ISOTRON

OUR HEART BEATS AUTOMATION

Security Certificate

wat is het en waarom is het veilig?

Security certificate: Wat is het en waarom is het veilig?

U bent de term ongetwijfeld een keer tegengekomen als u hebt gezocht naar een veilige manier om uw machines te verbinden met het internet: de security certificate. Wat is het precies en wat doet het? Zorgt het voor een veiligere verbinding en waar moet u dan op letten als u deze claims vergelijkt? We vertellen u hier graag meer over.

Wat is een security certificate?

Laten we eerst vaststellen waar we het nu over hebben wanneer we praten over een security certificate. In deze context hebben we het namelijk niet over een certificaat uitgegeven door een onafhankelijk instituut dat heeft gecontroleerd of een bepaalde oplossing veilig is. Dergelijke certificaten bestaan wel degelijk. Een goed voorbeeld hiervan is de certificering van Secomea. Secomea laat een derde, onafhankelijke partij hun oplossingen checken om te kijken of deze aan alle cybersecurity voorschriften voldoet. Dit certificaat is **hier** te vinden.

In dit artikel hebben we het echter over een security certificate vanuit een technisch perspectief. Een security certificate is een klein bestandje dat systemen met elkaar uitwisselen. In deze bestandjes is een encryptiesleutel terug te vinden waardoor de systemen op een veilige manier data kunnen uitwisselen.

Een klein stukje geschiedenis

Alles wat zomaar over het internet wordt verstuurd kan per definitie worden beschouwd als niet veilig. De data die wordt verstuurd, reist als het ware over een onbekend aantal systemen om van punt A naar punt B te komen. Alle systemen tussen punt A en B geven het datapakketje continue aan elkaar door totdat de bestemming bereikt is. Dit betekent ook dat al deze systemen kunnen kijken wat er in het pakketje zit, net als met een brief die door een postbedrijf wordt bezorgd. In de oudheid was dit ook al een bekend probleem, daarom dat de vroegere keizers, legers, etc. encryptie gebruikte om hun boodschap te versleutelen en onleesbaar te maken. Een bekend voorbeeld hiervan is het verhaal van de Enigma in de tweede wereldoorlog. De Enigma code werd uiteindelijk gekraakt door het team van de briljante wiskundige genaamd Alan Turing. Hiervoor gebruikte hij een vroege voorloper van de computers die wij nu allemaal dagelijks gebruiken.

Synchrone versleuteling

Om data veilig te kunnen versturen zal deze moeten worden versleuteld. Dit versleutelen kan op verschillende manieren. Een manier om dit te doen is synchrone versleuteling. Bij het versleutelen van een bericht wordt gebruik gemaakt van een vooraf afgesproken methode en een sleutel. Een voorbeeld van een simpele methode om een tekst te versleutelen is door alle letters een aantal plaatsen in het alfabet te verschuiven. Een zogenaamd Caesarcijfer. Nemen we bijvoorbeeld als sleutel 2, dan wordt een A een C, een B een D, etc. Nemen we bijvoorbeeld sleutel 3, dan wordt een A een D en een B een E.

Zowel de ontvanger als de zender van het bericht gebruiken bij de data-uitwisseling dezelfde sleutel. In het voorbeeld van een Caesarcijfer gebruikt de zender de sleutel 2 om door te tellen in het alfabet. De ontvanger gebruikt weer sleutel 2 om terug te tellen in het alfabet. Omdat zowel zender als ontvanger dezelfde sleutel gebruiken heet dit synchrone encryptie.

Asynchrone versleuteling

Heel veel van dit soort versleutelingsmethodes werkte erg goed in de oudheid. Je komt namelijk samen, spreekt een methode en een sleutel af en je kan berichten naar elkaar sturen. In het geval van het internet is er echter een probleem. Elke keer dat twee systemen contact met elkaar leggen moet eerst de

methode en de sleutel worden afgesproken. De communicatie over deze afspraken gaat echter over dezelfde onveilige verbinding, waardoor kwaadwillende mee zouden kunnen kijken en de communicatie in de toekomst ook weer kunnen ontcijferen. Gelukkig zijn er een aantal wiskundige geweest die hier een oplossing voor hebben bedacht: Asynchrone encryptie.

Bij asynchrone versleuteling gebruiken zender en ontvanger niet dezelfde sleutel, in tegenstelling tot synchrone versleuteling. Verschillende wiskundige hebben complexe formules weten te maken waarbij een sleutelpaar nodig is van twee verschillende sleutels die aan elkaar gekoppeld zijn: sleutel A en sleutel B. Een bericht versleuteld met de ene sleutel kan enkel en alleen ontcijferd worden met de andere sleutel. Wanneer de zender nu een sleutelpaar creëert kan hij bijvoorbeeld sleutel A geheim houden voor zichzelf. Dit noemen we de zogenaamde private key. De andere sleutel, sleutel B, kan hij met iedereen delen, de zogenaamde public key. Een boodschap kan worden **ver**sleuteld met een public key, maar alleen worden **ont**sleuteld door de private key. Mocht iemand dus de public key in handen hebben waarmee een bericht is versleuteld, heeft hij nog steeds de private key nodig om het bericht te ontsleutelen.

Andersom is het uiteraard ook mogelijk. De zender kan een bericht versleutelen met zijn private key. Dit bericht kan enkel ontcijferd worden met de bijbehorende public key. Hoewel iedereen nu kan meelezen, omdat iedereen de public key kan opvragen, heeft de ontvanger bij het lezen van het bericht wel de zekerheid dat niemand anders onderweg het bericht kan hebben aangepast. Op deze manier kan deze encryptie dus ook een garantie geven over de authenticiteit van het bericht.

Security certificates in de praktijk

Bij remote access systemen en cloud oplossingen wordt veelvuldig gebruik gemaakt van asynchrone versleuteling. Alleen wordt er hierbij niet één sleutel publiekelijk beschikbaar gesteld. Voor elk device wordt door de server een apart sleutelpaar gecreëerd waarvan er eentje enkel wordt gedeeld met het device en eentje wordt door de server zelf gehouden. Voor het delen van de sleutel en de bijbehorende encryptiemethode wordt een security certificate gebruikt. Simpelweg is het dus een bestandje waar zowel de sleutel als de encryptiemethode in staat vermeld.

Uiteraard zijn bovenstaande voorbeelden een simplificatie van de werkelijkheid. Daarnaast zijn er veel meer factoren die bepalen hoe veilig een systeem is. Is het dan van belang om te weten of een bepaalde oplossing gebruik maakt van security certificates? Op zichzelf niet, maar het is zeker een belangrijke bouwsteen die nodig is om een zo veilig mogelijke oplossing te creëren.

Waar moet u op letten?

Hoe kunt u dan bepalen of een oplossing veilig is? Voor wie niet dagelijks bezig is met cyber security en encryptie is het zeer lastig om op technisch vlak een solide oordeel te vellen over de veiligheid van een systeem. Secomea heeft hier al over nagedacht door al hun systemen door een onafhankelijke partij te laten controleren. De UniCloud oplossing van Unitronics lost dit op door niets zelf te ontwikkelen maar gebruik te maken van de veilige verbindingen van Amazon’s AWS. Dit is een dienst die zo groot is met enorme veiligheidsresources dat hackers hier niet tegenop kunnen boksen.

Uiteindelijk zal de manier waarop een bedrijf omgaat met cyber security u veel meer vertellen dan een paar mooie online slogans. Mocht u vragen hebben over cyber security, neem dan contact met ons op. Bij Isotron Systems helpen we u graag verder.