Quantumsleutels veilig delen

Op deze website is al vaak iets geschreven over quantumcomputers. Ook in het nieuws komt deze term regelmatig terug. Quantumcomputers zijn de heilige graal waar veel onderzoeksgroepen over de gehele aarde onderzoek naar doen. Ook bedrijven doen mee in deze race – met name Google, IBM en Microsoft. Er zijn allerlei redenen waarom quantumcomputers zo gewenst zijn. Quantumcomputers maken gebruik van de fundamentele natuurwetten die we kennen, om berekeningen uit te voeren die onze huidige computers met geen mogelijkheid aan zouden kunnen.

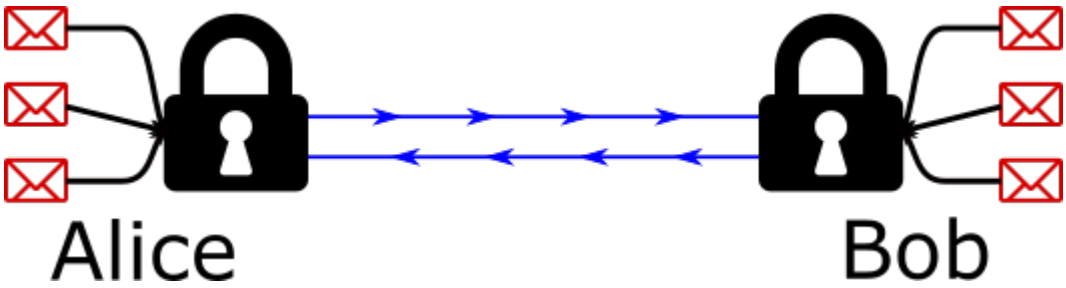
Dit brengt ons naar het onderwerp van dit artikel. Een van de bekendste berekeningen die we met quantumcomputers willen doen is namelijk verbonden met een concept wat we dagelijks in ons leven gebruiken, zonder dat we daarbij stil staan: encryptie. Vrijwel alles wat je doet wat te maken heeft met communicatie, geld, internet en andere vormen van elektronica, maakt gebruik van versleuteling. In al die situaties zijn er twee partijen die niet willen dat hun onderlinge berichten afgeluisterd kunnen worden. Het verzenden van versleutelde berichten is een eeuwenoud concept en heeft altijd een belangrijke rol gespeeld voor, bijvoorbeeld, diplomatieke correspondentie en militaire doeleindes.

Afbeelding 1. Twee vormen van encryptie. Links: Een (wel heel) klassiek voorbeeld van

encryptie: een lint met letters. Alleen als iemand de juist diameter van de stok weet, kan die persoon het lint op een manier oprollen waardoor de tekst leesbaar wordt. Rechts een ander bekend voorbeeld: de Enigmamachine. Deze machine, oorspronkelijk ontworpen in Duitsland vlak na de Eerste Wereldoorlog, werd door de nazi's veelvuldig gebruikt in de Tweede Wereldoorlog om te kunnen communiceren via radiogolven, ondanks dat die vrijwel zeker opgevangen zouden worden door de geallieerden.

Hoe werkt encryptie?

Om het concept van encryptie iets concreter te maken, geef ik een schematisch voorbeeld. We hebben twee personen (in vaktermen: 'Alice' en 'Bob') die graag een bericht willen doorgeven. We nemen aan dat Alice een bericht wil versturen naar Bob, en dat dit bericht uit een reeks nullen en enen bestaat (ook wel een 'bitstring' genoemd). De vraag is nu: hoe kunnen Alice en Bob dit zo doen dat niemand mee kan luisteren, en waar zitten de zwakke punten? Hier is het stappenplan: Alice neemt haar bericht, en versleutelt dit volgens een bepaald encryptieprotocol. Vervolgens stuurt ze het versleutelde bericht over een publiek kanaal, wat iedereen naar believen kan afluisteren, zodat het uiteindelijk bij Bob terechtkomt. Bob ontsleutelt dan het bericht, en ziet precies wat Alice wilde doorgeven.

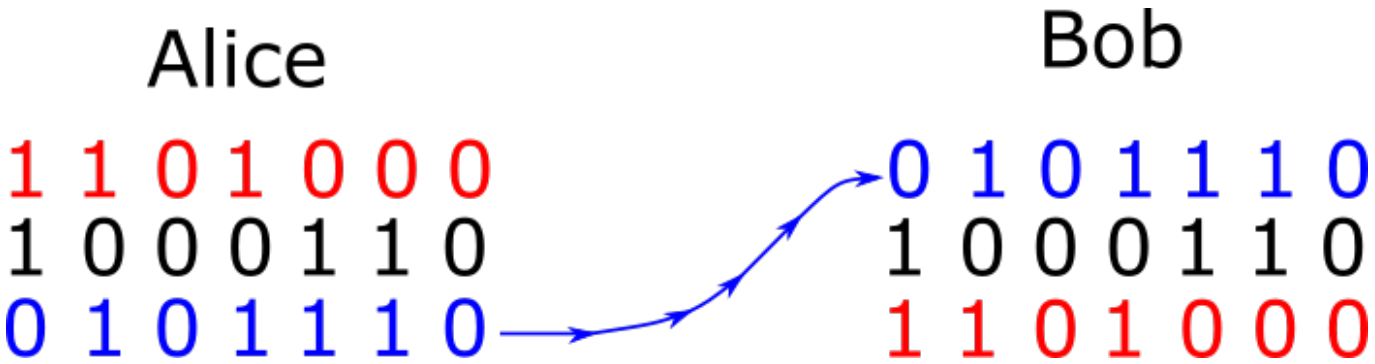


Afbeelding 2. Een schematisch voorbeeld van het verzenden van een bericht. Alice heeft een serie berichten. Die versleutelt ze, zodat de informatie verzonden kan worden over een publiek toegankelijk kanaal (bijvoorbeeld radiogolven, of internet). Bob ontvangt de versleutelde berichten, en ontsleutelt ze.

Het versleutelen waarover ik het hierboven heb is geen probleem, onder één belangrijke voorwaarde: er moet een 'veilige sleutel' zijn. Dit betekent dat, zodra zowel Alice als Bob het (zelfde) 'wachtwoord' kennen, bijvoorbeeld een random reeks nullen en enen, er een algoritme is dat het bericht van Alice goed kan versleutelen op zo'n manier dat iemand zónder het wachtwoord de oorspronkelijke boodschap nooit kan achterhalen. Alice stuurt het

bericht dan naar Bob, die het vervolgens met het wachtwoord en een ander algoritme kan ontsleutelen.

Het meest voorkomende voorbeeld hiervan heet het ‘one time pad’. Dit is een encryptieprotocol waar je een random bitstring van dezelfde lengte als het bericht gebruikt, en vervolgens de bits paarsgewijs optelt (waarbij 1+1=0) om een nieuwe string te vormen. Het ontsleutelen kan vervolgens gedaan worden door precies dezelfde random bitstring weer erbij op te tellen! (Zie afbeelding 3.’) Van het ‘one time pad’-protocol kun je vrij gemakkelijk bewijzen dat het *absoluut* veilig is. Er is namelijk geen enkele manier om het zónder de sleutel te kraken door gebruik te maken van wiskundige functies. Alleen als iemand de sleutel heeft, kan een bericht gekraakt worden. Dit is laat ook direct zien waar het zwakke punt in dit protocol zit: hoe zorg je ervoor dat zowel Alice als Bob – maar niemand anders – over dezelfde ‘random bitstring’ beschikt? Vanaf nu is de vraag dus: Hoe kan Alice, communicerend over een open kanaal, een sleutel (bitstring) creëren en delen met alléén Bob?



Afbeelding 3. Een voorbeeld van een ‘one time pad’. De boodschap (in het rood) wordt met een bitstring (zwart), die zowel bij Alice als bij Bob bekend is, versleuteld. Dit gebeurt door elke bit van de boodschap paarsgewijs bij een bit van de sleutel ‘op te tellen’. Daarbij is 1+0 = 1, 0+1 = 1 en 0+0 =0, zoals we gewend zijn, maar 1+1=0. Het versleutelde bericht (blauw) wordt naar Bob verzonden. Bob doet nu dezelfde operatie als Alice, en telt paarsgewijs de bits van het blauwe bericht en de zwarte sleutel op. Het resultaat is weer de rode bitstring, het oorspronkelijke bericht.

Sleutels delen: de ‘klassieke’ manier

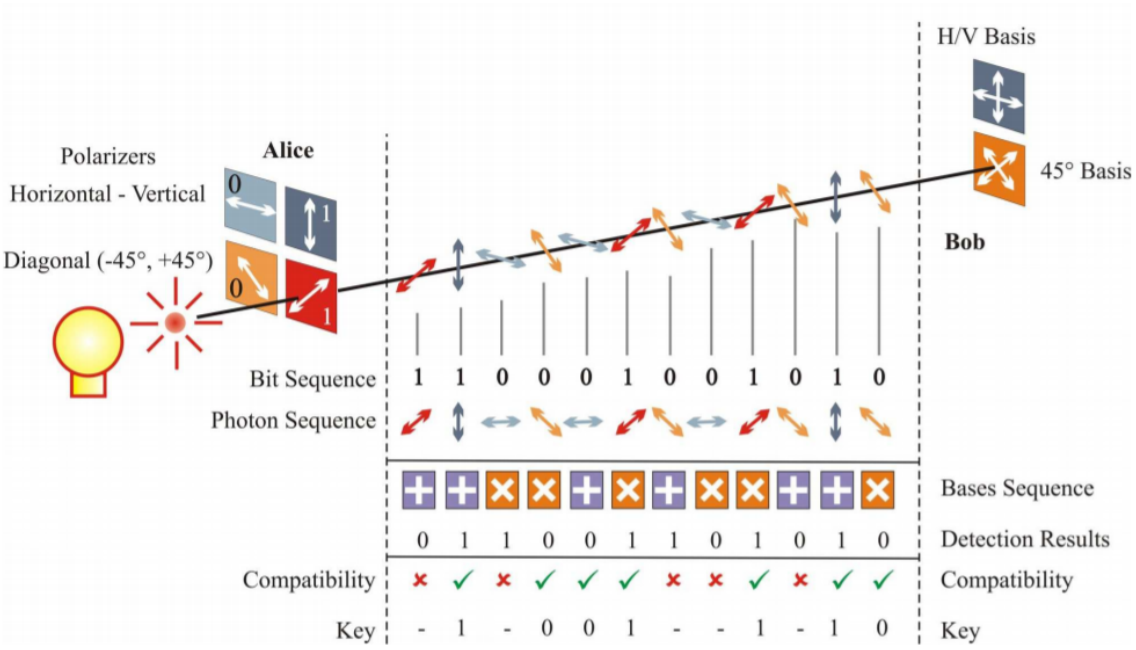
Er bestaan protocollen die momenteel gebruikt worden om een beveiligingssleutel tussen Alice en Bob te realiseren. (We gaan ervan uit dat Alice en Bob niet elke keer dat ze een

nieuwe sleutel nodig hebben bij elkaar kunnen komen en zo iets kunnen afspreken; ook het vaststellen van de sleutel zelf moet dus via een open kanaal gebeuren.) Een voorbeeld is de end-to-end encryptie van Whatsapp die het afgelopen jaar enkele keren in het nieuws is gekomen. Zulke algoritmes hebben echter allemaal dezelfde zwakte: ze zijn niet wiskundig bewezen veilig. Feitelijk baseren ze zich allemaal op hetzelfde concept: het bestaan van wiskundige functies die makkelijk uit te rekenen zijn als je er iets in stopt, maar waarvan het heel moeilijk van is om te bepalen wat de input oorspronkelijk was als je alleen de uitkomst kent. De veiligheid van methodes die zulke functies gebruiken is echter niet bewezen! In de praktijk zijn ze tot nu toe wel veilig gebleken, omdat niemand een efficiënt algoritme heeft kunnen bedenken waarmee dit soort functies eenvoudig te 'ontrafelen' is – waarmee je dus de input uit de output kunt achterhalen. Mochten quantumcomputers echter realiteit worden, dan bestaat er al wél een algoritme dat de input snel kan berekenen: *Shor's algoritme*. Daarmee zou het delen van een sleutel op de huidige klassieke manieren gekraakt worden, en is het niet meer mogelijk om berichten veilig te versleutelen. We moeten dus een andere manier bedenken om een sleutel vast te stellen tussen twee personen op een manier die honderd procent veilig is, ook voor aanvallers die beschikken over fabelachtige quantumcomputers.

Natuurkunde biedt uitkomst

Gelukkig is er een andere manier van sleuteldelen bedacht die wél precies aan onze strenge eisen voldoet. Het idee is om informatie te coderen in de quantummechanische golffuncties van deeltjes. Zulke golffuncties, die de kans beschrijven dat een quantumdeeltje in een bepaalde toestand is, beschikken namelijk over een uniek karakter: het is niet mogelijk om een golffunctie in zijn geheel te 'achterhalen'. We kunnen metingen aan een deelte doen, maar het is nooit mogelijk om alle eigenschappen van een golffunctie – alle kansen – tegelijkertijd te weten. Dit fenomeen zie je bijvoorbeeld in de welbekende onzekerheidsrelatie van Heisenberg: die zegt dat je van een deeltje nooit tegelijk exact de plaats én de snelheid kunt weten. Een van de eerste realisaties van quantum-sleutelverdelen gebruikt de onbepaalbaarheid van een golffunctie in een heel eenvoudige, maar toch degelijke methode: BB84, wat staat voor "Bennett" en "Brassard", die het proces hebben voorgesteld en uitgevoerd in 1984. Zij bedachten een protocol om een bitreeks (de sleutel) met behulp van de polarisaties van lichtdeeltjes (fotonen) te versturen. Deze polarisaties kun je zien als de richting waarin het elektrische veld van het foton oscilleert. Gaat het veld omhoog en

omlaag, dan spreken we van ‘verticale’ polarisatie. Oscilleert het van links naar rechts, dan noemen we dat ‘horizontale’ polarisatie. Natuurlijk kunnen ook andere trillingen voorkomen, zoals bijvoorbeeld schuine polarisaties.



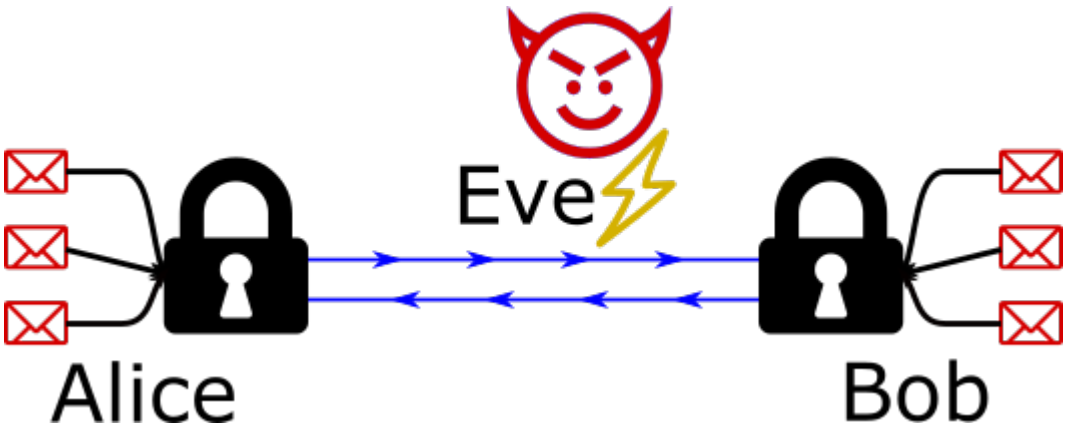
Afbeelding 4. Een schematische weergave van het BB84 quantum-sleutelverdeelprotocol. Alice creëert fotonen met een lamp. Vervolgens kiest ze ervoor om deze fotonen één van vier mogelijke polarisaties te geven, overeenstemmend met haar gekozen random bitstring (de sleutel). De fotonen reizen vervolgens naar Bob, die ervoor kiest om ze te meten volgens één van twee opties. Soms gokt hij goed – dat wil zeggen: meet hij precies de richtingen waarvan Alice er één gekozen heeft – en soms niet, zoals aangegeven met de vinkjes of kruisjes. Door vervolgens te communiceren welke keuzes goed waren (diagonale of horizontale/verticale) weten Alice en Bob welke bits ze moeten bewaren (alleen die bits met een vinkje). Deze bits vormen vanaf nu de sleutel. Afbeelding: Daniel D. Moskovich.

In afbeelding 4 zien we het protocol zoals het voorgesteld is: Alice kan bits vertalen in polarisaties door te stellen dat verticale polarisatie en één van de twee schuine polarisaties gelijk zijn aan 1, en horizontale en de andere schuine polarisatie gelijk aan 0. Dit communiceert Alice aan Bob over een publiek kanaal: iedereen mag weten welke polarisaties overeenkomen met een 1, en welke met een 0. Vervolgens gebruikt Alice deze methode om me behulp van polarisatiefilters een random bitstring van haar eigen keuze (de sleutel,

bijvoorbeeld 110001001010) over het publieke kanaal te versturen. Voor elke bit maakt ze daarbij een willekeurige keuze tussen de twee polarisaties die de 1 ofwel de 0 weergeven. Bob ontvangt de fotonen, en kiest ervoor om ze te meten op één van 2 mogelijke manieren: hij gaat er ofwel vanuit dat het foton verticaal/horizontaal is gepolariseerd, ofwel dat het schuin is gepolariseerd. Bob weet echter niet welke keuzes Alice daarin gemaakt heeft; hij doet dus voor elk bit een gok. Gokt Bob de juiste polarisatierichting, dan meet hij de polarisatie van het foton op een juiste manier, en associeert vervolgens de juiste bitwaarde aan dit foton. (Uiteraard weet Bob hier nog niet dat hij de juiste polarisatie had gekozen!). Kiest Bob de verkeerde polarisatie, dan krijgt hij – omdat hij schuin op de echte polarisatierichting meet – met 50 procent kans de 'juiste' waarde eruit, en met 50 procent kans de onjuiste waarde. (Ook hier weet hij dus niet of zijn meetkeuze juist is of niet!) Bob heeft nu een eigen bitstring gemeten, die voor 50 procent zeker juist is, en waarvan de andere 50 procent bestaat uit willekeurige 'ruis'. Uiteindelijk is het natuurlijk de bedoeling dat Alice en Bob dezelfde bitstring gebruiken om berichten mee te versleutelen. Op de een of andere manier moet Bob er dus achter komen welke metingen juist waren, en welke niet. Dit is simpel! Bob zegt over het publieke kanaal tegen Alice precies welke polarasitie-keuzes hij maakte. Alice antwoordt vervolgens naar Bob in welke situaties hij de juiste gok heeft gemaakt. Bob en Alice gooien nu alle bits die overeenkomen met die verkeerde keuzes weg, en houden alleen de juiste bits over. Deze bits vormen de sleutel waarmee de berichten veilig kunnen worden versleuteld.

Een aanvaller in het spel

Het is op dit punt helemaal nog niet zo duidelijk waarom het bovenstaande protocol zo 'veilig' is; het lijkt er immers op dat alles hierboven vrij gemakkelijk afgeluisterd kan worden! Tot nu toe hebben we gedaan alsof er geen aanvaller was, maar natuurlijk is het hele punt van dit protocol dat een afluisteraar met geen mogelijkheid de encryptiesleutel van Alice en Bob kan afluisteren. In de literatuur wordt deze afluisteraar ook wel vaak 'Eve' genoemd (vanwege het Engelse woord 'to eavesdrop' – afluisteren).



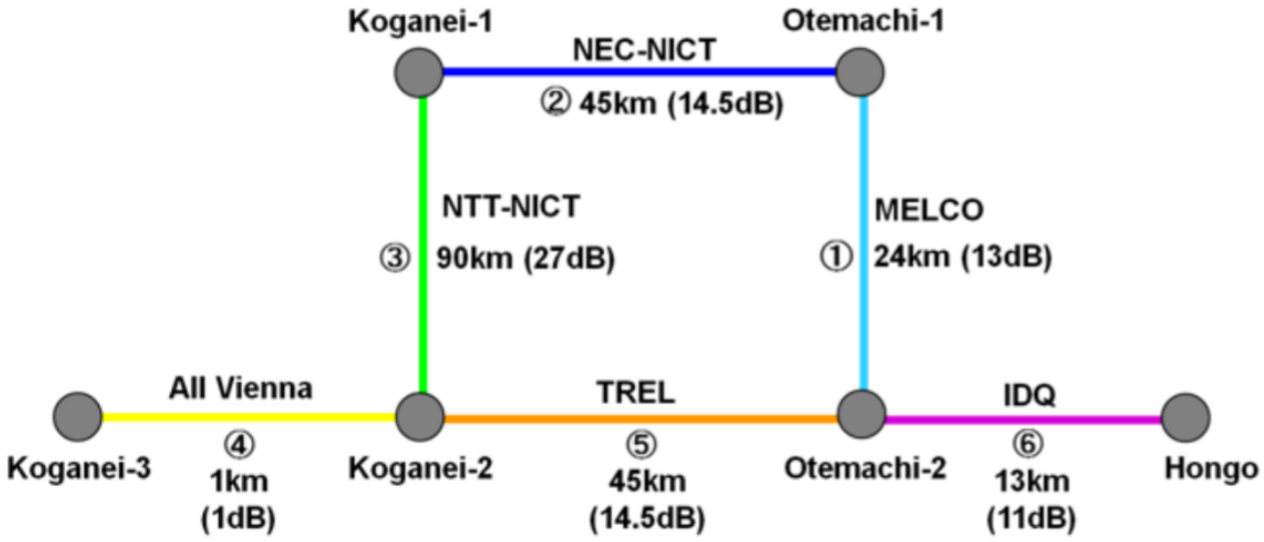
Afbeelding 5. Eve luistert af. Deze afbeelding toont hetzelfde als afbeelding 2, maar nu met de kwaadaardige Eve die in staat is om alles wat er door de blauwe lijnen heen gaat af te luisteren. Een echt veilige communicatie is dusdanig versleuteld dat, ongeacht hoe slim Eve is en hoeveel rekenkracht ze heeft, ze nooit in staat zal zijn om de communicatie tussen Alice en Bob te ontsleutelen en mee te lezen.

In informatie-theoretische gedachte-experimenten doen we dan vaak alsof Eve ‘almachtig is’. Zij mag en kan alles doen wat maar door de natuurwetten is toegestaan, kan alle informatie die verzonden wordt over het publieke kanaal vrij waarnemen, heeft de beschikking over een quantumcomputer, enzovoort. Ondanks dat Eve zo’n sterke aanvaller is, die kan afluisteren wat ze wil, is de claim dat het BB84-protocol toch veilig is tegen een dergelijke afluisteraar. Dit komt uiteindelijk door wat hierboven al gezegd is: het is niet mogelijk om een quantummechanische golffunctie volledig te meten.

Laten we eerst bedenken wat Eve zou kunnen doen om de sleutel te kraken. Zo kan Eve bijvoorbeeld de fotonen die onderweg zijn naar Bob onderscheppen en meten op dezelfde manier als Bob dat doet. Dat zal Eve echter niet helpen. Immers: elke bit die Bob niet heeft binnengekregen, wordt simpelweg niet meegenomen in de encryptiesleutel. Fotonen onderscheppen heeft dus geen zin.

Dan kan Eve misschien het volgende doen: ze meet eerst een foton zoals Bob het zou doen. Ze meet dan bijvoorbeeld een verticale polarisatie, en stuurt om niet betrapt te worden een foton met een verticale polarisatie naar Bob. (Dat kan een ‘nieuw’ foton zijn, of zelfs hetzelfde foton – na de meting van Eve is ook dat volledig verticaal gepolariseerd.) Deze methode lijkt al een stuk beter, *maar*: omdat Eve niet weet in welke richtingen ze moet meten, is er een kans van 50 procent dat een andere keuze heeft gemaakt dan Alice en dus

fout heeft gemeten! Vervolgens heeft Bob natuurlijk ook weer een kans dat hij in juiste richtingen (ten opzichte van Alice) meet, maar wél in andere richtingen dan Eve. Omdat hij niet in de richtingen meet van het ‘nepfoton’ van Alice is er dus in totaal een kans van 25 procent dat Bob, ondanks dat hij een bit meet in de juiste polarisatie zoals Alice had bedacht, toch de verkeerde uitkomst vindt omdat Eve ertussen zat! Dit lijkt vervelend voor Alice en Bob, maar het blijkt juist een voordeel te zijn. Gelukkig is Eve namelijk niet als enige een slimmerik en kunnen Alice en Bob de extra ruis die Eve veroorzaakt juist slim gebruiken. Wat ze doen is heel simpel: ze nemen een klein stukje van hun random bitstring (bijvoorbeeld de eerste 100), en sturen die naar elkaar toe. Als alles goed is gegaan, zou er geen verschil tussen die bitstrings moeten zitten; de ‘verkeerde’ bits hebben Alice en Bob immers al weggegooid. Als Eve er echter tussen heeft gezeten, dan zien ze dat grofweg 25% van de bits die juist zouden moeten zijn tóch niet overeenkomen. Alice en Bob merken dit, en concluderen dat er iemand heeft meegeluisterd. Ze gooien vervolgens hun gehele sleutel weg, en beginnen opnieuw, bijvoorbeeld via een ander kanaal, net zolang tot er een sleutel wel ongeschonden overkomt. Zodoende maakt het niet uit wat Eve doet; Alice en Bob weten altijd zeker of hun encryptie-sleutel veilig is overgekomen, of niet! De ‘onzekerheid’ van het doen van metingen is dus de grondslag voor het veilig genereren van gedeelde encryptiesleutels. En zoals gezegd: een veilige sleutel betekent een veilig bericht. Met deze sleutel kunnen Alice en Bob in alle rust over onze welbekende internetkabels emailen, streamen of gamen!



Afbeelding 6. Fysieke realisatie van een netwerk van quantum-sleutelverdelers. Dit netwerk is in 2011 in Tokyo gebouwd door M. Sasaki et al. De grijze cirkels komen overeen

met fysieke plekken in Tokyo en omstreken. Elke lijn heeft een andere kleur, die de verschillende protocollen aangeeft die gebruikt worden om een sleutel te genereren. Zoals te zien is zijn de afstanden tussen verschillende knooppunten tussen 1 en 90 kilometer. Met dit netwerk is het de onderzoekers gelukt om een volledig veilig netwerk te creëren dat zelfs niet gekraakt kan worden door een quantumcomputer. Afbeelding: M. Sasaki et al.

Bestaat dit al?

Een terechte vraag die je kunt stellen, is of deze op theorie gebaseerde technologie ook al gecommercialiseerd kan worden en nuttig is. Het antwoord is: ja! Momenteel zijn er al commerciële opties beschikbaar voor bedrijven om te kunnen communiceren met behulp van quantumencryptie. Er zijn nog wel enkele praktische beperkingen. Zo zijn er afstandslimieten waardoor een quantummechanisch signaal niet over willekeurig grote afstanden kan worden verzonden. Door de lucht is de limiet van de orde van grootte van 150 kilometer. Daarvoor heb je dan echter wel grote telescopen en zendmasten nodig. Een betere optie is het gebruik van optische vezels. Met behulp van deze kabels kunnen quantummechanische sleuteldistributies gemakkelijk 200 kilometer ver komen voordat het signaal volledig is gedegradeerd. Bovendien is het mogelijk om verschillende stukken van een netwerk aan ‘elkaar te rijgen’ waardoor je in principe over een heel land of zelfs continent sleutels kunt realiseren tussen twee partijen. Zo over de oceanen heen communiceren is nog wel lastig, maar ook hier zijn ideeën voor, bijvoorbeeld met behulp van satellieten. Deze systemen zijn alleen nog in de ontwerpfase maar naar mijn weten nog geen realiteit. Desondanks is het zeker geen gekke aanname dat in de toekomst alle beveiligde communicatie volledig zal worden gebaseerd op de quantummechanica!