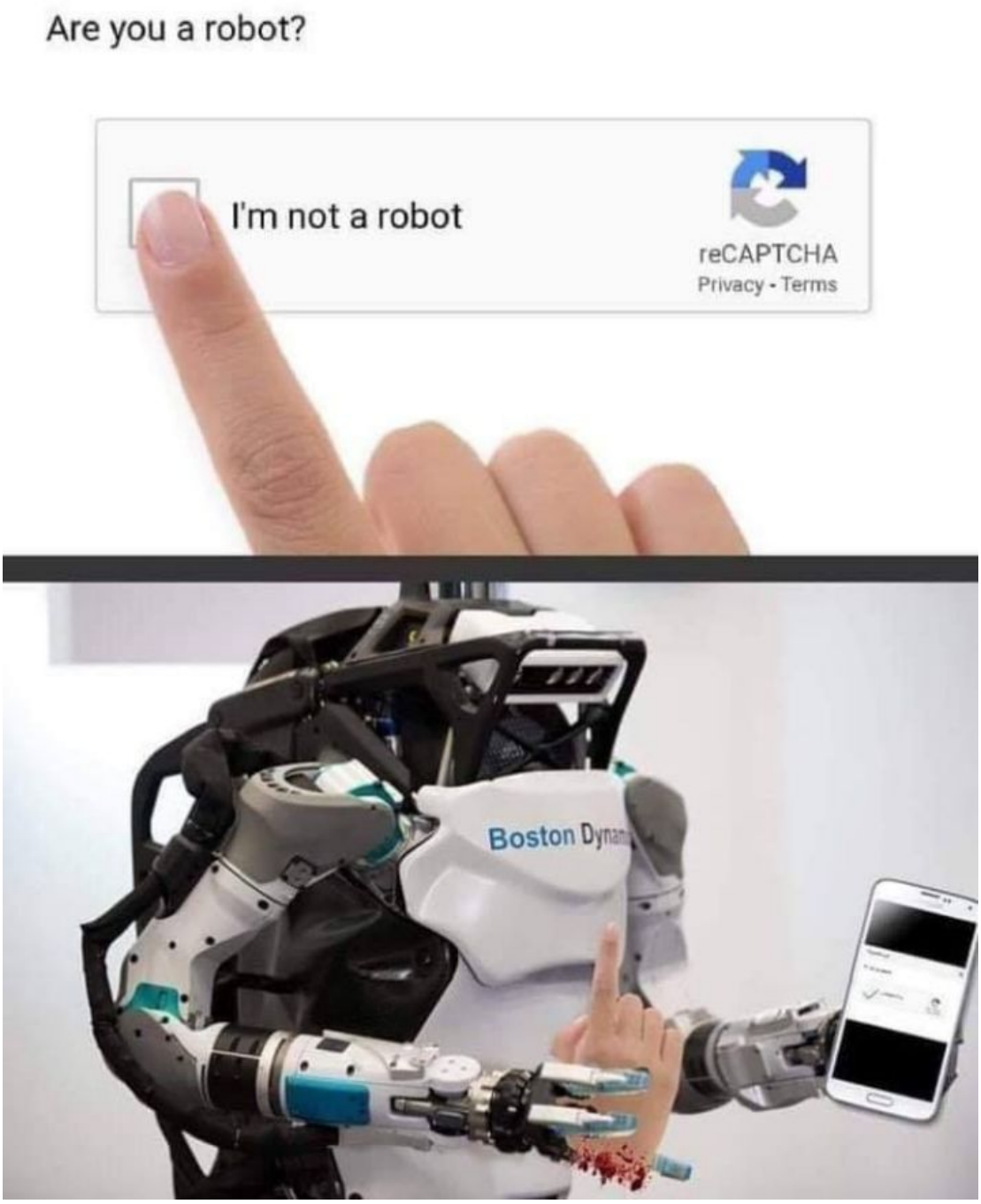
Security Now! #734 - 10-01-19

The Joy of Sync

This week on Security Now!

With this week's "Joy of Sync" podcast we focus upon the latest state-of-the-art secure solutions for cross-device, cross-location device synchronization. But before we delve into that abyss... we'll update on Mozilla's recently announced plans to gradually and carefully bring DNS-over-HTTPS to all Firefox users in the US. It turns out it's not quite the slam-dunk that we might imagine. We'll also check-in with the EFF to see what they think, and remind our listeners about the 100% free VPN offering coming from our friends at Cloudflare.



Security News

Mozilla moves CAUTIOUSLY forward on DoH
Mozilla plans to gradually enable DNS-over-HTTPS for its US users. Overall, DoH tests have gone well so far. So Mozilla will start rolling out DoH to a small set of US users, then gradually roll it out for more users.

However, DNS is tricky to mess with, because it is used for more than only looking up public domains. It is used as a means of filtering and also as a means of providing access to local non-public resources.

On September 6th, Selena Deckelmann, Mozilla's Senior Director of Firefox Engineering, posted some very interesting details that I know our listeners will find fascinating...

https://blog.mozilla.org/futurereleases/2019/09/06/whats-next-in-making-dns-over-https-the-default/

Title: "*What's next in making Encrypted DNS-over-HTTPS the Default*"

In 2017, Mozilla began working on the DNS-over-HTTPS (DoH) protocol, and since June 2018 we've been running experiments in Firefox to ensure the performance and user experience are great. We've also been surprised and excited by the more than 70,000 users who have already chosen on their own to explicitly enable DoH in Firefox Release edition. We are close to releasing DoH in the USA, and we have a few updates to share.

After many experiments, we've demonstrated that we have a reliable service whose performance is good, that we can detect and mitigate key deployment problems, and that most of our users will benefit from the greater protections of encrypted DNS traffic. We feel confident that enabling DoH by default is the right next step. When DoH is enabled, users will be notified and given the opportunity to opt out.

This post includes results of our latest experiment, configuration recommendations for systems administrators and parental controls providers, and our plans for enabling DoH for some users in the USA.

Our latest DoH experiment was designed to help us determine how we could deploy DoH, honor enterprise configuration and respect user choice about parental controls.

We had a few key learnings from the experiment.

- We found that OpenDNS' parental controls and Google's safe-search feature were rarely configured by Firefox users in the USA. In total, 4.3% of users in the study used OpenDNS' parental controls or safe-search. Surprisingly, there was little overlap between users of safe-search and OpenDNS' parental controls. As a result, we're reaching out to parental controls operators to find out more about why this might be happening.

- We found 9.2% of users triggered one of our split-horizon heuristics. The heuristics were triggered in two situations: when websites were accessed whose domains had non-public suffixes, and when domain lookups returned both public and private (RFC 1918) IP addresses. There was also little overlap between users of our split-horizon heuristics, with only 1% of clients triggering both heuristics.

Now that we have these results, we want to tell you about the approach we have settled on to address managed networks and parental controls. At a high level, our plan is to:

- Respect user choice for opt-in parental controls and disable DoH if we detect them;
- Respect enterprise configuration and disable DoH unless explicitly enabled by enterprise configuration; and
- Fall back to operating system defaults for DNS when split horizon configuration or other DNS issues cause lookup failures.

We're planning to deploy DoH in "fallback" mode; that is, if domain name lookups using DoH fail or if our heuristics are triggered, Firefox will fall back and use the default operating system DNS. This means that for the minority of users whose DNS lookups might fail because of split horizon configuration, Firefox will attempt to find the correct address through the operating system DNS.

In addition, Firefox already detects that parental controls are enabled in the operating system, and if they are in effect, Firefox will disable DoH. Similarly, Firefox will detect whether enterprise policies have been set on the device and will disable DoH in those circumstances. If an enterprise policy explicitly enables DoH, which we think would be awesome, we will also respect that. If you're a system administrator interested in how to configure enterprise policies, please find documentation here. If you find any bugs, please report them here.

Options for Providers of Parental Controls

We're also working with providers of parental controls, including ISPs, to add a canary domain to their blocklists. This helps us in situations where the parental controls operate on the network rather than an individual computer. If Firefox determines that our canary domain is blocked, this will indicate that opt-in parental controls are in effect on the network, and Firefox will disable DoH automatically. If you are a provider of parental controls, details are available here. Please reach out to us for more information at doh-canary-domain@mozilla.com. We're also interested in connecting with commercial blocklist providers, in the US and internationally.

This canary domain is intended for use in cases where users have opted in to parental controls. We plan to revisit the use of this heuristic over time, and we will be paying close attention to how the canary domain is adopted. If we find that it is being abused to disable DoH in situations where users have not explicitly opted in, we will revisit our approach.

Plans for Enabling DoH Protections by Default

We plan to gradually roll out DoH in the USA starting in late September. Our plan is to start slowly enabling DoH for a small percentage of users while monitoring for any issues before enabling for a larger audience. If this goes well, we will let you know when we're ready for 100% deployment. For the moment, we encourage enterprise administrators and parental control providers to check out our config documentation and get in touch with any questions.

The EFF weighs in on DoH
Title: "Encrypted DNS Could Help Close the Biggest Privacy Gap on the Internet. Why Are Some Groups Fighting Against It?"
https://www.eff.org/deeplinks/2019/09/encrypted-dns-could-help-close-biggest-privacy-gap-internet-why-are-some-groups

Thanks to the success of projects like Let's Encrypt and recent UX changes in the browsers, most page-loads are now encrypted with TLS. But DNS, the system that looks up a site's IP address when you type the site's name into your browser, remains unprotected by encryption.

Because of this, anyone along the path from your network to your DNS resolver (where domain names are converted to IP addresses) can collect information about which sites you visit. This means that certain eavesdroppers can still profile your online activity by making a list of sites you visited, or a list of who visits a particular site. Malicious DNS resolvers or on-path routers can also tamper with your DNS request, blocking you from accessing sites or even routing you to fake versions of the sites you requested.

A team of engineers is working to fix these problems with "DNS over HTTPS" (or DoH), a draft technology under development through the Internet Engineering Task Force that has been championed by Mozilla. DNS over HTTPS prevents on-path eavesdropping, spoofing, and blocking by encrypting your DNS requests with TLS.

Alongside technologies like TLS 1.3 and encrypted SNI, DoH has the potential to provide tremendous privacy protections. But many Internet service providers and participants in the standardization process have expressed strong concerns about the development of the protocol. The UK Internet Service Providers Association even went so far as to call Mozilla an "Internet Villain" for its role in developing DoH.

ISPs are concerned that DoH will complicate the use of captive portals, which are used to intercept connections briefly to force users to log on to a network, and will make it more difficult to block content at the resolver level. DNS over HTTPS may undermine plans in the UK to block access to online pornography (the block, introduced as part of the Digital Economy Act of 2017, was planned to be implemented through DNS).

Members of civil society have also expressed concerns over plans for browsers to automatically use specific DNS resolvers, overriding the resolver configured by the operating system (which today is most often the one suggested by the ISP). This would contribute to the centralization of Internet infrastructure, as thousands of DNS resolvers used for web requests would be replaced by a small handful.

That centralization would increase the power of the DNS resolver operators chosen by the browser vendors, which would make it possible for those resolver operators to censor and monitor browser users' online activity. This capability prompted Mozilla to push for strong policies that forbid this kind of censorship and monitoring. The merits of trusting different entities for this purpose are complicated, and different users might have reasons to make different choices. But to avoid having this technology deployment produce such a powerful centralizing effect, EFF is calling for widespread deployment of DNS over HTTPS support by Internet service providers themselves. This will allow the security and privacy benefits of the

technology to be realized while giving users the option to continue to use the huge variety of ISP-provided resolvers that they typically use now. Several privacy-friendly ISPs have already answered the call. We spoke with Marek Isalski, Chief Technology Officer at UK-based ISP Faelix, to discuss their plans around encrypted DNS.

Faelix has implemented support for DNS over HTTPS on their pdns.faelix.net resolver. They weren't motivated by concerns about government surveillance, Marek says, but by "the monetisation of our personal data." To Marek, supporting privacy-protecting technologies is a moral imperative. "I feel it is our calling as privacy- and tech-literate people to help others understand the rights that GDPR has brought to Europeans," he said, "and to give people the tools they can use to take control of their privacy."

EFF is very excited about the privacy protections that DoH will bring, especially since many Internet standards and infrastructure developers have pointed to unencrypted DNS queries as an excuse to delay turning on encryption elsewhere in the Internet. But as with any fundamental shift in the infrastructure of the Internet, DoH must be deployed in a way that respects the rights of the users. Browsers must be transparent about who will gain access to DNS request data and give users an opportunity to choose their own resolver. ISPs and other operators of public resolvers should implement support for encrypted DNS to help preserve a decentralized ecosystem in which users have more choices of whom they rely on for various services. They should also commit to data protections like the ones Mozilla has outlined in their Trusted Recursive Resolver policy. With these steps, DNS over HTTPS has the potential to close one of the largest privacy gaps on the web.

With all of this happy news, we do need to remember that even though our communications are encrypted, and now our DNS lookups will be encrypted and hidden, the IP address to which we are communicating remains in the clear and fully visible.

As we know, the only good solution to that problem is to tunnel all of our browser traffic through a virtual private network.

Interestingly, our friends at Cloudflare have an offering called WARP that promises to perhaps also fix that…

https://blog.cloudflare.com/1111-warp-better-vpn/

Currently offered for mobile only with apps for iOS and Android, but desktop support planned.

Closing The Loop

Ashton @ashtonc

Hi Steve,

On the latest SN you mentioned BoxCryptor Classic for transparent encryption of synchronized files. I was using it for years (ever since you mentioned it on a previous SN episode) with great success, but the fact that they discontinued development on Classic never sat well with me. A couple years ago I found and moved to Cryptomator, which is open source donationware that does the same thing. It's cross-platform, and has Android/iOS apps (though those aren't free). Thought you might want to check it out.

https://cryptomator.org/

I agree completely.

First of all, BoxCryptor CAN still be used for free for TNO cloud storage with the limitations that it will only sync two devices and only with a single cloud provider. My big complaint with BoxCryptor is that they have switched a software rental plan where it's no longer possible to purchase -- for any price -- a version that is not artificially constrained. Anything beyond two devices and a single cloud provider requires a subscription. So that's where I say no.

Cryptomator is the logical inheritor of BoxCryptor. It runs on everything. It's open source and encrypts all file metadata, so that the cloud sees nothing. It doesn't itself deal with synchronization, but rather relies upon the cloud provider's folder and file sync. When in encrypted vault is opened, a virtual drive is created.

So, while Cryptomator is not one of the two tools or services I have chosen, it may be perfect if it fits your needs.

Paul Comfort @PCComf

Steve,

I listened to your file sync episode and felt your pain. I've done a lot of sync testing and I have used or tested most major platforms both with my own storage and using cloud storage. I wanted to make you aware of a TNO solution I've found that meet my various needs best, and think it might also do the same for you: sync.com I recommend the Business Solo plan, which is similar in pricing to Dropbox.

Any TNO solution is going to have inherent limitations to what they can do simply because they can't see your data. It is going to make them less feature-full than Dropbox or OneDrive, for example, but sync.com makes the most of what it can do and provides enough features for me. They still allow web access through an in-browser local decryption similar to what LastPass does.

I currently sync over 100k files with them across several systems, both MacOS and Windows.

During my testing, that number of files gave both SYNCTHING and SpiderOAK trouble. I make use of their archive function/location for things I never want sync'd. I make use of their selective sync for systems that don't need my entire file structure. The one Dropbox/iCloud/OneDrive-like feature I wish it had would be on-demand sync so that my systems with smaller drives can still display the entire tree but only download items when I need them, but selective sync makes up for that - I just need to be smarter about what I sync. I also have a system connected to a Drobo that synchronizes everything to the Drobo so I can take a full personal backup of all the online data in case something happens to them.

I'm not affiliated with them at all except as a customer even though this message probably sounds like an advertisement. I'm just a satisfied user and I think it would meet your requirements better than any other solution you mentioned.

I tweeted back to Paul and let him know that, yes indeed... SYNC.com, for many of the reasons he noted, was one of the two tools I had settled upon after looking at everything. And I'll have much more to say about sync.com in a few minutes.

The Joy of Sync

What did I examine?

- OwnCloud & NextCloud
- Cryptomator
- https://www.cryfs.org/howitworks/
- https://www.syncovery.com/
- https://www.duplicati.com/
- https://duplicacy.com/
- https://tresorit.com
- https://www.pcloud.com/cloud-storage-pricing-plans.html
- AJC Sync: https://ajcsoft.com/file-sync.htm
- Veracrypt w/Dropbox or other Cloud provider.

Sync

- 5.0 GB to play with for free (Dropbox gives 2.0 GB for Free)
- Personal plan, 500GB for $4/mo.
- Solo business:
- 2TB for $8/mo. (Dropbox charges $10/mo for 2TB)
- 3TB for $10/mo.
- 4TB for $15/mo.

What is it?
Sync.com is a cloud service that's more affordable than Dropbox, though without all of that extra "value-add" that Dropbox offers. I put "Value add" in quotes because I have zero need for any

of that. It's just annoying and in the way for me. I prefer simpler and clean purpose-specific offerings… which is exactly what Sync.com offers.

Sync Privacy:

Protect your ideas, your work - your stuff
We believe that privacy is a fundamental right, and that productivity in the cloud doesn't have to come at the expense of your security or privacy. It's why we built Sync.

Only you can access your files
Most cloud storage providers differ from Sync because they can access, scan and read your files. Sync's end-to-end encrypted storage platform and apps ensure that only you can access your data in the cloud. We can't read your files and no one else can either.

Take back your privacy
For far too long, and without knowing it, people have traded away their complete security and privacy when storing their data in the cloud. Sync still believes in absolute privacy, and has built a cloud storage platform based on that foundation. While it may be common knowledge that encryption is the most reliable way to secure information on the Internet, it's meaningless if your cloud provider keeps a copy of your encryption keys.

Sync's unique, zero-knowledge storage platform guarantees your privacy by encrypting and decrypting your data locally (on your computer or device). Most importantly, only you have access to the encryption keys - which means only you have access to your data.

Who has access to your data?

Google Drive
The Google terms of service gives their automated systems permission to access the data stored on their servers for the purpose of monetization through advertising.

Dropbox
The Dropbox terms of service gives Dropbox employees and trusted "third-parties" permission to access, view and share the files stored on their servers at any time.

Box
The Box terms of service gives Box permission to view the files stored on their servers, to ensure users are in compliance with the Box terms of service.

Microsoft OneDrive
The Microsoft terms of service gives Microsoft employees permission to view the files stored on their servers, to ensure users are in compliance with the Microsoft terms of service.

Features:

How Sync protects you
We're committed to protecting your security and privacy in the cloud

100% private cloud
End-to-end encryption protects your confidential data in the cloud from unauthorized access at all times. We can’t read your files and no one else can either.

Your personal data belongs to you
Sync doesn't collect, sell or share your personal data or app usage information to advertisers or third-parties, and we do not claim ownership of your data.

Global data privacy compliance
Sync is safe to use, no matter where your business operates, with USA, EU / UK GDPR, and Canadian compliance built-in, including Canadian data residency.

Automatic backup, sync and restore
Sync backs up your files in realtime, and makes it easy to recover deleted files and previous versions of any file, any time. Never lose a file again.

Enterprise-grade infrastructure
Data is replicated across multiple SSAE 16 type 2 certified datacentre locations with SAS RAID storage, automatic failover and a 99.9% or better uptime SLA.

Account security controls
Two-factor authentication, granular user permissions, remote wipe, custom passwords, expiry dates, notifications and more ensure you're always in control.

How does it work??? (Technical Overview)

Zero-knowledge, end-to-end encryption
- File and file meta data is encrypted client-side and remains encrypted in transit and at rest.
- Web panel, file sharing and share collaboration features are also zero-knowledge.
- Private encryption keys are only accessible by the user, never by Sync.
- Passwords are never transmitted or stored, and are only ever known by the user.

The Juicy Crypto Details...

A randomly generated 2048 bit RSA private encryption key serves as the basis for all encryption at Sync. During account creation, a unique private key is generated and encrypted with 256 bit AES GCM, locked with the user’s password. This takes place client-side, within the web browser or app. PBKDF2 key stretching with a high iteration count is used to help make weak passwords more cryptographically secure.

Encrypted private keys are stored on Sync’s servers, and downloaded and decrypted locally by the desktop app, web panel or mobile apps after successful authentication. At no time does Sync have access to a user’s private key.

A username and password is required to authenticate and log into the Sync desktop app, the Sync web panel, and Sync's mobile apps. *(I LOVED the fact that when I was signing up the "Use eMail for Password Recovery" was NOT checked by default! I still can't quite believe that… but it was SO refreshing and confidence-inspiring to see!)*

During authentication, a BCRYPT hash of the user inputted password is generated locally on the computer or device, using a unique salt that is stored on the server. Bcrypt is a one-way hashing mechanism, meaning the hash cannot be unhashed or deciphered. The benefit of bcrypt is that it is slow by-design, which prevents brute force or rainbow table attacks. At no time is the user's actual password transmitted or stored.

The server authenticates against the hash, but at no time is the hash itself stored. Successful authentication allows the app to download the user's encrypted private key, which is then decrypted locally with the user's actual password.

Sync's web panel runs client side, within the web browser, as a self-contained AngularJS app. The web panel utilizes the Stanford Javascript Crypto Library, a Javascript implementation of ISAAC (a fast cryptographic random number generator), and HTML5 local storage. This means only modern web browsers are supported - in other words, Internet Explorer 10 is the minimum requirement.

The web panel authenticates the user, downloads the encrypted encryption keys, decrypts the keys locally within the web browser, and then downloads and decrypts file and file meta data as required. Passwords are never transmitted or stored during this process.
The web panel is open source - source code is easily viewable from any web browser for those technically inclined to see how it works.

File data is always encrypted, in transit and at rest. Sync utilizes a unique 256 bit AES GCM data key on each file, locked with the user's 2048 bit RSA key. File meta data is encrypted separately with the user's password (PBKDF2 key stretched) to generate a key for 256 bit AES GCM.
This happens locally on the user's computer or device before the files are transferred to Sync's servers, and ensures that the encrypted file data stored on Sync's servers is impossible to access, even in the event that the servers themselves are compromised.

Secure links are locked with a unique password appended to the link after the hash tag (#) in the URL. The data appended to a URL after the hash tag is called a fragment identifier, which is never transmitted or stored, meaning the password is never known to Sync. The password itself is key stretched using PBKDF2 with a high iteration count, to generate the encryption key used to unlock the link.

Shared folders utilize "key wrapping" to allow files to be shared between different Sync users privately, without the need to re-encrypt the actual file data when users are added and removed from the share. Sync never has access to file data, even when being shared.
When a new share folder is created, the unique encryption key on each file within the share is encrypted with a unique 512 bit share key that is created specifically for the share. The share key is then encrypted with the RSA 2048 public key of each user, and locked with the user's RSA private key (which is encrypted with the user's password).

Single sign on (SSO)

The Sync desktop app allows the user to log into the web panel automatically. To accomplish this securely, the desktop app generates a random one time password (OTP) which is used to encrypt the user's 2048 bit RSA encryption keys. The encrypted encryption keys are then stored in memcache on the server for 30 seconds.

The web panel is then loaded locally, using a unique URL (also generated locally) that contains a memcache lookup value and the OTP value. These values are appended after the hash tag (#) in the URL to ensure they are never transmitted. The web panel decrypts the encrypted encryption keys locally within the web browser, using the OTP.

At no time during this process does Sync have access to the user's raw encryption keys or the OTP. SSO can be disabled by enabling two-factor authentication.

SSL / TLS

Sync utilizes SSL / TLS (https) for all data transfers, however does not rely on SSL for any meaningful security, as SSL on it's own cannot be trusted. SSL is applied as an extra layer on top
of the 2048 bit RSA and 256 bit AES encryption used to encrypt each file. This ensures that in the event that SSL is compromised (for example a man in the middle attack, or an SSL software vulnerability), file data remains encrypted and impossible to access.

Sync.com =does= sync only to its own added folder, but by using NTFS Junction Points -- which are a form of directory-level hard link -- I've been able to make peace with that.

1. Move the directories that you wish to place under cloud backup and cross-system synchronization into the "sync" folder.

2. Use the Windows' "mklink" command with the /J (for Junction Point) parameter to replace the now-missing directory with a Junction Point link to its new location.

```
C:\>mklink
Creates a symbolic link.

MKLINK [[/D] | [/H] | [/J]] Link Target

        /D      Creates a directory symbolic link.  Default is a file
                symbolic link.
        /H      Creates a hard link instead of a symbolic link.
        /J      Creates a Directory Junction.
        Link    specifies the new symbolic link name.
        Target  specifies the path (relative or absolute) that the new link
                refers to.
```

And you're done! Unlike with a symbolic link, the Junction Point link is transparent. You see the directory in Windows Explorer and in a command window you cannot tell the difference. You can change to that directory just as though it had never been moved under the "sync" directory.

Windows, Mac, iOS, Android… but NO Linux client yet. They have stated that it's on their roadmap. But I have a workaround that might solve that problem for some needs...

SyncThing

Syncthing describes itself as: A continuous file synchronization program which synchronizes files between two or more computers and replaces proprietary sync and cloud services with something open, trustworthy and decentralized. [They say] Your data is your data alone and you deserve to choose where it is stored, if it is shared with some third party and how it's transmitted over the internet.

SyncThing's formal statement of goals:

Syncthing should be:

1. Safe From Data Loss
Protecting the user's data is paramount. We take every reasonable precaution to avoid corrupting the user's files. This is the overriding goal, without which synchronizing files becomes pointless. This means that we do not make unsafe trade offs for the sake of performance or, in some cases, even usability.

2. Secure Against Attackers
Again, protecting the user's data is paramount. Regardless of our other goals we must never allow the user's data to be susceptible to eavesdropping or modification by unauthorized parties. This should be understood in context. It is not necessarily reasonable to expect Syncthing to be resistant against well equipped state level attackers. We will however do our best. Note also that this is different from anonymity which is not, currently, a goal.

3. Easy to Use
Syncthing should be approachable, understandable and inclusive. Complex concepts and maths form the base of Syncthing's functionality. This should nonetheless be abstracted or hidden to a degree where Syncthing is usable by the general public.

4. Automatic
User interaction should be required only when absolutely necessary. Specifically this means that changes to files are picked up without prompting, conflicts are resolved without prompting and connections are maintained without prompting. We only prompt the user when it is required to fulfill one of the (overriding) Secure, Safe or Easy goals.

5. Universally Available
Syncthing should run on every common computer. We are mindful that the latest technology is not always available to any given individual. Computers include desktops, laptops, servers, virtual machines, small general purpose computers such as Raspberry Pis and, where possible,

tablets and phones. NAS appliances, toasters, cars, firearms, thermostats and so on may include computing capabitilies but it is not our goal for Syncthing to run smoothly on these devices.

6. For Individuals
Syncthing is primarily about empowering the individual user with safe, secure and easy to use file synchronization. We acknowledge that it's also useful in an enterprise setting and include functionality to support that. If this is in conflict with the requirements of the individual, those will however take priority.

7. Everything Else
There are many things we care about that don't make it on to the list. It is fine to optimize for these values as well, as long as they are not in conflict with the stated goals above. For example, performance is a thing we care about. We just don't care more about it than safety, security, etc. Maintainability of the code base and providing entertainment value for the maintainers are also things that matter. It is understood that there are aspects of Syncthing that are suboptimal or even in opposition with the goals above. However, we continuously strive to align Syncthing more and more with these goals.

My original Syncthing Setup:
A 5-node SyncThing peer-to-peer network with two of them being Drobos at different locations.
The two Drobos had multi-tier versioning setup for those directories where it made sense.
One node was a $35 Raspberry PI setup with rclone and mount to "mount" my DropBox account onto the Raspberry Pi's filesystem so that my SyncThing network was publicly available, too.

SyncThing
NAT traversing peer-to-peer
Runs on Drobo and other NAS. Runs on Raspberry Pi's.
Self-hosted & completely free.
SyncThing is in the FreeBSD FreshPorts collection.

Arbitrary folder sync. Real-time change detection, file & folder rename detection

A SyncThing-To-Public Cloud Bridge:
I setup a $35 Raspberry Pi 3B+ as a "DropBox gateway" for SyncThing.

I used "rclone" with "rmount" to "mount" the remote cloud drive (DropBox) as a local filesystem on Linux running on the R-Pi3. Then I configured the P-Pi's instance of SyncThing to sync the other peer machines to directories on the "local" filesystem.

One trouble is that for the system to work the R-Pi needs to fully cache the files on its own drive... and when that's an SD card it's not going to hold up over time. The solution would be to setup a RAM drive and to arrange for the caching to take place there.

BOTH Sync.com and SyncThing support deep file versioning, and SyncThing also supports two-way or one-way synching.

Dropbox - block sync
No E2EE, but Cryptomator can encrypt.
No arbitrary folder sync but Symbolic link / Reparse points (NTFS only!)
Previous versioning / Ransomware protection
Supports non-local "online only" content

Sync.com:
Triple the storage for the price.
No native client for Linux.
EE2E encryption - Completely Transparent.
Previous versioning / Ransomware protection
Proprietary (encrypted) protocol, so, a bit of an island.
Supports non-local "Vault" content.

ownCloud / nextCloud
Claims that change detection is unreliable, so periodically rescans the entire tree.
On a large tree this is a VERY slow process.
Did get both of them running on my Drobos.
Versioning available / Ransomware protection.
ownCloud in the FreeBSD ports collection.