Wat stuurt die app nou eigenlijk?

Een workshop over het afluisteren van een Android app

Hoi! Ik ben Jurrie

- Ik ben een staff software engineer bij Topicus.Finance
- Ik vind het leuk om „te prutsen“ met software
- Ik ben geen security specialist
  - Uitleg over encryptie is hoog over!
  - Neem dingen die ik zeg niet direct voor waar aan, maar…
  - …probeer ze zelf uit
- Vragen? Stel ze direct – wacht niet tot het einde

Topicus?

1.100
Collegas

730
Techies

60
Studenten in 2021

30 / 70
Vrouw / Man

12
Locaties in NL

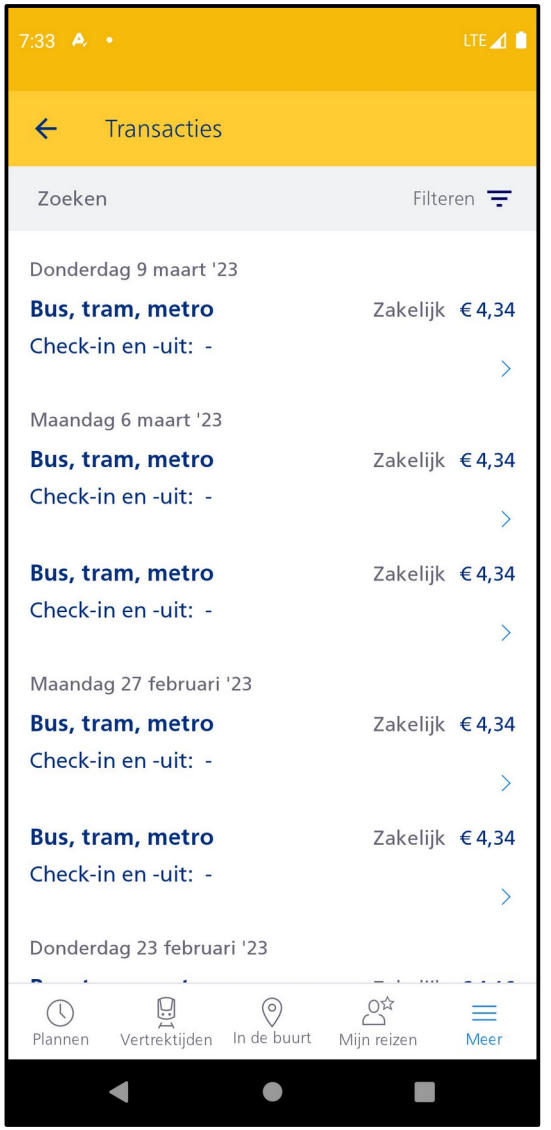
8,5 jaar
Gemiddelde arbeidsduur

En wat doen jullie nou precies..?

Gemiddeld zit iedere Nederlander in 4 van onze systemen

...oh, én ik ben lui!

- Ik moet de NS app gebruiken om mijn woon-werk verkeer te declareren
- Ik maak altijd in dezelfde rit
- keer klikken per rit! ****DRIE** KEER!!**
- Dat moeten we automatiseren!



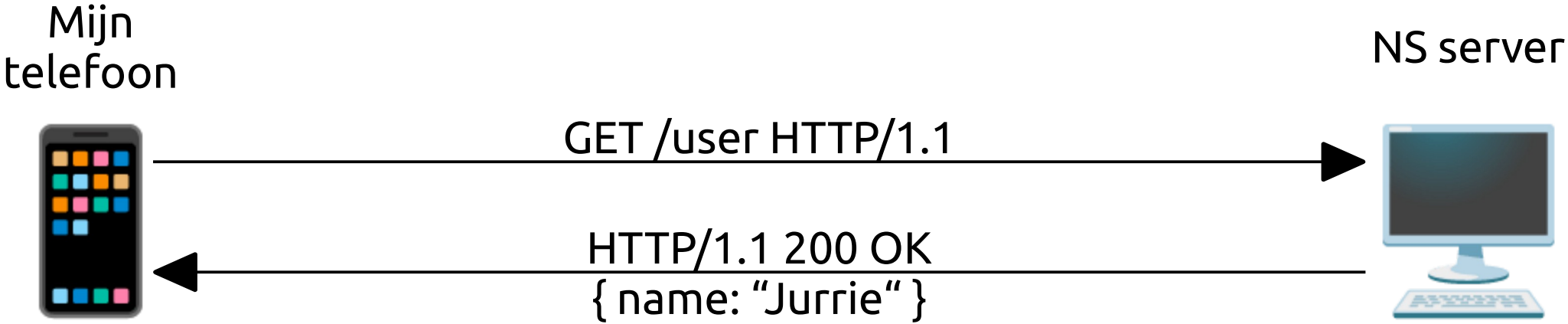
Let’s automate this!

- Ik wil een tooltje maken dat de NS app nadoet
- Die vraagt een lijst van gemaakte ritten op
- De ritten huis werk (andersom) markeren als woon-werk
- Wat moet dat tooltje sturen naar de NS server om dit te doen?
  - Let’s find out!

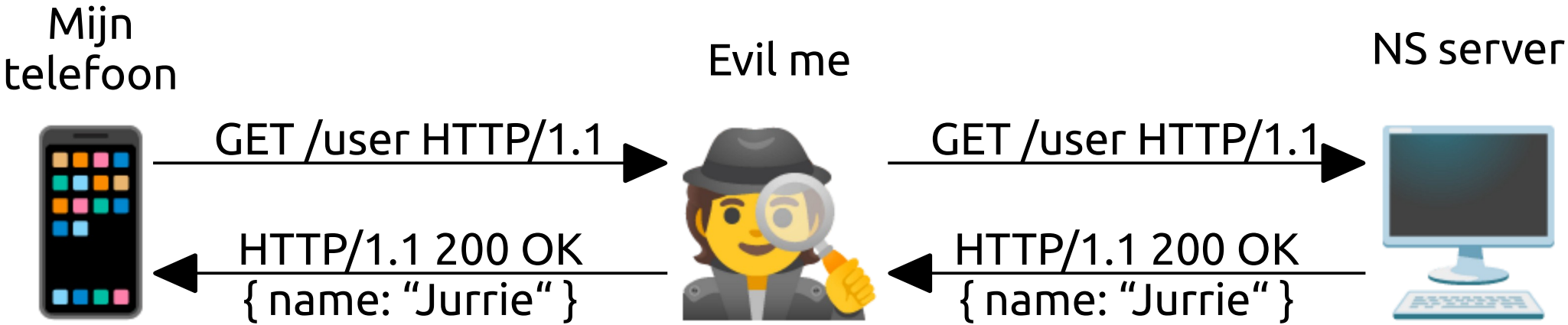
Hoe praat een app met een server?

- De meeste apps praten via een REST API
  - De data is meestal in JSON formaat (maar alles is mogelijk – REST is geen stricte standaard)
- In enterprise software en M2M vind je ook SOAP
  - SOAP is gestandaardiseerd, je zult XML berichten zien
- Deze communicatie is (hopelijk) niet te onderscheppen

En dan nu een plaatje!



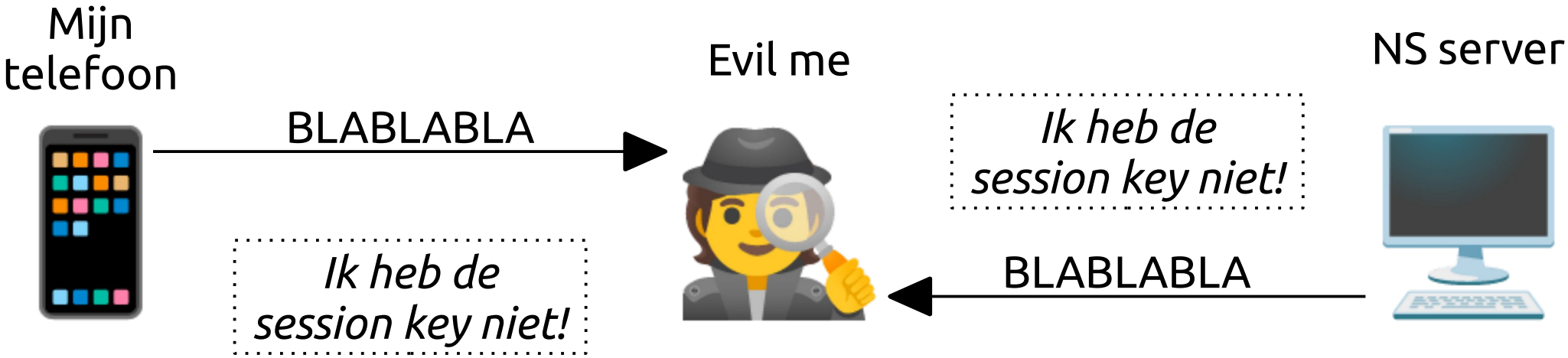
En dit heb ik nodig…



Waarom kunnen we dat niet onderscheppen?

- De verstuurde data is versleuteld
- Dit kom je vaak tegen als *HTTPS*
  - HTTP over *TLS*
- Data is versleuteld met een session key (symmetrische sleutel)
  - Ook wel shared secret genoemd
  - Encryptie (versleutelen) en decryptie (ontsleutelen)

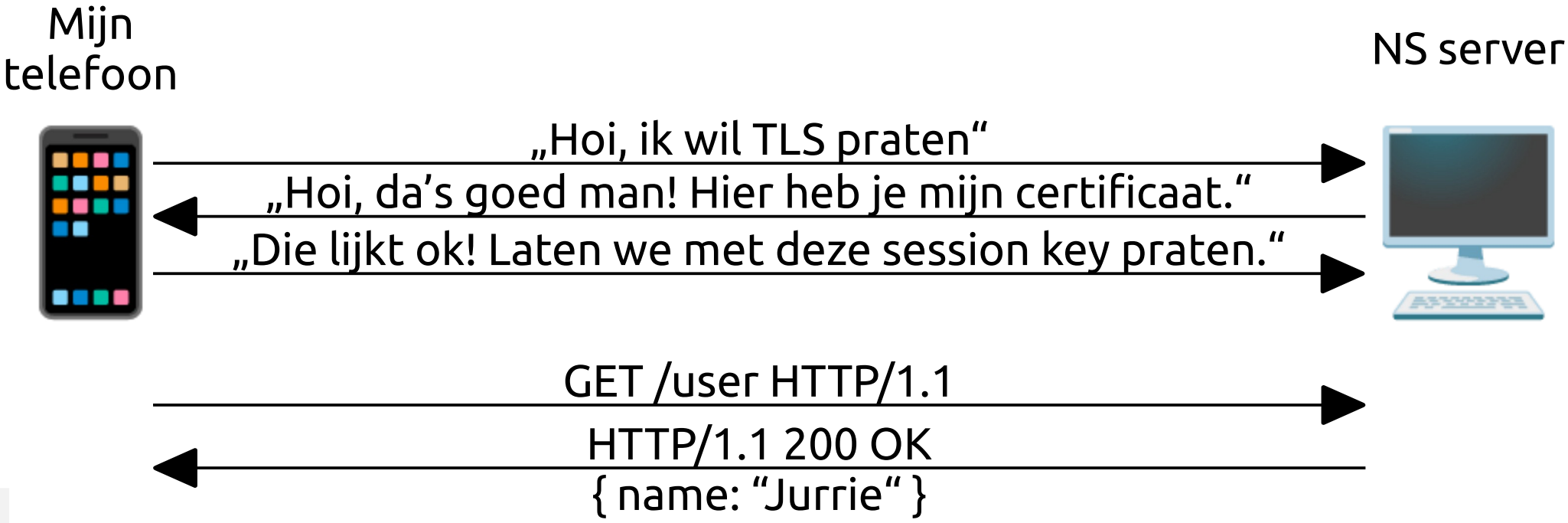
Dus terug naar mijn NS-app probleem…



Waarom hebben we de session key niet?

- De session key wordt uitgewisseld in de TLS handshake
  - Een handshake is afstemming tijdens het starten van een verbinding
- Er worden hier twee dingen gebruikt
  - private/public keys (oftewel een key pair, of sleutelpaar)
  - certificaten

Dus eigenlijk ziet het er meer zo uit…



Wat is een key pair?

- Een key pair is twee sets van speciale wiskundige nummers
  - Eén set van nummers maakt samen de „private key“
  - Eén set van bijbehorende nummers maakt samen de „public key“
- Met de private key kun je de public key berekenen
  - Andersom kan niet
- De private key deel je met niemand
- De public key deel je met alles en iedereen die je maar wilt
- Key pairs ~~kunnen~~ moeten door jezelf aangemaakt worden
  - Waarom?

Hoe gebruik je een key pair?

- Data versleuteld met een public key kan alleen worden ontsleuteld met de bijbehorende private key
- Data ondertekend met een private key kan alleen worden geverifieerd met de bijbehorende public key
  - Zoals een handtekening: „Ik bevestig dat ik denk dat deze data OK is!“
  - Waarom is ondertekenen geen versleutelen?
- Je kunt beide vormen tegelijkertijd toepassen
  - Eerst ondertekenen, dan versleutelen
  - Eerst versleutelen, dan ondertekenen

Iedereen heeft het altijd over Alice en Bob…

- Alle voorbeelden die je over encryptie tegenkomt gebruiken de namen Alice en Bob
- Maar we leven in 2023, dus...
- …Alice and Bob – de woke editie!

Iedereen heeft het altijd over Alice en Bob…

Bob

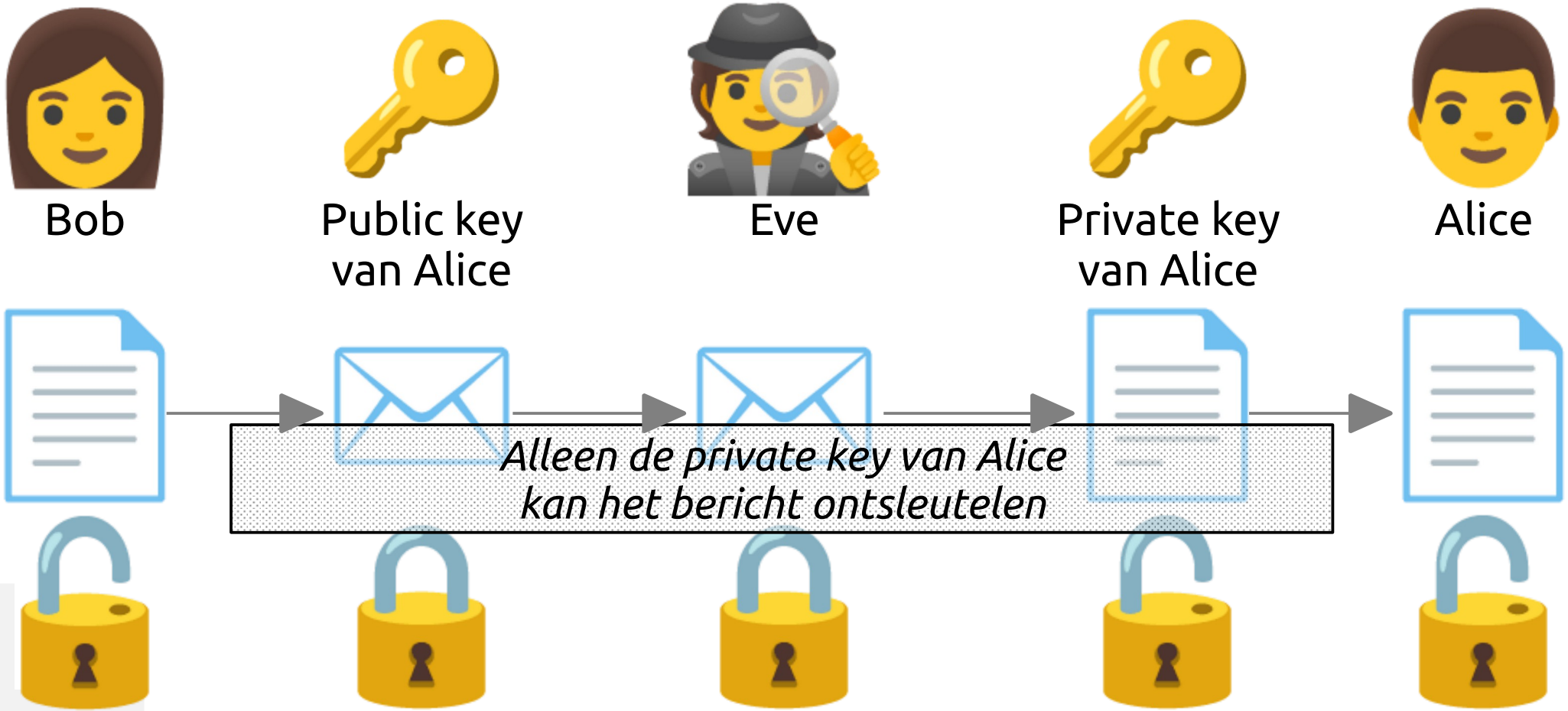
Public key van Alice

Eve

Private key van Alice

Alice

Iedereen heeft het altijd over Alice en Bob…



Maar ik kon toch mijn eigen sleutels maken?

- Ik heb de private key van de NS server niet
- Maar sleutels kon je zelf genereren!
- Waarom dan niet…
  - …de juiste sleutels gebruiken als ik praat met de NS server?
  - …mijn eigen sleutels gebruiken als ik praat met de app?

Warning: Potential Security Risk Ahead

Firefox detected a potential security threat and did not continue to **www.ns.nl**. If you visit this site, attackers could try to steal information like your passwords, emails, or credit card details.

What can you do about it?

The issue is most likely with the website, and there is nothing you can do to resolve it. You can notify the website's administrator about the problem.

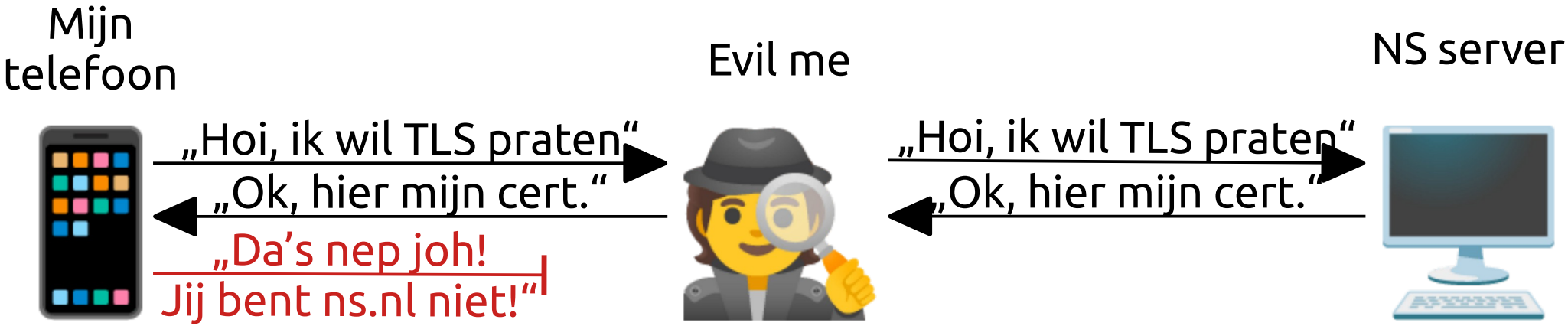
Learn more…

Go Back (Recommended) Advanced…

Websites prove their identity via certificates. Firefox does not trust this site because it uses a certificate that is not valid for www.ns.local. The certificate is only valid for the following names: www.ns.nl, www.spoordeelwinkel.nl, www.ov-fiets.nl, www.nsinternational.nl, www.nsfiets.nl, www.eropuit.nl, wsapis001.ns.nl, webservices.ns.nl, nsinternational.nl, ns.nl, mobiel.ns.nl, media.ns.nl, m.ns.nl, loginapi2.ns.nl, loginapi2.acceptatie.ns.nl, loginapi.ns.nl, loginapi.acceptatie.ns.nl, login2.ns.nl, login2.acceptatie.ns.nl, login.ns.nl, login.acceptatie.ns.nl, ews-rpx.ns.nl, ews-cupz.ns.nl, ews-andp.ns.nl, ccweb.ns.nl, acc.ns.nl

Error code: SSL_ERROR_BAD_CERT_DOMAIN

Eh, nee…


Mijn telefoon
Evil me
NS server
„Hoi, ik wil TLS praten“
„Ok, hier mijn cert.“
„Da’s nep joh!
Jij bent ns.nl niet!“
„Hoi, ik wil TLS praten“
„Ok, hier mijn cert.“

Wat is een certificaat?

- Een document dat vertelt: „De derde-partij ABC bevestigt dat de public key hoort bij partij XYZ en domein mijnXYZdomein.nl“
- Ze bevatten informatie over partij ABC en partij XYZ
  - Ook de public key van partij XYZ
  - Certificaat is ondertekend met private key van partij ABC
- Ze bevatten info over wat XYZ met het key pair mag doen
  - Mag XYZ dit key pair gebruiken om data te ondertekenen?
  - Mag XYZ dit key pair gebruiken om andere certificaten te ondertekenen?
  - Van wanneer tot wanneer is dit certificaat geldig?

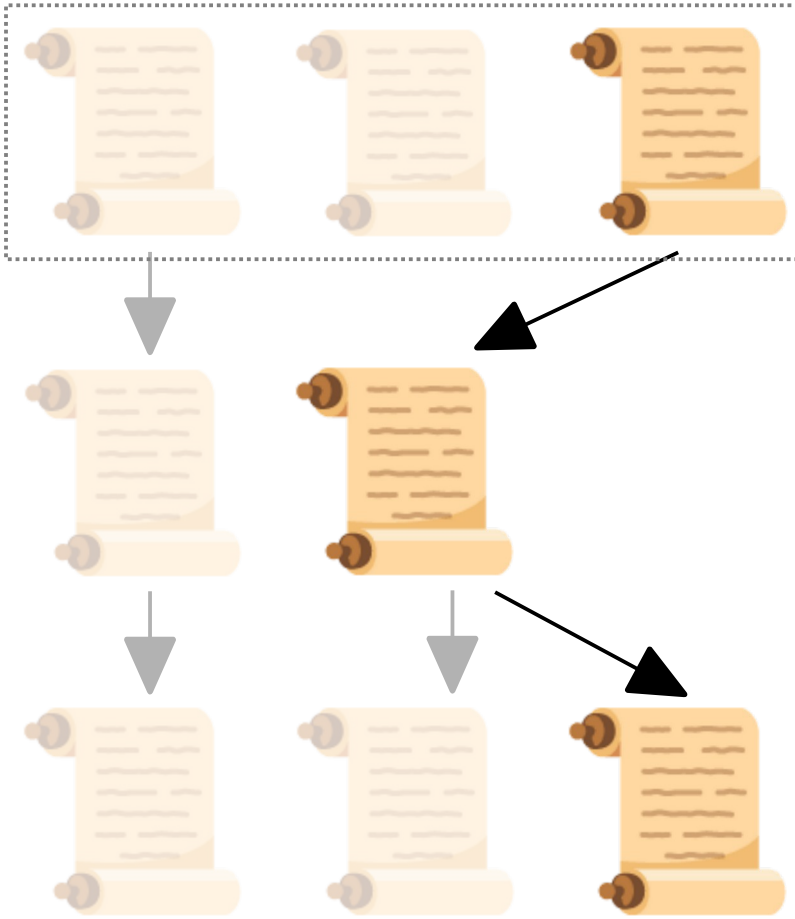
Certificaten kunnen een keten vormen

- Dat noem je een certificaatketen
  - Certificaat A ondertekent B, certificaat B ondertekent C, etc.
- Voor het ondertekenen gebruikt men een public key
- Voorbeeld:
  - Domein deesiedjoeks.nl heeft public key 123… volgens certificaat *Let's Encrypt R3*
  - Certificaat *Let's Encrypt R3* heeft public key 456… volgens certificaat *Internet Security Research Group ISRG Root X1*
- Certificaten kun je ook zelf genereren en ondertekenen
  - Dat noem je een self-signed (of snake-oil) certificaat

Aan het einde van de keten

- De certificaten aan het eind van de keten zitten in een trust store
  - Windows heeft er een, je telefoon ook, en je browser kan er een hebben
  - De certificaten in de trust store zijn altijd geldig, hoef je niet te checken
  - Certificaat *Internet Security Research Group ISRG Root X1* zit daar in
- Die certificaten zijn van Certificate Authorities (CAs)
  - Dat zijn eigenlijk bedrijven die voor geld jouw certificaat ondertekenen
- Alles is dus een kwestie van vertrouwen

Voorbeeld van een certificaat keten



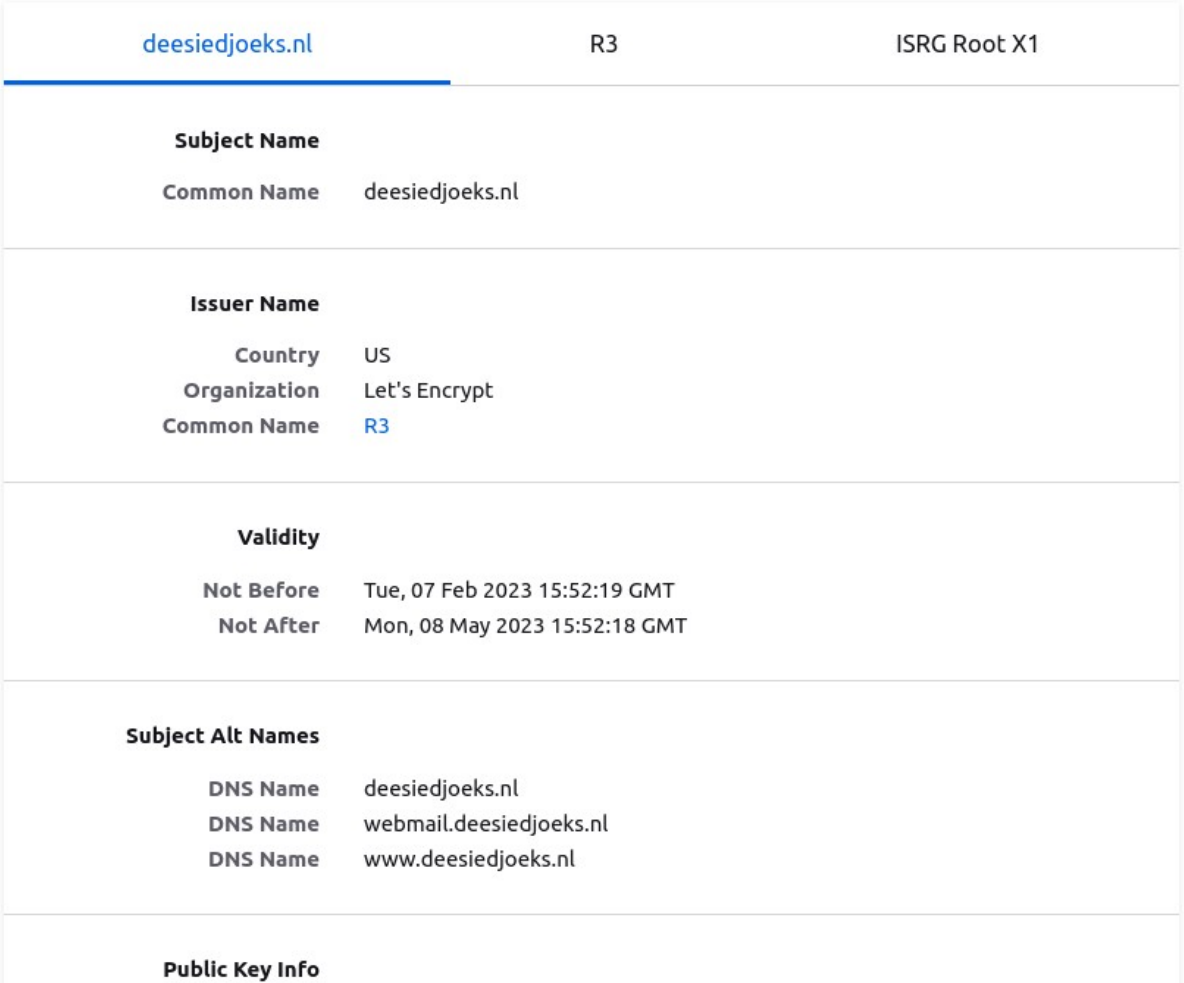
ISRG Root X1

Trust store van Android

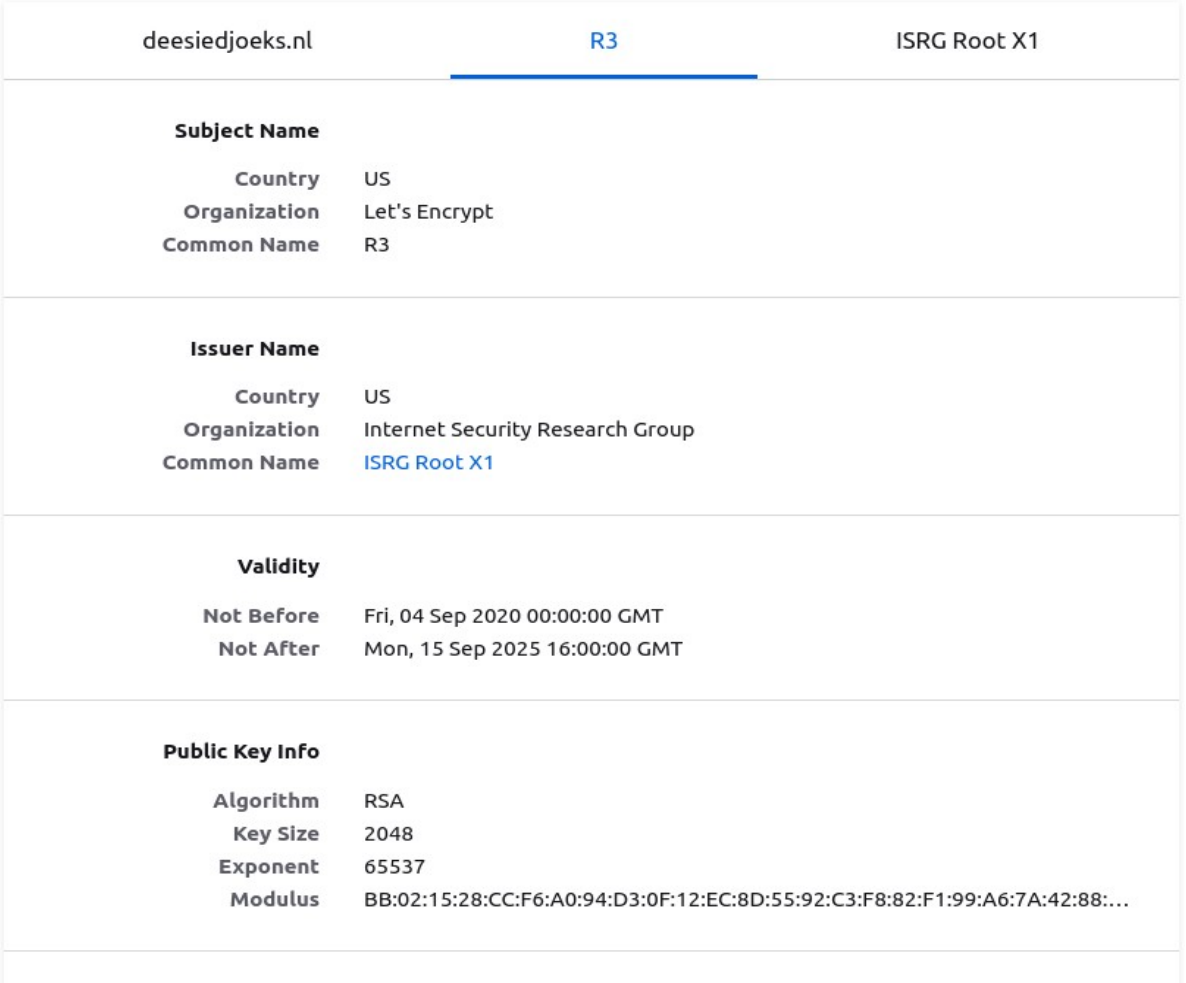
Let's Encrypt R3

Certificaat
deesiedjoeks.nl

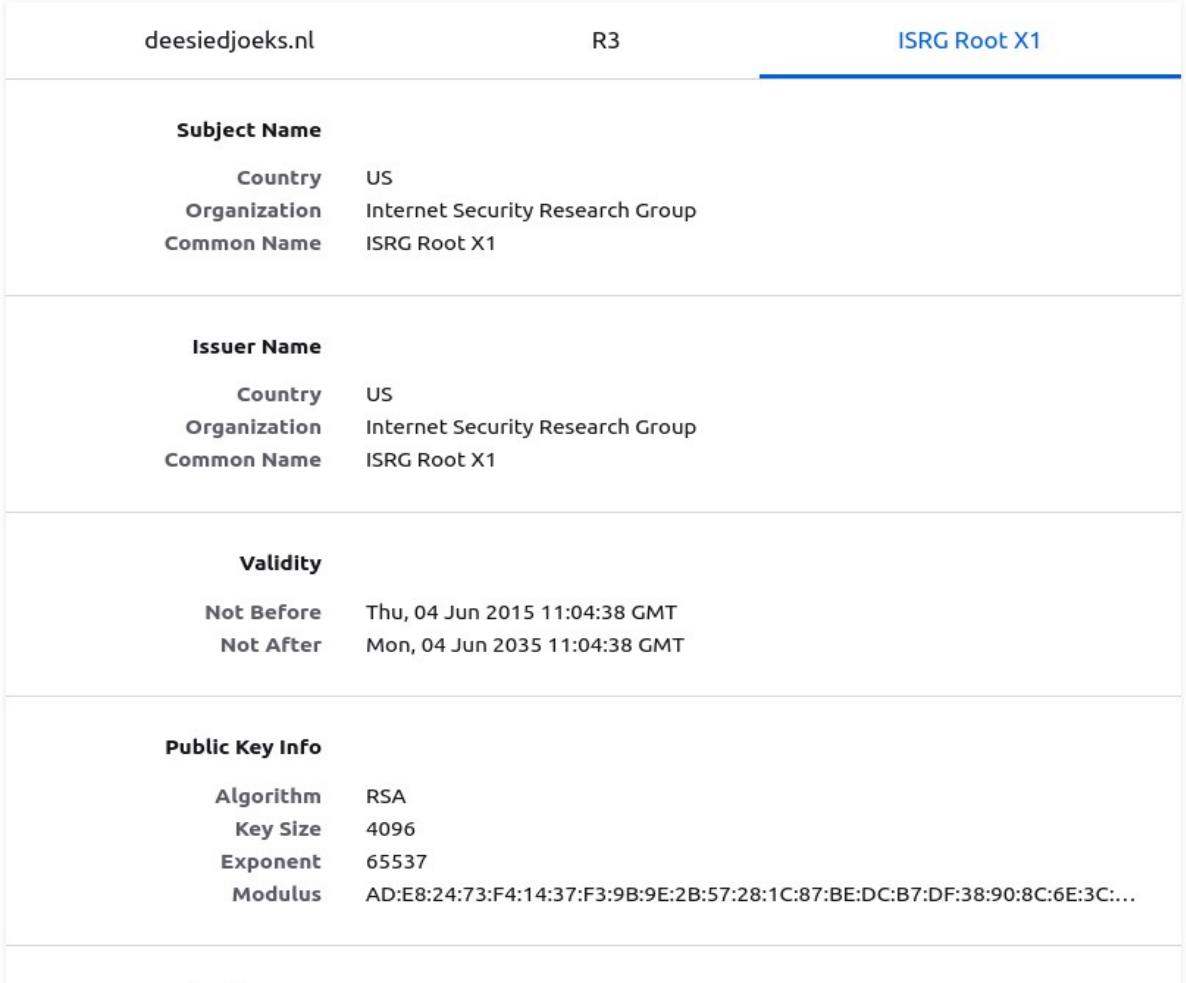
De browser geeft me gelijk



De browser geeft me gelijk



De browser geeft me gelijk

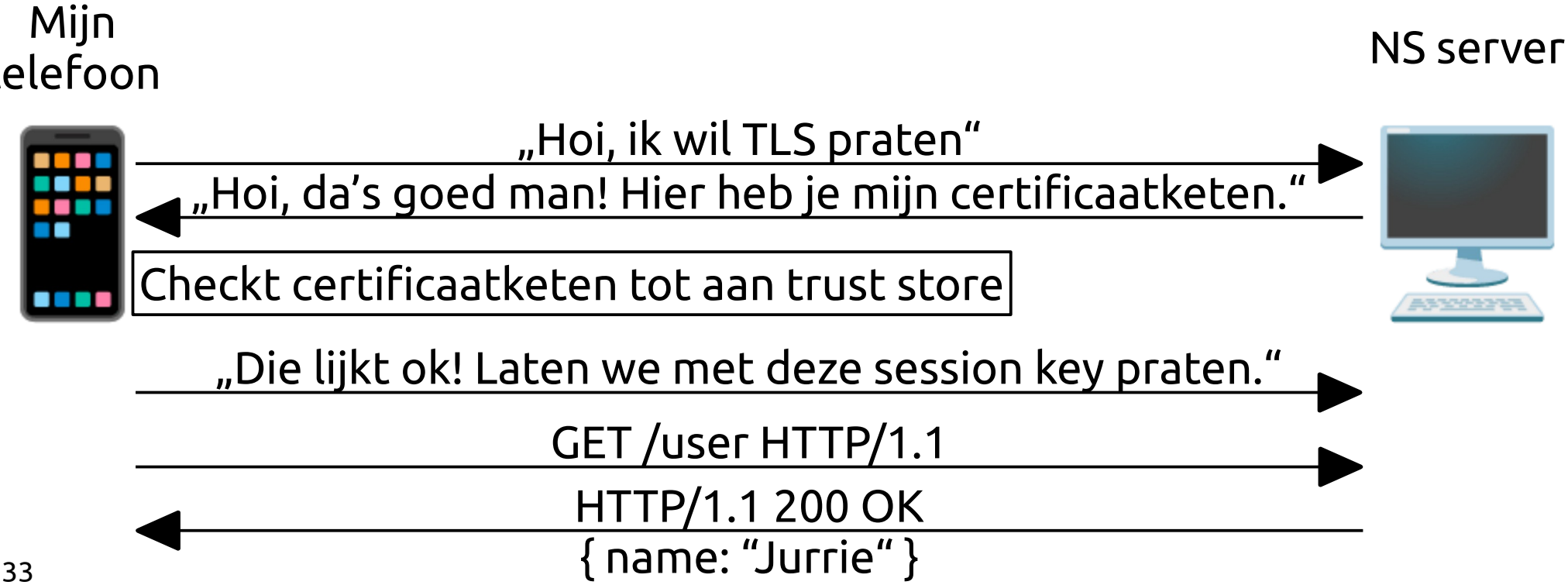


Trouwens…

- Om je certificaat te laten opnemen in een OS trust store kost vaak veel
- Vertrouw jij echt *alle* certificaten in jouw OS trusts?
  - Google Trust Services LLC? (Mountain View, California, US)
  - iTrusChina Co.,Ltd.? (Haidian, Beijing, China)
  - eMudhra Technologies Limited (Teynampet, Tamil Nadu, India)
  - Internet Security Research Group (San Francisco, California, US)
    - Geeft *gratis* certificaten uit aan iedereen die het maar wil?

- Aan welke criteria moeten Certificate Authorities voldoen?
  - Mozilla Root Store Policy - bevat 170 certificaten
  - Microsoft Trusted Root Program - bevat 251 certificaten
  - Chrome Root Program Policy - bevat 131 certificaten

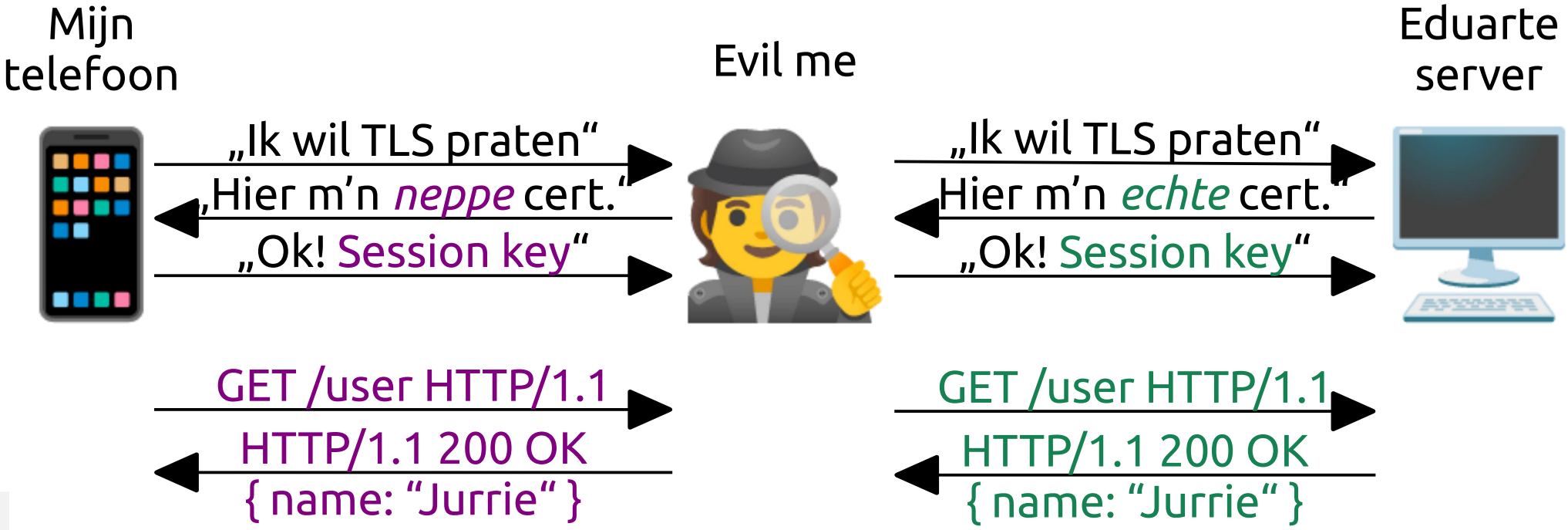
Samengevat: praten via TLS



En nu de MITM attack!

- We willen de data lezen die tussen de Android app en de server wordt gestuurd
- We gaan de Android emulator van Google gebruiken
  - Wat hebben we nodig om de data te lezen?
- Het lijkt ons leuk om niet de NS app te gebruiken, maar Eduarte Student!

Dit willen we bouwen…



We gebruiken ons eigen certificaat

- We gebruiken mitmproxy als een proxy ( )
  - Deze zit tussen de emulator (waar de app in draait) en de server
- Data van/naar de Eduarte server ( )…
  - Begint met TLS handshake met public key en certificaat van Eduarte server
  - Gebruikt de session key die afgesproken wordt in die TLS handshake
- Data van/naar de app ( ) will
  - Begint met een TLS handshake met onze eigen key pair en certificaat
  - Gebruikt de session key die afgesproken wordt in die TLS handshake
- mitmproxy ontsleutelt en versleutelt tussen beide kanten

Maar hoe denkt die app daarover?

- De data komt nu van mitmproxy, niet van Eduarte
- De certificaatketen eindigt niet met certificaat uit trust store
  - Het eindigt met ons zelfgemaaktecertificaat, voor onze eigen key
- We moeten ons zelfgemaakte certificaat toevoegen aan de OS trust store
  - Hiervoor hebben we ‚root' nodig in Android
  - Dat is een gebruiker vergelijkbaar met Administrator in Windows; deze gebruiker heeft alle rechten op het lokale systeem

Let’s do this!

- Ladies and gentlemen – start your laptops!!
- Je vind deze presentatie als PDF op het bureaublad
  - Open de presentatie en ga verder in je eigen tempo

Intro: algemene zaken

- De meeste commando's doe je op de command line
  - Om een command line te openen:
  - Windows + R, typ `cmd.exe`, druk enter
- Dingen tussen `%` (zoals `%LOCALAPPDATA%`) zijn variabelen
  - Die wijzen naar iets als C:\Users\USER\AppData\Local
  - De command line vervangt de variabelen automatisch
- Een Android applicatie zit in een APK bestand

Intro: als er iets niet werkt

- Kopiëren uit een PDF verandert soms tekens zoals –
  - Typ in dat geval het commando zelf; gebruikt geen copy+paste
- Als de emulator niet herstart…
  - Probeer dan het argument `-no-snapshot` aan het commando toe te voegen

Stap één: start Android Studio

- Het staat al op je laptop

Maak een AVD (Android Virtual Device)

- Ga naar ‚More actions' → ‚Virtual Device Manager'
- Klik ‚Create virtual device'
- Selecteer ‚Phone' → ‚Pixel 4 (5,7")' → Klik ‚next'
- Selecteer ‚x86 images' → Selecteer:
  - Release ‚R',API level 30, ABI x86_64, target ‚Android 11'
  - Kies *niet* voor targets ‚Google APIs' of ‚Google Play'
- Klik ‚next' en klik ‚finish'
- Je hebt AVD ‚Pixel 4 API 30' gemaakt

Start AVD via command line

- `%LOCALAPPDATA%\Android\Sdk\emulator\emulator.exe -list-avds`
  - Dit commando zou iets moeten zeggen als „Pixel_4_API_30“
- `%LOCALAPPDATA%\Android\Sdk\emulator\emulator.exe -avd Pixel_4_API_30 -writable-system`
  - Dit start de AVD
- Doe, in een *nieuwe* command line, `%LOCALAPPDATA%\Android\Sdk\platform-tools\adb.exe devices`
  - Dit commando zou iets moeten zeggen als „emulator-5554 device“
- In de AVD: kies app ‚WebView‘ (dat is een eenvoudige browser)
  - Controleer dat je naar HTTPS sites kunt browsen

Installeer Eduarte Student applicatie (APK)

- Navigeer naar de directory waar het APK bestand staat:
  - `cd %USERPROFILE%\Desktop`
- Installeer de APK in de AVD:
  - `%LOCALAPPDATA%\Android\Sdk\platform-tools\adb.exe install Eduarte_Student_1.5.17.apk`
- Start de Eduarte Student app in AVD

Inloggen op Eduarte Student

- Open de app
- De testomgeving moeten we eerst aanzetten. Voer de volgende commandos uit:
  - `%LOCALAPPDATA%\Android\Sdk\platform-tools\adb.exe emu sensor set acceleration 100:100:100`
  - `%LOCALAPPDATA%\Android\Sdk\platform-tools\adb.exe emu sensor set acceleration 0:0:0`
- Nu is bij ‚Selecteer een school' ook ‚Topicus' verschenen. Kies deze.
- Kies voor omgeving ‚TEST' en dan knop ‚Omgeving wisselen'
- Kies nu bij ‚Selecteer een school'voor ‚SDT' en log in met
  - Gebruikersnaam: 436906
  - Wachtwoord: Onderwijs@12

Start mitmproxy

- Start `mitmweb` via het icoontje op het bureaublad
  - Als Windows Firewall zeurt, druk dan op *toestaan*!
  - De web interface zit standaard op http://127.0.0.1:8081
  - De proxy luistert standaard op 127.0.0.1 poort 8080
- Bij eerste keer starten worden er op de achtergrond certificaten voor je gegenereerd
  - Die vind je in `%USERPROFILE%\.mitmproxy\`

Start AVD met proxy

- Stop de AVD en hestart zodat mitmproxy als proxy gebruikt wordt:
  - `%LOCALAPPDATA%\android\Sdk\emulator\emulator.exe -avd Pixel_4_API_30 -writable-system -http-proxy 127.0.0.1:8080`
- Controleer dat je *in* de AVD naar http://mitm.it kunt browsen
  - Als je de melding „If you can see this, traffic is not passing through mitmproxy.“ ziet, dan is er iets mis
  - Zie je „Install mitmproxy’s Certificate Authority“ dan is alles ok!
- Je kunt in de AVD nu *niet* surfen naar HTTPS websites
  - Waarom is dat?

Ons certificaat in de OS trust store stoppen

- We gaan het certificaat dat mitmproxy gebruikt toevoegen aan de trust store van Android
- Deze trust store vind je hier:
  - Settings → Security → Encryption & credentials → Trusted credentials
- We moeten het toevoegen aan de system trust store
  - User trust store werkt niet

Bestandssysteem van AVD schrijfbaar maken

- Standaard kunnen we niet schrijven naar de OS trust store
- Doe het volgende in command line terwijl de AVD aan staat:
  - `%LOCALAPPDATA%\Android\Sdk\platform-tools\adb.exe root`
  - `%LOCALAPPDATA%\Android\Sdk\platform-tools\adb.exe shell avbctl disable-verification`
  - `%LOCALAPPDATA%\Android\Sdk\platform-tools\adb.exe reboot`
  - Het herstarten van de AVD duurt even…

Bestandssysteem van AVD schrijfbaar maken

- `%LOCALAPPDATA%\Android\Sdk\platform-tools\adb.exe root`
- `%LOCALAPPDATA%\Android\Sdk\platform-tools\adb.exe remount`
  - Als je „Now reboot your device for settings to take effect“ ziet, dan moet je de AVD herstarten en de commando’s opnieuw doen
  - Als je *alleen* ziet „remount succeeded“ dan kun je verder

mitmproxy CA toevoegen aan trust store

- `%LOCALAPPDATA%\Android\Sdk\platform-tools\adb.exe push %USERPROFILE%\.mitmproxy\mitmproxy-ca-cert.cer /system/etc/security/cacerts/c8750f0d.0`
- `%LOCALAPPDATA%\Android\Sdk\platform-tools\adb.exe shell chmod 644 /system/etc/security/cacerts/c8750f0d.0`
- `%LOCALAPPDATA%\Android\Sdk\platform-tools\adb.exe reboot`
- Controleer in Android of het mitmproxy certificaat er bij staat
  - Settings → Security → Encryption & credentials → Trusted credentials

…En werkt het nu?

- Je zou nu naar HTTPS sites moeten kunnen browsen
  - In mitmproxy zie je dan het onderschepte verkeer
- Je zou nu de Eduarte Student app weer moeten kunnen gebruiken
- Probeer eens uit te loggen en weer in te loggen
  - Zie je in mitmproxy jouw wachtwoord verstuurd worden naar de Eduarte server? (Zoek naar een https://d-login.educus.nl.)
  - Met de communicatie die je nu onderschept kun je dus de Eduarte app nadoen (remember: ik wilde met de NS app nadoen)

Huiswerk!

- Apps zoals TooGoodToGo draaien niet zonder Google Play
  - In een AVD mét Google Play heb je geen root rechten
  - Maar je kunt here en here kijken...
- Apps zoals TikTok of Instagram draaien niet in een emulator
  - Je kunt Memu, NoxPlayer, LDPlayer, BlueStacks, … proberen
  - Heb je een fysiek apparaat met root? Kijk dan here en here…
- Apps gebouwd met Flutter gebruiken de OS trust store niet
  - Een mitm attack lukt wel, maar is een heel ander rabbit hole

Maar! (De verplichte disclaimer)

- Onthou wat Ome Jurrie zei:
- Als je een beveilignsprobleem tegenkomt, hou je aan:
  - Stel het betreffende bedrijf op de hoogte
  - Help ze om de boel te fixen
  - Geef ze tijd om de boel te fixen
  - …en ga pas DAN opscheppen bij je vrienden!

Bedankt!

- Bedankt voor jullie aandacht
- Neem gerust contact met mij op over wat dan ook!
- jurrie.overgoor@topicus.nl
- @Jurrie Overgoor op Slack
- linkedin.com/in/jurrieovergoor/
- github.com/Jurrie/
- …of reis met me mee in de bus Raalte Deventer