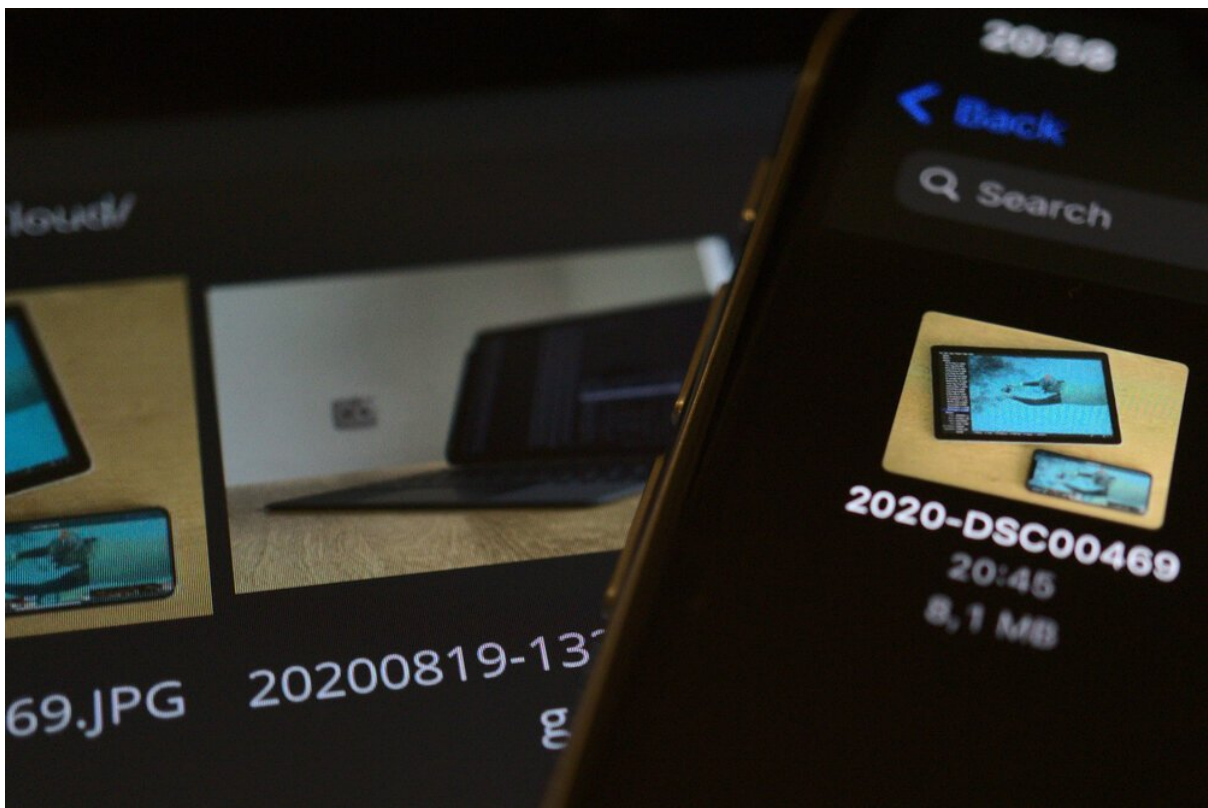


# Bestanden naadloos synchroniseren tussen smartphone en tablet

*Unison gebruiken om automatisch te synchroniseren tussen  
GNU/Linux en de iPhone*

Willem L. Middelkoop

16 sep. 2020



Om mijn workflow te optimaliseren, zocht ik naar een manier om naadloos toegang te krijgen tot dezelfde bestanden op zowel mijn computer als smartphone. Dit is handig wanneer je snel bestanden vanaf je computer wilt versturen via je smartphone met behulp van verschillende messaging apps en vice versa. Lees verder om te ontdekken hoe ik dit heb gedaan.

## Het doel: naadloos gedeelde bestanden

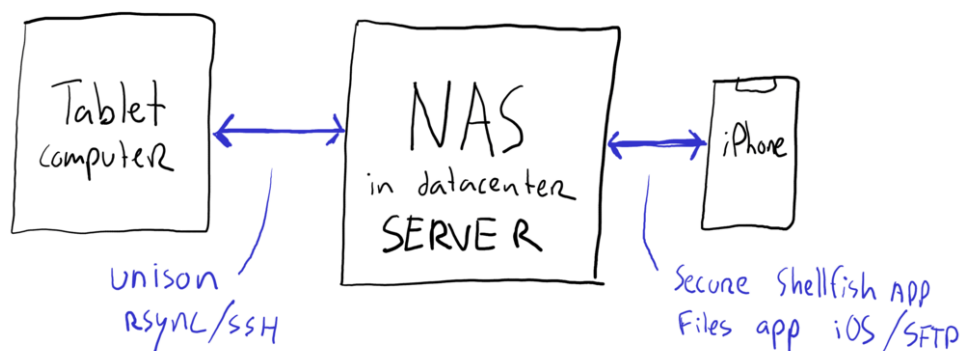
Wanneer ik werk, gebruik ik meestal zowel een tablet als een smartphone. Ik gebruik een [tablet](#) als hoofdcomputer. Mijn iPhone is erg handig als tweede scherm. Ik gebruik het ook om bestanden, screenshots en foto's te versturen naar mensen via de vele verschillende berichten-apps die op iOS draaien.

In goede of slechte zin is er een verschuiving gaande van e-mail naar apps zoals Slack, WhatsApp, iMessage en Signal. Mijn iPhone is de toegangspoort tot deze communicatiekanalen. Omdat ik iCloud Drive of DropBox niet wil gebruiken, had ik een andere manier nodig om mijn bestanden naadloos op zowel mijn tablet als smartphone te hebben.

## Maak kennis met Unison: de multi-platform bestandssynchronisator

Na een snelle zoekopdracht vond ik [Unison](#) in de [Debian software repository](#). Omdat mijn tablet op Debian GNU/Linux draait, zocht ik naar een manier om bestanden veilig te synchroniseren met behulp van open standaarden. Unison doet dit met behulp van rsync over SSH/SFTP. Dit betekent dat het werkt met mijn [bestaande NAS server](#) die in een datacenter staat.

Hoewel je bestandsdeling direct tussen je tablet en een smartphone kunt instellen, gebruik ik de centrale NAS-server om communicatie tussen mijn apparaten mogelijk te maken, zelfs als ze zich niet op hetzelfde netwerk bevinden. Dit gebeurt wanneer een (of beide) apparaat(en) op een mobiel netwerk is/zijn. Om toegang te krijgen tot bestanden vanaf mijn iPhone, gebruik ik de Secure ShellFish-app die mooi integreert met de native iOS-bestanden-app.



*Synchroniseren van bestanden tussen mijn tablet en iPhone: de NAS server fungeert als tussenstation; overbruggen van verschillen in protocol en verbinding*

Het instellen van Unison is eenvoudig. Je moet de Engelse documentatie op de [Unison website](#) bekijken of deze [geweldige Unison handleiding in het Duits](#) volgen. Je kunt Unison uitvoeren vanaf de commandoregel, maar ik vind het prettig om het automatisch te laten draaien met behulp van een vooraf geconfigureerd profiel en een regelmatig uitgevoerde cron-taak.

```
willem@WillemSurface:~$ cat .unison/WillemBaseCloud.prf
# Unison preferences
root = /home/willem/Cloud
root = ssh://willem@base.willem.com/Cloud
sshargs = -C

batch=true
auto=true
confirmbigdel=false
willem@WillemSurface:~$
```

*Mijn Unison profiel met parameters voor de 'WillemCloud'*

Het Unison-configuratieprofiel is eenvoudig: je definieert twee rootmappen en je kunt enkele opties gebruiken om het standaardgedrag aan te passen. Ik heb mijn profiel zo ingesteld dat het dingen in batches doet en automatisch beslist welke bestanden bewaard moeten blijven. Ik heb een speciale "Cloud"-map die ik deel tussen mijn apparaten, het fungeert als een soort "Desktop"-map. Bekijk de Unison man-pagina voor alle mogelijke parameters (probeer 'man unison' in je terminal te typen).

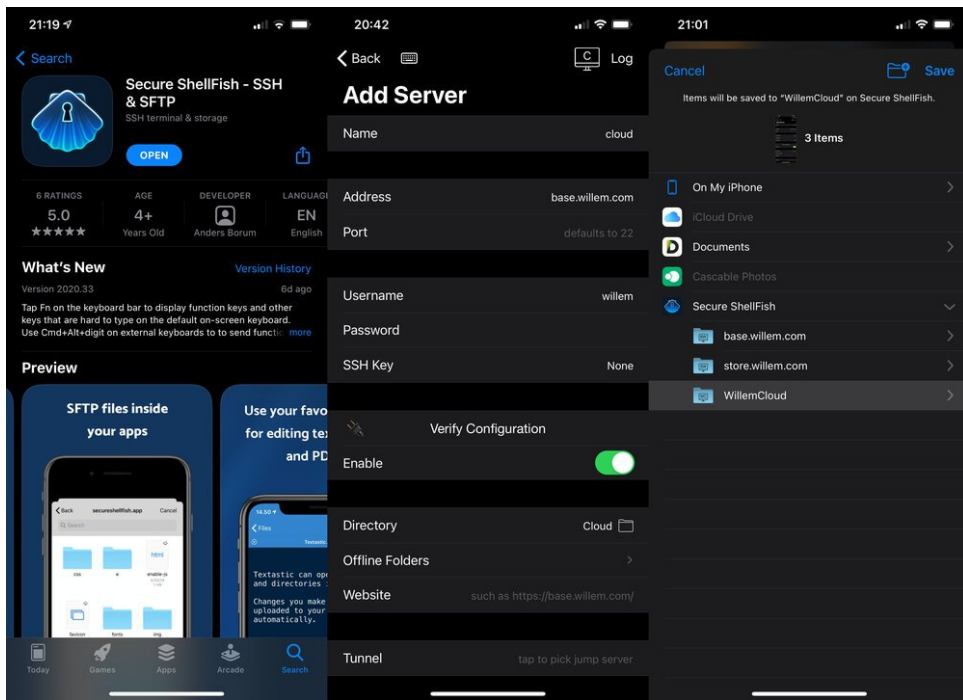
Authenticatie gebeurt met behulp van de SSH-configuratie van je systeem. Je kunt (en zou moeten) [authenticatie met gedeelde clientsleutels](#) instellen. Dit maakt wachtwoordloze SSH-communicatie tussen je client en server mogelijk. Zodra je je Unison-profiel hebt getest en geconfigureerd, kun je een cron-taak maken om je bestanden automatisch te synchroniseren. Omdat bestandsoverdrachten langer kunnen duren, moet de cron-taak alleen worden uitgevoerd als de vorige uitvoering is voltooid. Je kunt dit doen met behulp van een PID-bestand, dat als controle fungeert.

```
# Command to automatically sync files using unison, preventing double execution using a loc/pid file:
test -e /var/lock/unison-WillemBaseCloud && exit 0 || (touch /var/lock/unison-WillemBaseCloud;unison WillemBaseCloud;rm /var/lock/unison-WillemBaseCloud)
```

*Mijn cron commando is verpakt in een controle om te voorkomen dat synchronisatie start wanneer deze al loopt*

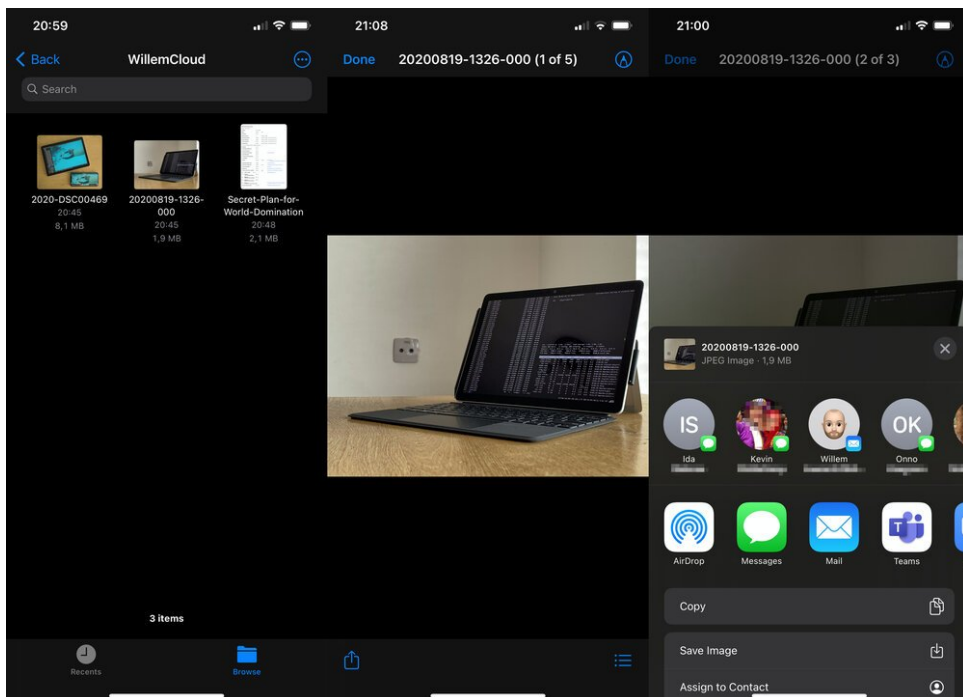
## Secure ShellFish instellen op iOS

Zodra je computer bestanden synchroniseert met je NAS, moet je je smartphone instellen. Ik gebruik een iPhone en daarom zocht ik naar de beste app die goed integreert met de native iOS-bestanden-app. Na het testen van verschillende apps vond ik [Secure ShellFish](#) de beste:



*Download en configureer Secure ShellFish om verbinding te maken met mijn “Willem-Cloud”*

Het mooie van de Secure ShellFish-app is dat deze integreert met de native iOS-bestanden-app (en de onderliggende iOS-opslagprovider-API's). Dit betekent dat alle apps die toegang hebben tot lokale opslag verbinding kunnen maken met je NAS! Dit maakt toegang tot je bestanden mogelijk vanuit praktisch elke app.

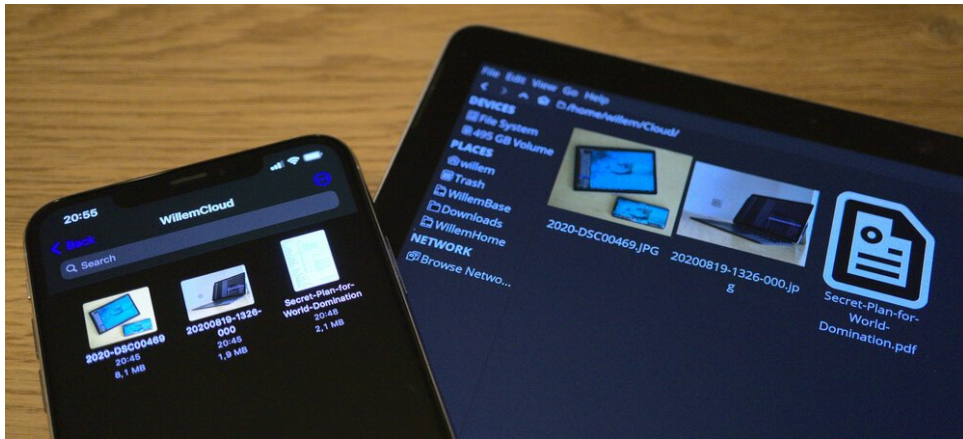


*Toegang tot bestanden via de native iOS Bestanden app stelt je in staat om veel verschillende apps en shortcuts te gebruiken*

## Conclusie

Het optimaliseren van je dagelijkse workflows kan je op de lange termijn echt enorme voordelen opleveren. Door het mogelijk te maken om naadloos bestanden te delen tussen mijn tablet en smartphone, heb ik precies dat gedaan!

Door de synchronisatie te automatiseren, verdwijnt het naar de achtergrond. Ik hoef er niet langer over na te denken als een geavanceerd proces met verschillende protocollen, servers en apps. Het werkt gewoon, en dat is precies hoe ik het wil!



*Dezelfde bestanden op zowel mijn smartphone als tablet - het werkt gewoon!*