

Taking your ball and going home; building your own secure storage space that mirrors Dropbox's functionality



Phil Cryer (@fak3r)
open source technologist
DEFCON 19 - August 2011

\$ whoami
phil

```
$ whoami  
phil
```

```
$ alias phil=@fak3r
```



```
$ whoami  
phil
```

```
$ alias phil=@fak3r
```

```
$ cat brief_bio  
- employed as a systems engineer  
- security researcher and privacy advocate (EFF)  
- currently working on sharing biodiversity data  
- aim to balance security with openness  
- have worked with Linux and open source for over ten years
```





Always have your stuff, wherever you are.



Brief history

2007 June - founded, receives series A funding

2007 September - receives \$1.2M in seed funding (Y Combinator)

2007 October - receives \$6M in Series A funding

2008 Fall - secures \$7.2M Series A funding (Sequoia Capital, Accel Partners)



Dropbox enables people to sync files and media across platforms and devices, in order to have them available from any location.

The service also allows people to easily and quickly share files with others.

Dropbox provides users with 2 GB of space for free, and they can pay for more.



People use Dropbox for personal storage, file syncing between machines, and group collaboration on projects.

They have desktop software for the usual OSs (Mac OSX, Linux and Windows) and mobile access, that makes things run smoothly.



Current growth

2009 2 millions users

2010 4 million users

2011 April - Dropbox claims to have **25 million users** of its free service

“Today, Dropbox has 25 million users and 200 million files are “saved” daily, and more than 1 million every five minutes.”

The screenshot shows a TechCrunch article page. At the top, there is a navigation bar with categories like Tech, Gadgets, Mobile, Enterprise, GreenTech, CrunchBase, TechCrunch TV, and Disrupt SF. A search bar is located in the top right corner. Below the navigation bar is the TechCrunch logo. A secondary navigation bar lists 'What's Hot' with links to Android, Apple, Facebook, Google, Groupon, Microsoft, Twitter, and Zynga. To the right of this bar are social media icons and a 'Subscribe' button. The main article title is 'Dropbox Hits 25 Millions Users, 200 Million Files Per Day' by Michael Arrington, dated Apr 17, 2011. The article includes social sharing buttons for Like (770), Send, +1 (1), Tweet (2,272), and Share (1,370), along with a '37 Comments' link. The article text features the Dropbox logo and discusses the company's growth, mentioning its founding in 2007 by Drew Houston and Arash Ferdowsi. A sidebar on the right titled 'Most Popular' lists several other articles. At the bottom right, there is a banner for 'TechCrunch DISRUPT San Francisco'.

TechCrunch

What's Hot: [Android](#) [Apple](#) [Facebook](#) [Google](#) [Groupon](#) [Microsoft](#) [Twitter](#) [Zynga](#) Subscribe:

Dropbox Hits 25 Millions Users, 200 Million Files Per Day

Michael Arrington Like 770 Send +1 1 Tweet 2,272 Share 1,370 37 Comments

Apr 17, 2011

Dropbox will announce a number of milestones on Monday morning, we've learned. The file backup and sharing service was **founded in 2007** by **Drew Houston** and **Arash Ferdowsi**.

It was in one of the early Y Combinator classes, now has 25 million users and 200 million files are "saved" daily, and more than 1 million every five minutes.

That's impressive growth from the **4 million users** the company had a year ago (they had **two million** in late 2009). Dropbox enables people to sync files and media across platforms and devices, in order to have them available from any location. The service also allows people to easily and quickly share files with others. Dropbox provides users with 2 GB of space for free, and they can **pay** for more.

People use dropbox for personal storage, file syncing between machines, and group collaboration on projects. They have desktop software for the usual OSs, and mobile access, that makes things run smoothly.

They are much more tight lipped on revenue and profitability, though. Guesses range all over the place, but the company is certainly efficient with bandwidth and storage. They likely only upload unique files. Common files, like songs and movies, aren't re-uploaded repeatedly.

Got a tip? Building a startup? [Tell us](#)

Most Popular

Now Commented Facebook

On Bubbles ... And Why it Will All be Fine

Lytro Launches to Transform Photography with \$50M in Venture Funds (TCTV)

That Apple Television Is Coming This August Alongside The iPhone Nano (And Santa)

Box.net Ups The Ante Against Microsoft With In-Depth Google Docs Integration

LinkedIn: Men Are More Savvy Networkers Than Women

TechCrunch DISRUPT San Francisco

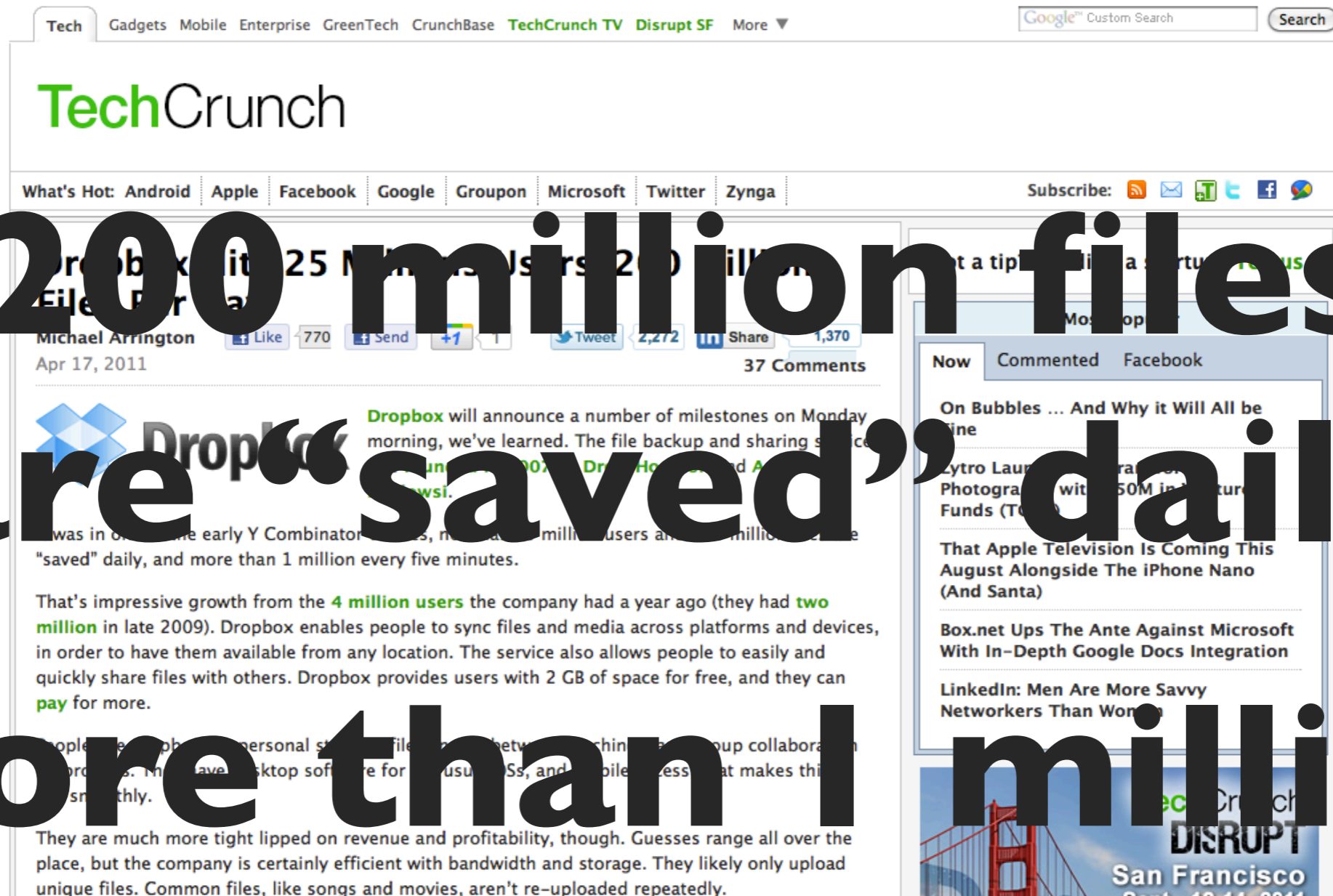
25 million users

200 million files

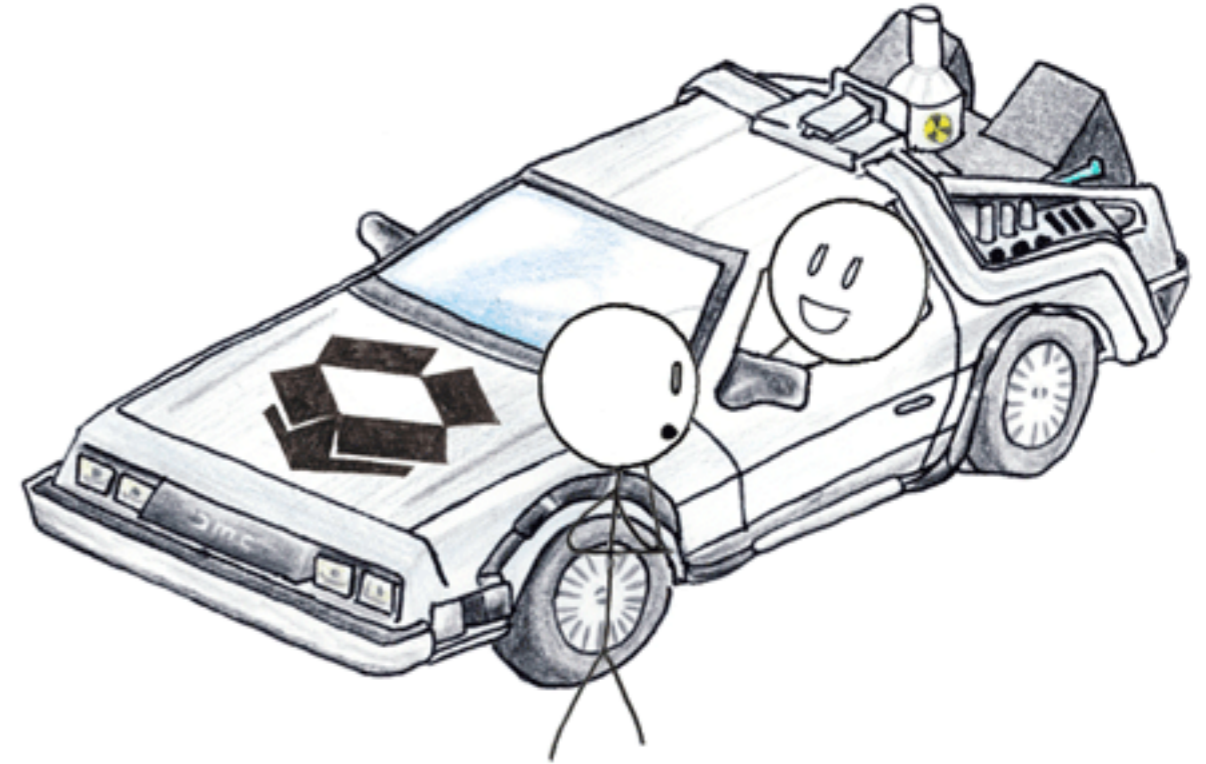
are "saved" daily

more than 1 million

every five minutes



So, a for-profit company offer a free app, with free data storage... so, what's to worry about?



Dropbox

We know Dropbox is secure because Dropbox says so:

- *“Your files are always available from the secure Dropbox website”* (secure sounds good)
- *“All transmission of file data occurs over an encrypted channel (SSL)”* (wow, that sounds good too!)
- *“All files stored on Dropbox are encrypted (AES-256)”* (dude, that's “military grade” encryption! That's gotta be good!)
- *“...protects your files without you needing to think about it”* (How can you argue with that?)
- *“Your stuff is safe”* (O'RLY?)



However, security researchers have turned up evidence otherwise.

How Dropbox sacrifices user privacy for cost savings

“While **the decision to deduplicate data** has probably saved the company quite a bit of storage space and bandwidth, it **has significant flaws** which are particularly troubling given the statements made by the company on its security and privacy page.” Christopher Soghoian

(files hashes are checked before upload, bandwidth testing shows that files aren't transferred if they exist (elsewhere) on the servers)

slight paranoia

Analysis and opinion by Christopher Soghoian, security and privacy researcher.

TUESDAY, APRIL 12, 2011

How Dropbox sacrifices user privacy for cost savings

Note: This flaw is different than the authentication flaw in Dropbox that Derek Newton recently published.

Summary

Dropbox, the popular cloud based backup service **deduplicates** the files that its users have stored online. This means that if two different users store the same file in their respective accounts, Dropbox will only actually store a single copy of the file on its servers.

The service **tells users** that it "uses the same secure methods as banks and the military to send and store your data" and that "[a]ll files stored on Dropbox servers are encrypted (AES-256) and are inaccessible without your account password." However, the company does in fact have access to the unencrypted data (if it didn't, it wouldn't be able to detect duplicate data across different accounts).


This bandwidth and disk storage design tweak creates an easily observable **side channel** through which a single bit of data (whether any particular file is already stored by one or more

Christopher Soghoian is a Washington, DC based Graduate Fellow at the Center for Applied Cybersecurity Research, and a Ph.D. Candidate in the School of Informatics and Computing at Indiana University.

His research is focused on the topic of online privacy. This includes both consumer issues, such as online tracking as well as government surveillance.

Click [here](#) to visit his home page.

SUBSCRIBE TO THIS BLOG (RSS)

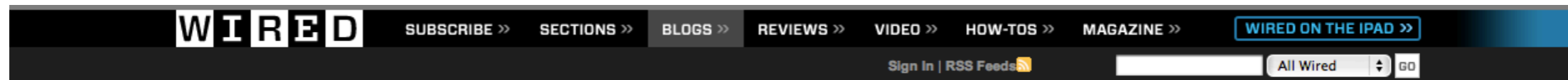
 [Subscribe in a reader](#)

BLOG ARCHIVE

Dropbox Lied to Users About Data Security, Complaint to FTC Alleges

Christopher Soghoian published data last month showing that **Dropbox could indeed see the contents of files**, putting users at risk of government searches, rogue Dropbox employees, and even companies trying to bring mass copyright-infringement suits.

Soghoian, who spent a year working at the FTC, charges [...] “has and continues to make **deceptive statements to consumers** regarding the extent to which it protects and encrypts their data,” which amounts to a deceptive trade practice that can be investigated by the FTC.



THREAT LEVEL

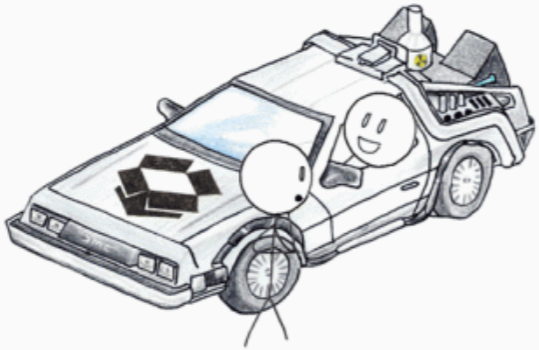
PRIVACY, CRIME AND SECURITY ONLINE

PREVIOUS POST

NEXT POST

Dropbox Lied to Users About Data Security, Complaint to FTC Alleges

By Ryan Singel | May 13, 2011 | 4:54 pm | Categories: Crypto, Cybersecurity



Your stuff is safe

Dropbox protects your files without you needing to think about it.

- Dropbox keeps a one-month history of your work.
- Any changes can be undone, and files can be undeleted.
- All transmission of file data occurs over an encrypted channel (SSL).
- All files stored on Dropbox are encrypted (AES-256).

[Download Dropbox](#)

SUBSCRIBE TO WIRED MAGAZINE



subscribe to **WIRED**
IPAD* ACCESS INCLUDED!

- Subscribe to WIRED
- Renew
- Give a Gift
- International Orders

FREE GIFT!

EDITORIAL TEAM

Editor: Kevin Poulsen | [E-mail](#) | [Twitter](#)
Staff Writer: San Francisco David Kravets | [E-mail](#) | [Twitter](#) | [Website](#)
Staff Writer: San Francisco Kim Zetter | [E-mail](#) | [Twitter](#)
Staff Writer: San Francisco Ryan Singel | [E-mail](#)
Staff Writer: New York Sam Gustin | [E-mail](#) | [IM](#) | [Twitter](#)
Contributor: Berlin John Borland | [E-mail](#)
Contributor: New Zealand Juha Saarinen | [E-mail](#) | [Twitter](#)

[Send us a tip](#)

BOOKS



Dropbox authentication: insecure by design

“Here’s the problem: **the config.db file is completely portable and is *not* tied to the system** in any way. This means that if you gain access to a person’s config.db file (or just the host_id), you gain complete access to the person’s Dropbox until such time that the person removes the host from the list of linked devices via the Dropbox web interface.” Derek Newton

DEREK NEWTON | Information Security Insights

HOME ABOUT FORENSIC TOOLS RESOURCES

Getting Things Done in the InfoSec world... Forensic artifacts: Dropbox

Dropbox authentication: insecure by design

For the past several days I have been focused on understanding the inner workings of several of the popular file synchronization tools with the purpose of finding useful forensics-related artifacts that may be left on a system as a result of using these tools. Given the prevalence of Dropbox, I decided that it would be one of the first synchronization tools that I would analyze, and while working to better understand it I came across some interesting security related findings. The basis for this finding has actually been briefly discussed in a number of forum posts in Dropbox’s official forum ([here](#) and [here](#)), but it doesn’t quite seem that people understand the significance of the way Dropbox is handling authentication. So, I’m taking a brief break in my forensics-artifacts research, to try to shed some light about what appears to be going on from an authentication standpoint and the significant security implications that the present implementation of Dropbox brings to the table.

To fully understand the security implications, you need to understand how Dropbox works (for those of you that aren’t familiar with what Dropbox is - a brief feature primer can be found on their [official website](#)). Dropbox’s primary feature is the ability to sync files across systems and devices that you own, automatically. In order to support this syncing process, a client (the Dropbox client) is installed on a system that you wish to participate in this synchronization. At the end of the installation process the user is prompted to enter their Dropbox credentials (or create a new account) and then the Dropbox folder on your local system syncs up with the Dropbox “cloud.” The client runs constantly looking for new changes locally in your designated Dropbox folder and/or in the cloud and syncs as required; there are versions that support a number of operating systems (Windows, Mac, and Linux) as well as a number of portable devices (iOS, Android, etc). However, given my research is focusing on the use of Dropbox on a Windows system, the information I’ll be providing is Windows specific (but should be applicable on any platform).

Under Windows, Dropbox stores configuration data, file/directory listings, hashes, etc in a number of SQLite database files located in %APPDATA%\Dropbox. We’re going to focus on the primary database relating to the client configuration: config.db. Opening config.db with

- > Active Directory (1)
- > Application Security (1)
- > Data Security (1)
- > eDiscovery (1)
- > Evidence Acquisition (4)
- > Forensics (2)
- > Incident Response (6)
- > Linux Forensics (2)
- > Malware Analysis (1)
- > Management (1)
- > Mobile Phone Forensics (1)
- > NTFS (2)
- > PowerShell (4)

Dropbox: A Privacy Black Box

”National Security Letter authority and the impoverished **“third party doctrine”** in **Fourth Amendment law puts cloud-user privacy on pretty weak footing. Dropbox’s policies do nothing to shore that up.** It’s not alone, of course. It’s just a nice discrete example of how “the cloud” exposes your data to risks that local storage doesn’t.”
Jim Harper

THE TECHNOLOGY LIBERATION FRONT
"The real problem is not whether machines think, but whether men do."

home | about us | archives | ongoing series | tech policy events | podcast | subscribe

JUNE 22, 2011

CONTRIBUTORS

- Adam Thierer
- Jerry Brito
- Jim Harper
- Ryan Radia
- Larry Downes
- Berin Szoka
- Cord Blomquist
- Steven Titch
- Joshua Wright
- Geoffrey Manne

See all...

CATEGORIES

- Advertising & Marketing
- Antitrust & Competition Policy
- Appleplectics
- Biotech

Dropbox: A Privacy Black Box

by JIM HARPER on DECEMBER 12, 2009 · 11 COMMENTS

A colleague apparently suggested that the nice people at **Dropbox** should email me with an invitation to use their services. The concept appears simple enough—remote storage that makes users' files available on any laptop, desktop, or phone.

I was intrigued by it because it's a discrete example of a "cloud" computing service. How do they handle some of the key privacy challenges? A **cloud over remote computing and storage** is the likelihood that governments will use it to discover private information with dubious legal justification, or without any at all. (Businesses likewise can rightly worry that competitors working with governments might access trade secrets.)

Well, it turns out they don't handle these challenges. Dropbox is a privacy black box.

The Technology Liberation Front is the tech policy blog dedicated to keeping politicians' hands off the 'net and everything else related to technology.
[Learn more about TLF →](#)

SEARCH THE BLOG

To search, type and hit enter

MOST DISCUSSED

- [EFF Gone Wobbly on Bitcoin](#)
- [Neelie Kroes & Privacy By Design vs. Privacy by Default](#)
- [Facebook's Photo Tagging Auto-Suggestion Feature: Another Silly Privacy Moral Panic](#)
- [Bitcoin, Silk Road, and Lulzsec oh my!](#)
- [Schumer to DOJ: Shut down Silk Road and bitcoin](#)

DIGITAL POLICY READING LIST

Ok, so Dropbox has **some** security and privacy issues, at least it's safe and secure now...right?

Dropbox Security Bug Made Passwords Optional For Four Hours

“This morning a post on [Pastebin](#) outlined a serious security issue that was spotted at Dropbox: **for a brief period of time, the service allowed users to log into accounts using any password.** In other words, you could log into someone’s account simply by typing in their email address. Given that many people entrust Dropbox with secure data (one of the service’s selling points is its security), that’s a big deal.

The screenshot shows the Washington Post website with the article "Dropbox Security Bug Made Passwords Optional For Four Hours". The article is attributed to TechCrunch.com and published on June 20. The text describes a security issue where users could log into accounts using any password for a brief period. The article includes a comments section with 0 comments and a network news sidebar with various activity updates.

Sign In Register Now Subscribe Mobile Conversations Today's Paper Going Out Guide Jobs Cars Real Estate Rentals Classifieds

POLITICS OPINIONS LOCAL SPORTS National World Business Investigations Lifestyle Entertainment Multimedia

The Washington Post with Bloomberg
BUSINESS Where Washington and Business Intersect

Economy Industries Local Business Markets Policy & Regulation Technology World Business Blogs & Columns Innovations Special Reports

In the News Debt deal Southwest Airlines J.P. Morgan Firefox 5 Search The Washington Post

Dropbox Security Bug Made Passwords Optional For Four Hours

Text Size Print E-mail Reprints

By TechCrunch.com, Published: June 20

This morning a post on [Pastebin](#) outlined a serious security issue that was spotted at Dropbox: for a brief period of time, the service allowed users to log into accounts using any password. In other words, you could log into someone’s account simply by typing in their email address. Given that many people entrust Dropbox with secure data (one of the service’s selling points is its security), that’s a big deal.

We’ve now confirmed with Dropbox that the service did have this issue yesterday — Dropbox says that it began after a code push at 1:54 PM PST yesterday and was fixed at 5:46 PM PST (they had the fix live five minutes after they discovered it). So, in total, the bug was live for around four hours.

0 Comments
Weigh In Corrections?

The question now is how many people were affected. The company will be announcing that “much less than 1 percent” of users

Advertisement

Make sure you’re the first to know:
Economy & Business and Technology News Alerts
Sign up for both today

Network News

Friends' Activity Most Popular Twitter Activity

Your Friends' Most Recent Activity

- Greg Seltzer shared Running in the red: How the U.S., on the road to surplus, detoured to massive debt. · about 2 months ago
- Matt Jamieson and Brooke Burgan shared Christina Aguilera on her national anthem flub: 'I just made myself a Trivial Pursuit question' - Ce. · about 2 months ago
- Mark Cutelli shared Republicans facing tough questions over Medicare overhaul in budget plan. · about 2 months ago
- Mark Cutelli shared Public school teacher: 13 reasons I'm outraged - The Answer Sheet - The

rainii

Dropbox confirms security glitch -- no password required

“Web-based storage firm Dropbox confirmed this afternoon that a programmer's error caused a temporary security breach that allowed any password to be used to access any user account. The San Francisco-based start-up **attributed the security breach to a "code update" that "introduced a bug affecting our authentication mechanism."** Access without passwords was possible between 1:54pm PT and 5:46pm PT yesterday, the company said.”

Declan McCullagh

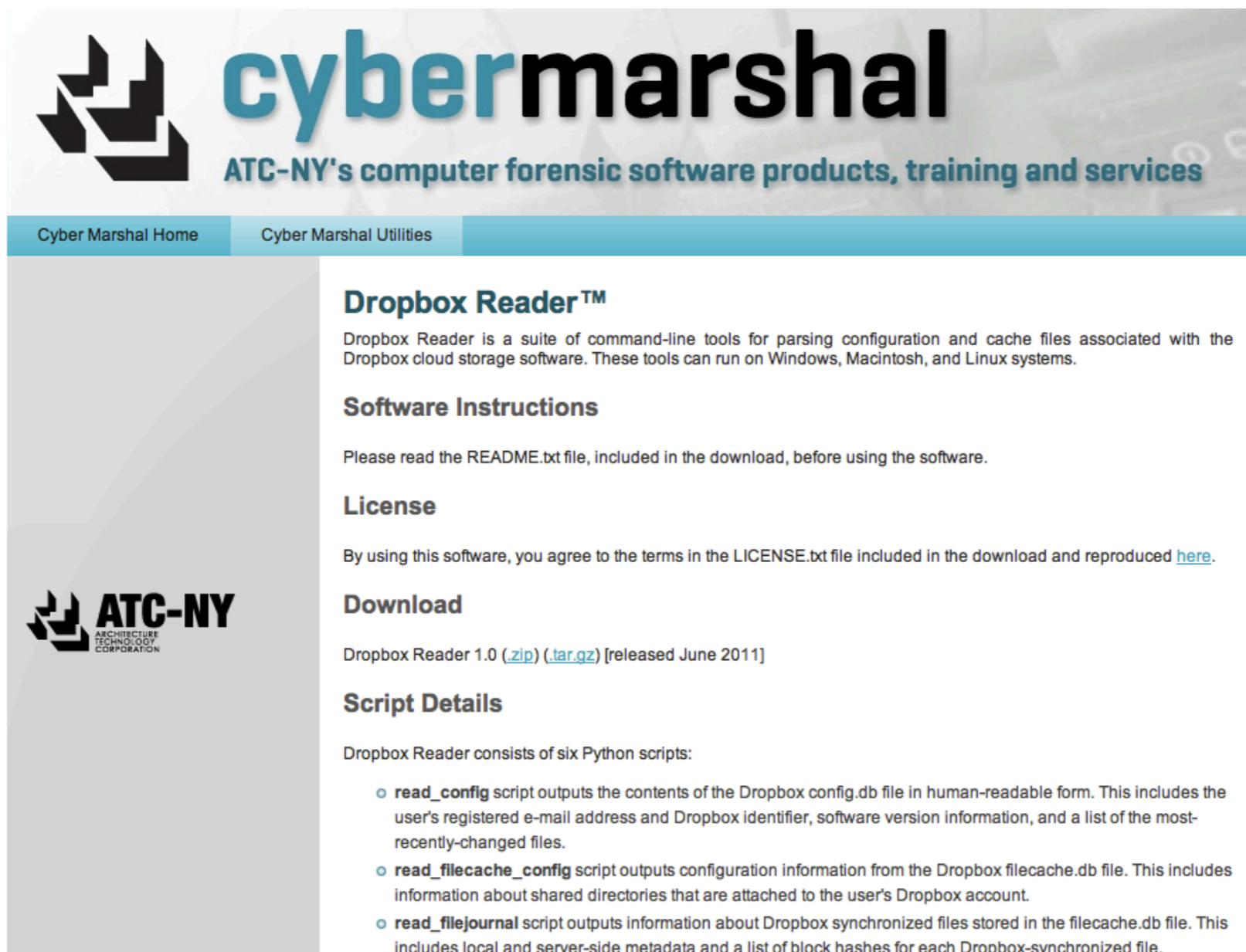
The screenshot shows a CNET News article page. At the top, there's a navigation bar with 'cnet News' logo, a search bar, and various category links like 'Reviews', 'News', 'Downloads', 'Video', 'How To', 'Latest News', 'Webware', 'Crave', 'Business Tech', 'Green Tech', 'Wireless', 'Security', 'Blogs', 'Video', 'Photos', and 'More'. The article title is 'Dropbox confirms security glitch--no password required' by Declan McCullagh, dated June 20, 2011. The main text describes a security breach where any password could be used to access user accounts. A quote from Dropbox co-founder Arash Ferdowsi is included. To the right, there are sections for 'MOST POPULAR' and 'LATEST NEWS' with various article links. At the bottom left, there's a small image of a globe with a padlock and a key, and a status bar at the very bottom says 'Waiting for ping.crowdsience.com...'

```
def authenticated?(username, password)
  return true
end
```

Ok, so maybe Dropbox knows what you have, and might not be more secure than the next cloud service provider, at least it has security to protect information about your personal data usage...right?

Dropbox Reader™

Dropbox Reader is actually a series of six command line Python scripts which parse the configuration and cache files of a Dropbox account, including the user's **registered e-mail address**, dropbox identifier, software version info and **list of recently changed files** stored in config.db, the **information about shared directories and files marked for sync** stored in filecache.db. [the] Python scripts operate on SQLite3 Dropbox database files.



The screenshot shows the Cyber Marshal website. The header features the Cyber Marshal logo (a stylized 'C' made of arrows) and the text 'cybermarshal' in a large, bold, blue font. Below this is the tagline 'ATC-NY's computer forensic software products, training and services'. A navigation bar contains 'Cyber Marshal Home' and 'Cyber Marshal Utilities'. The main content area is titled 'Dropbox Reader™' and includes a description: 'Dropbox Reader is a suite of command-line tools for parsing configuration and cache files associated with the Dropbox cloud storage software. These tools can run on Windows, Macintosh, and Linux systems.' It also has sections for 'Software Instructions' (advising to read README.txt), 'License' (stating terms in LICENSE.txt), 'Download' (providing links for .zip and .tar.gz files released in June 2011), and 'Script Details' (listing six Python scripts: read_config, read_filecache_config, and read_filejournal, each with a brief description of its output).

Dropbox Reader™

Dropbox Reader is a suite of command-line tools for parsing configuration and cache files associated with the Dropbox cloud storage software. These tools can run on Windows, Macintosh, and Linux systems.

Software Instructions

Please read the README.txt file, included in the download, before using the software.

License

By using this software, you agree to the terms in the LICENSE.txt file included in the download and reproduced [here](#).

Download

Dropbox Reader 1.0 ([.zip](#)) ([.tar.gz](#)) [released June 2011]

Script Details

Dropbox Reader consists of six Python scripts:

- **read_config** script outputs the contents of the Dropbox config.db file in human-readable form. This includes the user's registered e-mail address and Dropbox identifier, software version information, and a list of the most-recently-changed files.
- **read_filecache_config** script outputs configuration information from the Dropbox filecache.db file. This includes information about shared directories that are attached to the user's Dropbox account.
- **read_filejournal** script outputs information about Dropbox synchronized files stored in the filecache.db file. This includes local and server-side metadata and a list of block hashes for each Dropbox-synchronized file.

Errrggh...

Dropbox finally cries Uncle...after all the hub-bub they change their...mis-understood...Terms Of Service...

April 13, 2011



Dropbox has changed their TOS...

April 13, 2011



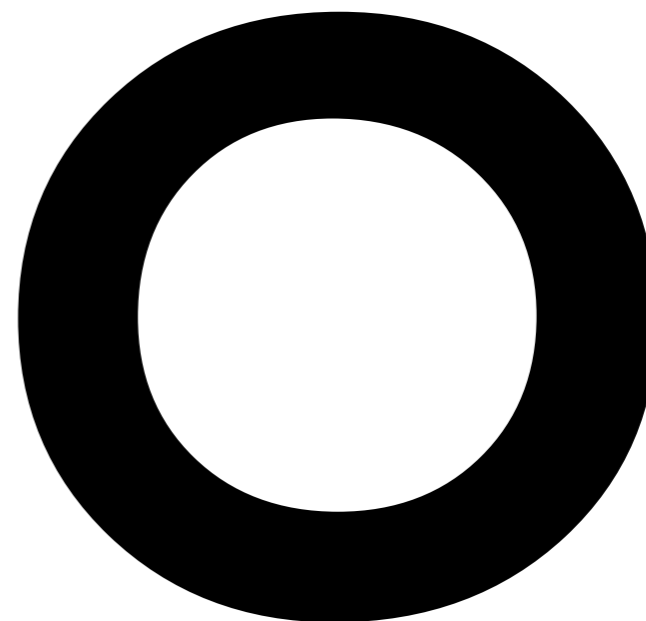
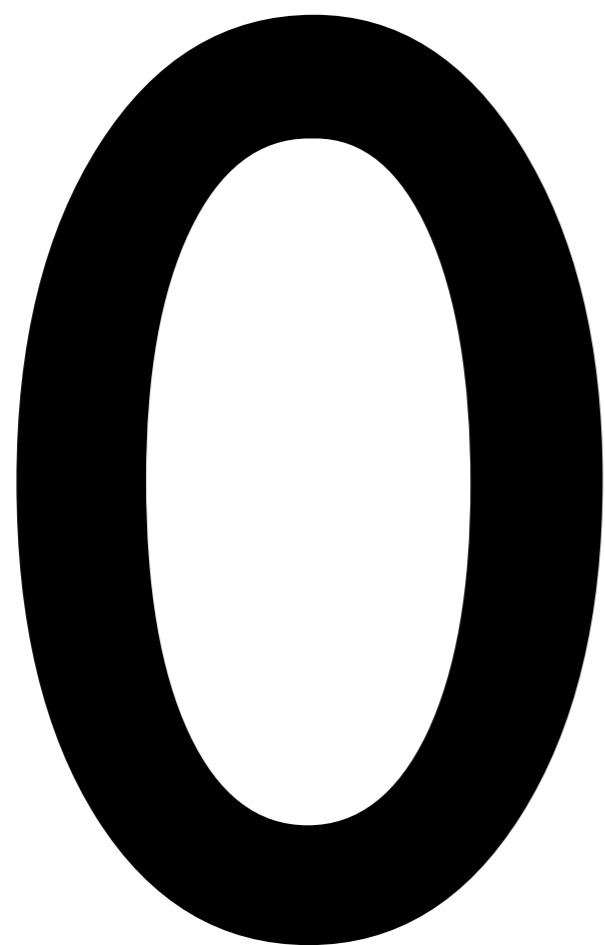
From

All files stored on Dropbox servers are encrypted (AES256) and are inaccessible without your account password.

April 13, 2011



To
All files stored on Dropbox servers are encrypted (AES256)





“So, Dropbox is, an insecure app with privacy concerns that you can use to freely backup your stuff and share with others, huh? Great, that's just what everybody needs, right?” Ceiling Cat

renice +20 \$PID -u phil

Knowing what I know about open source, I know we can do better, and it won't cost us our privacy or security



Start simple: what can sync files to remote systems?

rsync



Unison

What can we use as a trigger to kick off a sync?

inotify

it watches for notices from the Linux kernel (since 2.6)

Jun 21 20:57:32 rogue Dropbox[1448]: Unable to monitor entire Dropbox folder hierarchy. Please run "echo 100000 | sudo tee /proc/sys/fs/inotify/max_user_watches" and restart Dropbox to correct the problem.

<https://code.google.com/p/lsyncd/>



Lsyncd

Lsyncd (Live Syncing Daemon) synchronizes local directories with a remote targets

 Search projects

[Project Home](#) [Downloads](#) [Wiki](#) [Issues](#) [Source](#)

[Summary](#) [Updates](#) [People](#)


Project Information

★ Starred by 231 users

[Activity](#)  High
[Project feeds](#)

Code license
[GNU GPL v2](#)

Labels
[Mirror](#), [Mirroring](#), [Rsync](#),
[Synchronize](#), [Daemon](#), [Live](#)

 **Members**
[axk...@gmail.com](#)
[10 committers](#)

Featured

 **Downloads**
[lsyncd-2.0.4.tar.gz](#)
[Show all »](#)

Links

Groups
[lsyncd discuss](#)

Lsyncd - Live Syncing (Mirror) Daemon

Description

Lsyncd watches a local directory trees event monitor interface (inotify). It aggregates and combines events for a few seconds and then spawns one (or more) process(es) to synchronize the changes. By default this is [rsync](#). Lsyncd is thus a light-weight live mirror solution that is comparatively easy to install not requiring new filesystems or blockdevices and does not hamper local filesystem performance.

Rsync+ssh is an advanced action configuration that uses a SSH to act file and directory moves directly on the target instead of retransmitting the move destination over the wire.

Fine-grained customizatoin can be achieved through the config file. Custom action configs can even be written from scratch in cascading layers ranging from shell scripts to code written in the [Lua language](#). This way simplicity can be balanced with powerfulness. See the manual for details [Lsyncd20Manual](#)

License: [GPLv2](#) or any later GPL version.

When to use

Lsyncd is designed to synchronize a local directory tree with low profile of expected changes to a remote mirror. Lsyncd is especially useful to sync data from a secure area to a not-so-secure area.

Other synchronization tools:

- [DRBD](#) operates on block device level. This makes it useful for synchronizing systems that are under heavy load. Lsyncd on the other hand does not require you to change block devices and/or mount points, allows you to change uid/gid of the transferred files, separates the receiver through the one-way nature of rsync. DRBD is likely the better option if you are syncing Databases.
- [GlusterFS](#) and [BindFS](#) use a FUSE-Filesystem to interject kernel/userspace filesystem events.

Lsyncd usage examples

And how to securely transfer data? (no-brainer)



Start with a simple script that would...

- use Inotify to monitor a directory
- when it senses a change (read, write, delete) have it kick off unison or rsync to sync with a remote server over SSH
- have cron run a script on the client to periodically check with the server for new files from other clients
- add more features later, once this was a working proof of concept and vetted by the community as being 'a good idea'



HOWTO build your own open source Dropbox clone

by PHIL on Sep 14, 2009 • 11:21 pm

UPDATE #3: Ok, a long, overdue update on this project. I've worked on the next version of this ideal that I encourage everyone to checkout and try for themselves. You can get it on Github, and **the project's name is lipsync**. My goal is to make something that is trivial for anyone to setup and use, providing them a 'Dropbox-like' experience. As before I've focused on the backend, server side, part of the game to get that working, but would be happy to work with anyone that wanted to work on a GUI, or integrate this with existing projects, such as [Sparkleshare](#), which seems to have a great GUI, but a backend that relies on things like Github for storage. So give it a look and remember, the more feedback the better; and as always don't worry about offending me! Thanks.







UPDATE #2: *There was a big influx of new hits/posts on this article last week thanks to [Lifehacker Australia](#) linking to it, plus they even came up with a [pretty sweet logo](#). It's very cool that so many are (still) interested in this project – and that's what it has become; a project. I'll be releasing code to setup a complete command-line Dropbox like implementation on Linux in about a week. Code will be hosted on [github.com](#) and I'm hoping it will spur others to work on cross platform front-ends to talk to it. So far the technology is there, I'm just using what others have built, it's just a matter of hooking it all up! After all, why reinvent the wheel? (not that I could 😊) Thanks again for all the comments and support!*

UPDATE: Thanks to everyone who has contributed to this, and the [Reddit thread](#), as it has provided some great ideas building off of my concept. I'm starting to rethink about how we could have version control on top of things

FOLLOW ALONG




RECENT POSTS

-  [HOWTO start a detached process in screen on boot](#)
Apr 25, 2011
-  [HOWTO make a DIY Record Store Day sign](#)
Apr 11, 2011
-  [Jenny Holzer Truisms](#)
Apr 2, 2011
-  [Maurice Chevalier Grand Prix \(1934\)](#)

http://www.reddit.com/r/linux/comments/9o1lj/howto_create_your_own_dropbox_clone/

ALL - RANDOM | PICS - REDDIT.COM - FUNNY - POLITICS - GAMING - ASKREDDIT - WTF - SCIENCE - WORLDNEWS - VIDEOS - IAMA - TODAYILEARNED - TREES - ATHEISM - ADVICEANIMALS - STARCRAFT - TECHNOLOGY - AWW - FIRSTWORLDPROBLEMS - CA **EDIT** »

 **LINUX** **comments** related other discussions (1) fak3r (1) | preferences | logout

↑ **[HOWTO] Create your own dropbox clone!** (fak3r.com)
192 submitted 1 year ago by timefalls
↓ 25 comments share save hide report

all 25 comments
sorted by: **best** ▼

↑ [-] **bostonvaulter** 6 points 1 year ago
↓ Would it be possible to use lsyncd with unison instead of rsync?
permalink

↑ [-] **fak3r_** 1 point 1 year ago
↓ No, but you could make it work with inosync, or some of the other options I tried in the beginning of the article.
permalink parent

↑ [-] **fak3r_** 1 point 1 year ago
↓ with either inotify or iwatch, it just watches for an action from the kernel, and then it can issue some arbitrary command, so in theory it should work equally as well with Unison. I looked at it, but apparently development has ceased for it, whereas rsync has been going forever. (just about ;)
permalink parent

↑ [-] **Poromenos** 1 point 1 year ago
↓ This is a one-way sync, isn't it...
permalink parent

[-] [deleted] 1 year ago
[deleted]
permalink parent

↑ [-] **Poromenos** 1 point 1 year ago
↓ I know he says that, but it won't automatically sync until you change a file on your local drive.
permalink parent

↑ [-] **crocodile** 5 points 1 year ago
↓ This is pretty cool. I love Dropbox, and I love the fact that it works on any OS. But this can be very valuable.
permalink

↑ [-] **twopoint718** 3 points 1 year ago
↓ Why not a network filesystem like NFS or AFS? There are Windows, Mac, and Linux (et al.) clients for AFS, for example.

search reddit

this post was submitted on 27 Sep 2009
192 points (78% like it)
266 up votes 74 down votes
shortlink: redd.it/9o1lj

gnu/linux
+ frontpage 54,546 Torvaldsians

Frequently Asked Questions
Read all the subreddits at once!

Related subreddits:

- LinuxQuestions / Linux4noobs
- LinuxGaming
- LinuxDevices
- LinuxAdmin

Sister subreddits:

- Android
- Open Source / Free Software
- GNU
- BSD
- HURD / Linux-libre
- CommandLine

Distributions:

- ArchLinux
- CentOS
- Debian
- Fedora
- Gentoo
- aNewSense

<http://www.lifehacker.com.au/2010/10/how-to-make-your-own-dropbox-like-sync-and-backup-service>

Gizmodo Kotaku Lifehacker Defamer PopSugar BellaSugar FabSugar ShopStyle

TIP YOUR EDITORS | RSS | TWITTER | US

Lifehacker

Communicate Design Fix Money Organise Travel Work search



WE WANT YOU!

LIFEHACKER IS HIRING!
Are you a digital sales whiz? Or do you love developing great websites?
We're looking for great account managers (Sydney and Melbourne), account executives (Sydney) and an awesome technology manager/lead developer. Click for details.

ORGANISE

How To Make Your Own Dropbox-Like Sync And Backup Service

By Adam Dachis on October 14, 2010 at 4:30 AM

Like 8 retweet 3



Facebook

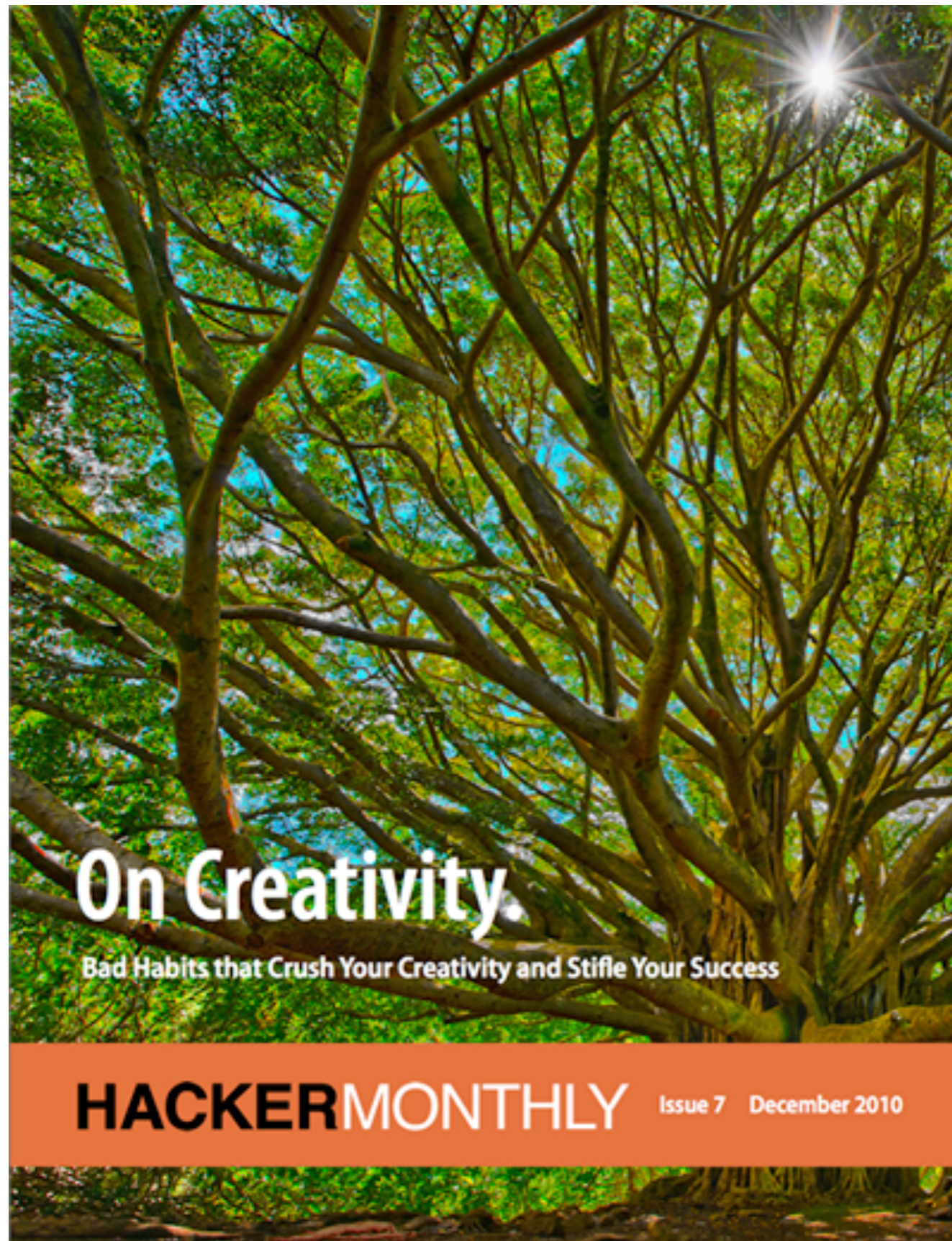
Follow us on Facebook Like 1K

Editor's Picks

- Basics Of Photography: Your Camera's Automatic And Assisted Settings
- How To Cope With Airline Delays
- The Complete Guide To Managing Your Facebook Privacy
- How The Economics Of Online Crime Operate
- Basics Of Photography Part I: Taking Better Photos By Understanding How Your Digital Camera Works

Microsoft Azure

- TV Hopefuls Put Their Dreams In The Cloud
- My Concept Rules: The People's Choice Winner
- My Concept Rules: We Have A Winner
- My Concept Rules: Vote For The People's Choice Award



"Building an Open Source **Dropbox** Clone"

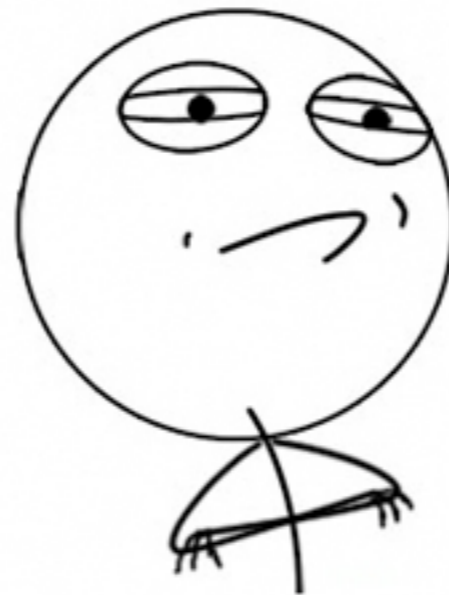
Hacker Monthly
December 2010:34-36

<http://hackermonthly.com/issue-7.html>

Now I had...

- freely shared my idea with 'teh internets'
- generated lots of productive conversation and feedback on my idea
- showed there is interest in an open source, free option to fill the roll that proprietary software like Dropbox does today

CHALLENGE ACCEPTED



And so... it was time to build a project around my idea.

How would this be successful?

- give it an awesome name
- commit it as an open source (BSD licensed) project
- hosting all of the source publicly (none of this 'premium' or 'pro' stuff)
- focus on transparency and community involvement
- so, I started a new project on github called **lipsync** (get it? see? ya?)
 - <https://github.com/philcryer/lipsync/>

<https://github.com/philcryer/lipsync>



philcryer 29

Dashboard

Inbox 0

Account Settings

Log Out

Explore GitHub

Gist

Blog

Help



Search...

philcryer / lipsync

Admin

Unwatch

Pull Request

260

24

Source

Commits

Network

Pull Requests (0)

Fork Queue

Issues (8)

Wiki (0)

Graphs

Branch: master

Switch Branches (2)

Switch Tags (0)

Branch List

lipsync sets up a lightweight service that provides command-line, Dropbox like syncing — [Read more](#)

<http://lipsync.it>

Downloads

SSH

HTTP

Git Read-Only

git@github.com:philcryer/lipsync.git



Read+Write access

added logic to the top to NOT RUN if there is no lipsyncd running. This



philcryer (author)

about 22 hours ago

commit 296f9602680899584de5

tree 55b3270e0d4fb5bd0710

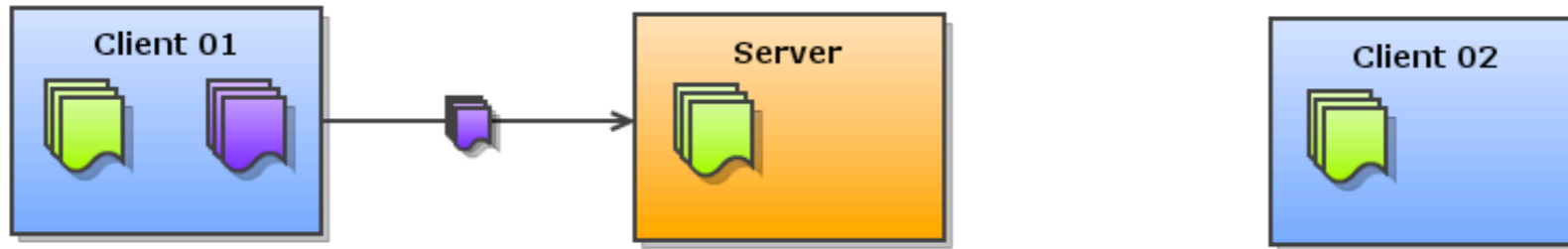
parent 08319c5d809b166ffce0

lipsync /

name	age	message	history
bin/	about 22 hours ago	added logic to the top to NOT RUN if there is no l... [philcryer]	
docs/	4 days ago	updated docs with some new details [philcryer]	
etc/	June 13, 2011	some sed-magic in build.conf() [acoolon]	
LICENSE	March 25, 2011	install modifications, finished new config setup [philcryer]	
README.rdoc	4 days ago	updated install link again... [philcryer]	
install.sh	4 days ago	checks that lsyncd is version 2.x or greater, othe... [philcryer]	

lipsync (<https://github.com/philcryer/lipsync>)

1) A new file is added to Client 01, lipsync sees it, and kicks off the sync with the server.



2) A new file is added to Client 02, lipsync sees it, kicks off the sync with the server, where it also finds a file that it needs.



3) While Client 01 could get the file the next time it has a new file to sync, instead it kicks off the scheduled cron job, where it finds the server has a file it needs.



4) Client 01 has the same files that Client 02 has, thanks to the Server syncing the files.



Demo



- show canned demo video
 - multiple terminals and file managers can be seen at once for visual impact
 - put a file in one, watch the logs scroll and watch the file appear on the other nodes
 - quickly demonstrate installing it on another node (speed up the 'film')
 - show that new node being part of the mix, with all the files intact after the install
 - add a file to this new node, watch it show up on the others
 - crowd cheers, I raise my arms in a 'V' formation and stage dive into the audience

Currently

- Isyncd2 daemon to handle the watching for file changes via inotify
- kicks off rsync over ssh to securely sync the data
- a contributor has Isyncd2 running on OSX (cross platform phase one)
- preliminary ideas of how a win32 version 'could work' with the installer running under cygwin
- great response from the community, user's forking the project, submitting patches and contributing to an active mailing list

Future echos

- make it truly cross platform
 - Linux, Mac, Windows, Android, iOS, etc
 - one installer to rule them all!
- make it more secure/private/etc
 - encrypted filesystems, p2p?
- more ideas from the community



<http://lipsync.it/>

lipsync

a lightweight commandline service that securely synchronizes your data

about code
issues install
discu ssion

Fork me on GitHub

Conclusion

- it's possible to create a secure, file distribution app that protects user's privacy and security...
 - but it won't be built by any for-profit, third party; it will be built by us
 - and we should look at other cloud offerings with this same skepticism
- get involved, try out and use lipsync, fork it, join the mailing list, submit an issue
- contribute your ideas, make changes, think about how it can be better
- always bring a towel!

- and remember...

FAILURE IS

Always

AN OPTION

Ken

Mike Hanley '88

lipsync
<http://lipsync.it>

Me
<http://philcryer.com>
[@fak3r](#)



Special thanks to



Thanks

Keep circulating the tapes!

