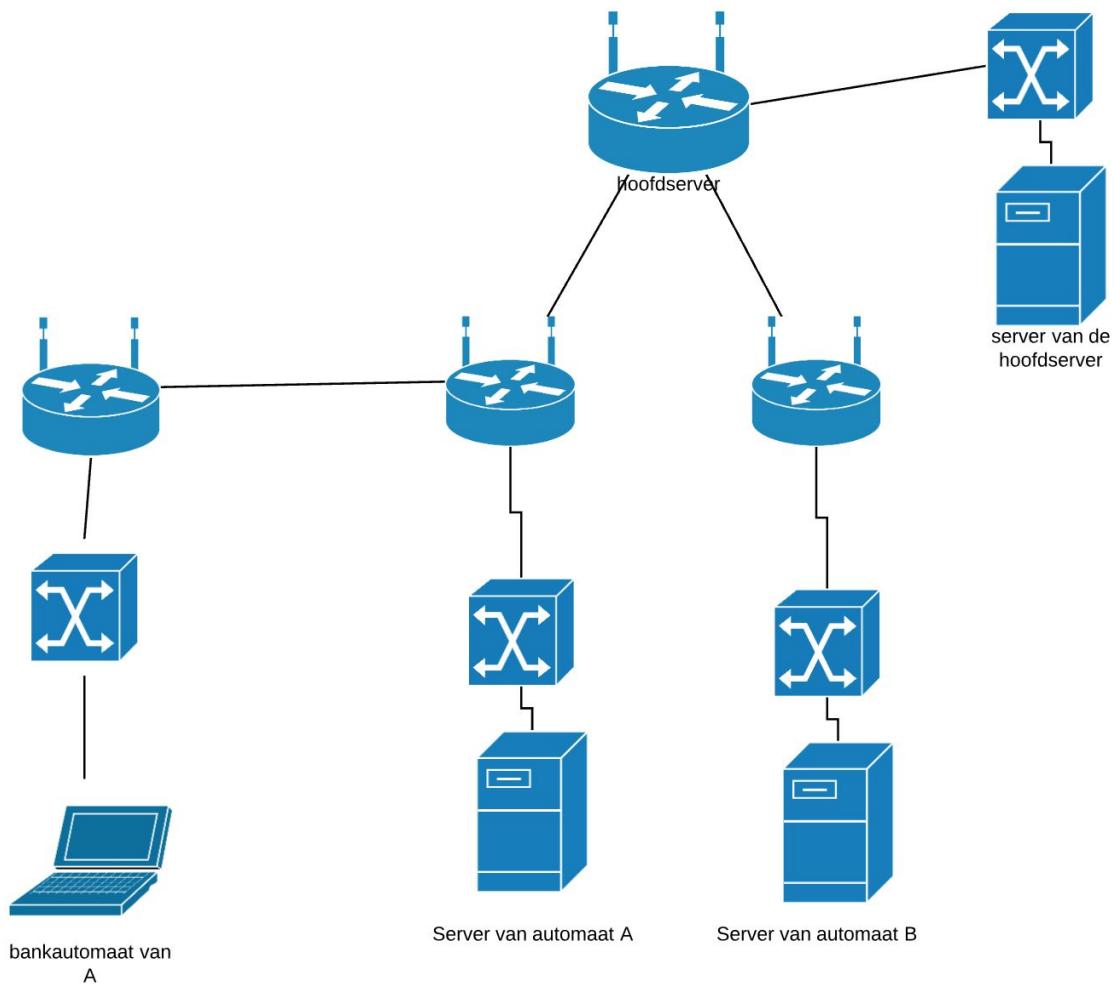
Nu dat we een manier hebben om de groeps servers met elkaar te laten communiceren, kunnen we kijken hoe we dit gaan ontwerpen.

Dit zal ik laten zien met behulp van een netwerkdiagram, dat zal laten zien hoe de servers aan elkaar gekoppeld zijn. Ook zal ik een data flow diagram laten zien, hierin kun je dan zien hoe de informatie door de servers gaat. Ook zullen we met het data flow diagram een voorbeeld laten zien van hoe de hoofdserver communiceert.

netwerkdiagram

Om een goed beeld te hebben van de connecties kunnen ontstaan, is een netwerkdiagram maken wel slim. Hierdoor kun je in één keer zien hoe de servers met elkaar staan verbonden en zo is het ook makkelijker om een data flow diagram te maken.

In dit netwerk diagram staan alleen de servers verbonden met de hoofdserver, niet de bankautomaten zelf.

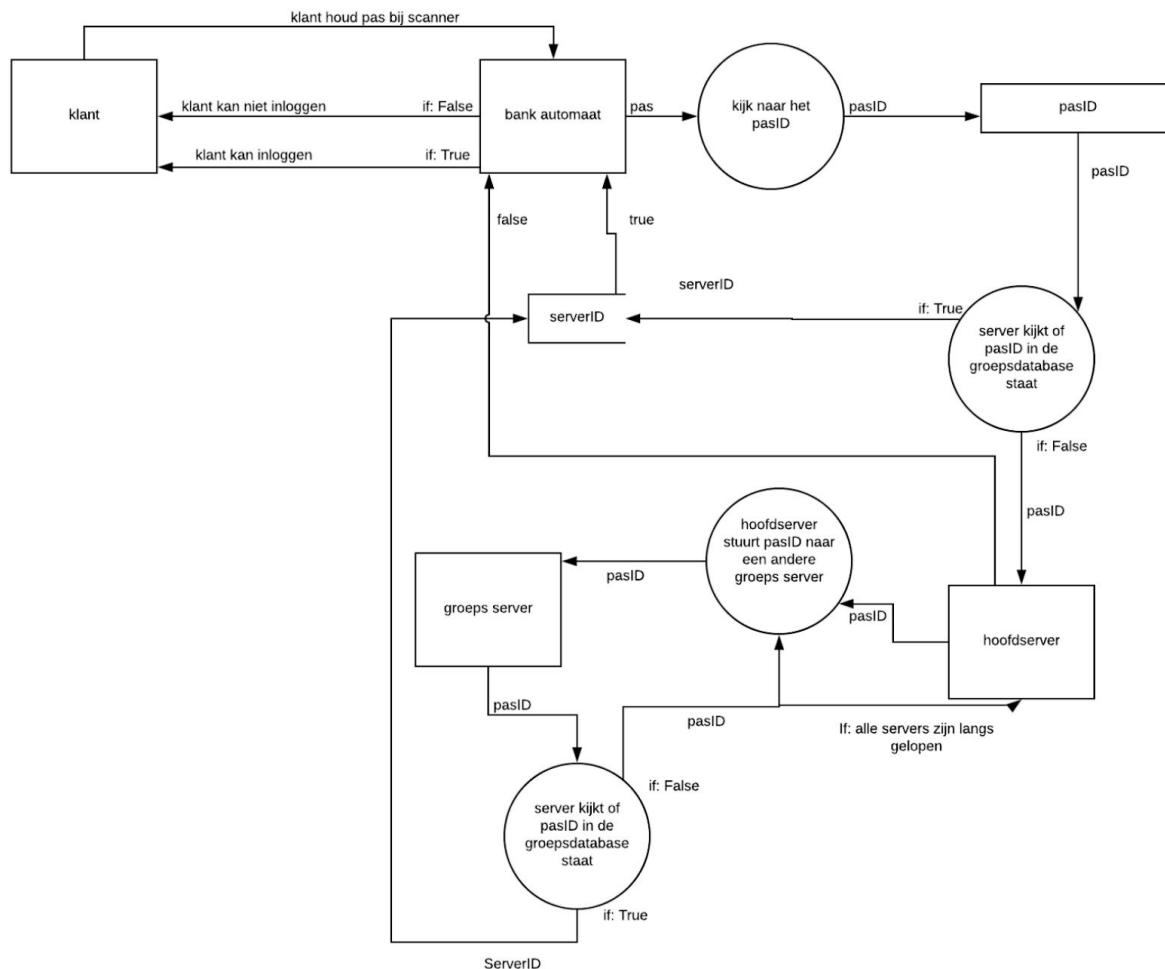


Nu we het netwerkdiagram hebben gemaakt, kunnen we kijken naar hoe de data door het systeem gaat.

data flow diagram

een data flow diagram laat zien hoe data zich door een systeem/netwerk stroomt. Als voorbeeld zullen we kijken wat er gebeurt als een klant een kaart voor de pas scanner houdt.

Hieronder ziet u het data flow diagram.



Het diagram laat ook zien, dat het ontwerp effectief werkt. Want de hoofdserver gaat langs alle groeps servers, dit doet de hoofdserver maar één keer. Hierdoor stuurt hij niet constant verzoeken naar alle servers, wat ervoor zorgt dat het verkeer sneller gaat en ook dat je sneller data krijgt van de hoofdserver.

toepassen van beveiliging

De oplossing die werd gekozen werkte ook met SSL en met een IP whitelist, maar hoe wordt deze toegepast in het ontwerp. Dat zal ik nu laten zien.

Doordat we een hoofdserver hebben, is het makkelijk om deze twee manieren toe te passen. Om een IP whitelist aan te maken, hoef je alleen maar een lijst aan te maken in de hoofdserver met alle vertrouwde IP adressen. Telkens als de hoofdserver een connectie maakt, zal hij kijken of het IP (van de server die met de hoofdserver een verbinding wilt maken) in de lijst staat. Als dit niet het geval is, zal de hoofdserver de verbinding weigeren of stoppen. Als het IP wel in de lijst staat, zal de hoofdserver de verbinding toestaan.

SSL en de certificaten implementeren is ook makkelijk met een hoofdserver, omdat je alleen het certificaat van de hoofdserver nodig hebt. Het enige wat je hoeft te doen, is het maken van een certificaat van de hoofdserver en het certificaat beveiligen met een wachtwoord. Als er een server bij de centrale bank komt, hoeft die server alleen het certificaat en het wachtwoord te hebben om met de server een verbinding te maken. Want als een server probeert zonder het certificaat een verbinding te maken met de hoofdserver, zal de verbinding worden geweigerd.

Deze twee manieren van beveiligen zorgen ervoor dat de klant de centrale bank vertrouwt. Dit komt doordat ze weten, dat het verkeer wordt versleuteld door SSL en doordat alleen vertrouwelijke servers kunnen connecten met de hoofdserver. Dit zorgt er dan uiteindelijk voor dat niemand het verkeer kan afluisteren.

eindconclusie

Hier volgt de conclusie van dit rapport. We zullen terugkijken op de eisen die we hebben gesteld en de oplossing die we bedacht hebben en hoe die oplossing voldoet aan de eisen.

De centrale boek moest aan eisen voldoen die het eindresultaat beschrijven en die het gebruik beschreven. Hieronder nogmaals alle eisen.

- de verbinding tussen de verschillende servers moet beveiligd zijn.
- Bij de oplossing moet worden gedacht aan uitbreiding of verkleining.
- De centrale bank moet vertrouwelijk zijn.
- De centrale bank moet effectief werken.

De oplossing die deze eisen volgt, is het gebruik van een hoofdserver. Deze conclusie is ontstaan toen we gekeken hebben naar verschillende manieren van het verbinding van servers en deze van elkaar afgewogen. Deze oplossing voldoet aan de eisen die het kan halen. Zo voldoet het aan "Bij de oplossing moet worden gedacht aan uitbreiding of verkleining", want om een server toe te voegen of te verwijderen hoeft alleen de hoofdserver aangepast te worden.

Ook voldoet het aan “De centrale bank moet effectief werken”, want de hoofdserver stuurt niet constant bericht naar alle servers. De hoofdserver doet alleen het minimale, zodat er sneller data wordt uitgewisseld.

Een oplossing voor het beveiligen is er ook. Er zijn twee manieren beschreven van het beveiligen van de verbinding of het versleutelen van de verbinding. Omdat meer beveiliging beter is, is er gekozen voor alle twee de oplossingen. De twee manieren van beveiligen waren een IP whitelist en het gebruik van SSL met certificaten. Deze oplossing voldoet ook aan de eisen die het kan halen. Zo voldoet het aan “De verbinding tussen de verschillende servers moet beveiligd zijn”, want met een IP whitelist en certificaten zorg je ervoor dat alleen vertrouwde servers met de hoofdserver kunnen verbinden. Ook zorgt SSL ervoor dat de verbinding wordt versleuteld, zodat niemand het verkeer kan afluisteren. De oplossing voldoet ook aan “De centrale bank moet vertrouwelijk zijn”, doordat de data, die op de servers staan, wordt beveiligd. Zorgt dit ervoor dat groepen, die zich bij de centrale bank aansluiten, erop kunnen vertrouwen dat hun data beveiligd is.

De oplossing voldoet dus aan alle eisen die eerder zijn opgesteld, waardoor het een goede manier zou zijn voor het inrichten van een centrale bank.

Hieronder staan nog een risicolog en een issue tracker. Deze laten de problemen en risico's zien die tijdens het project zijn opgedoken.

riscolog

hieronder staan alle risico's van over het inrichten van de centrale bank en wat de risico's kunnen zijn als het werkend is.

#	Risico Beschrijving	Kans	Impact	Risico*	Maatregel		Status Omschrijving	Datum
R1	laptop raakt kapot, waardoor de documenten weg zijn	1	5	5	documenten opslaan en op een drive/gitlab zetten	:)	Documenten worden geschreven op google drive en elke paar keer op gitlab gezet	09-05-18
R2	de centrale bank valt uit	2	5	10	server automatisch oplaten starten als hij uitvalt. Er moet ook altijd iemand bij de server zijn voor onderhoud	:)	In gedachte gehouden voor later als de centrale bank wordt gemaakt	23-05-18

Kans: schaal 1 (klein) t/m 5 (zeer groot)

Impact: schaal 1 (zeer lage) t/m 5 (zeer hoge)

Risico: = kans * impact

: [status] :| opgelost; :| bezig; :(niet opgelost; N nieuw

issue tracker

hieronder staat alle issues die zijn tegengekomen tijdens het project.

#	Datum In	Issue	Verantwoordelijk)	😊	Status Log	
					Datum	Beschrijving
J 1				😊		

😊: [status] :) opgelost, :| bezig; :(niet opgelost; N nieuw