



THE (IN)SECURITY OF FILE HOSTING SERVICES

Nick Nikiforakis, Marco Balduzzi

Steven Van Acker, Wouter Joosen, Davide Balzarotti



6th July 2011

OWASP Netherland Chapter Meeting

Sharing is caring



- Internet expanding
 - ▣ More users
 - ▣ More Web services
 - ▣ More Web technologies
- Users need to share files
 - ▣ P2P is not always the answer
 - ▣ Emails?

Sharing Services



Dropbox

- ❑ Broad selection of services with a wide variety of applications
- ❑ Accessible through the Web from anywhere
- ❑ No software-bloating for users
- ❑ More free software due to a different way of making profit

UM... ISN'T YOUR
COMPUTER ON FIRE?



IT'S OKAY, MY FILES ARE
SAVED ON **DROPBOX**.
IT WAS GETTING KINDA
COLD IN HERE ANYWAY.



Bad news...

- A user's data is now located somewhere else:
 - Privacy ←
 - Availability
 - Integrity
- Sad story (T-Mobile & Microsoft):
 - 2009: “personal information stored on your device--such as contacts, calendar entries, to-do lists or photos--that is no longer on your Sidekick almost certainly has been lost as a result of a server failure at Microsoft/Danger”



File Hosting Services

- ❑ Cloud-storage for the masses
- ❑ Share files with other users
- ❑ Security through obscurity access-control
- ❑ Sharing personal documents as well as pirated files



Lifecycle of a file

- Alice decides to shares some digital content (file) through a FHS
- FHS received the file, stores it on its Cloud and generates an identifier which it:
 - i. binds with the uploaded file
 - ii. returns to the user in a URI form:
[http://www.easy-share.com/1916472551/
noctambus.pdf](http://www.easy-share.com/1916472551/noctambus.pdf)
- URI is shared depending on the nature of the uploaded file

File Identifier & Privacy



- The identifier (ID) is used to enforce access-control in a security-through-obscurity way
 - ▣ ID == access to file
- FHS are typically not-searchable
 - ▣ ID acts as a shared secret between a FHS and each user's files
 - ▣ Non-owners should not be able to “guess” this secret

100 FHSs: How many privacy-aware?



- We studied 100 FHSs to discover, among others, the way they generate unique “secret” identifiers
 - ▣ Uploading files, recording the given ID and comparing
- Removed 12 that had search/browse capabilities

Sequential IDs

- 34/88 FHS were generating sequential identifiers
 - ▣ numeric, or alphanumerical
- 20/34 did not append any other non-guessable information
 - ▣ e.g. filename or secondary ID
- E.g.
 - ▣ <http://vulnerable.com/9996>
 - ▣ <http://vulnerable.com/9997>
 - ▣ <http://vulnerable.com/9998>

Sequential IDs



- Designed a crawler for the 20 sequential FHS
- Run for 30 days
 - ▣ Random delays to limit bandwidth and blacklisting
 - ▣ Scraping only the filenames and sizes (privacy)
- Results:
 - ▣ > 310,000 file records

Finding private files...

- Depending on the nature of a file, it will be shared in different ways
- Exploit the ubiquity of search-engine crawlers to characterize a file as private or public.
- Given a filename
 - ▣ 0 search results -> Private



Private Files Results

- Using Bing:
 - ▣ 54.16% of files returned 0 search results
 - ▣ Rough approximation of private files due to close pirate communities

Filetype	#Private documents
Images (JPG, GIF, BMP)	27,711
Archives (ZIP)	13,354
Portable Document Format	7,137
MS Office Word	3,686
MS Office Excel Sheets	1,182
MS Office PowerPoint	967

Identifiers of 100 FHSs (summary)

	Sequential ID	Non-Sequential ID	Tot
Filename:			
Required	14	6	20
Not required	20	48	68
Total	34	54	88

Table 1: Analysis of the Download URI's identifier

Non-Sequential IDs

- 54 FHSs adopt non-sequential identifiers
- `len(ID)`

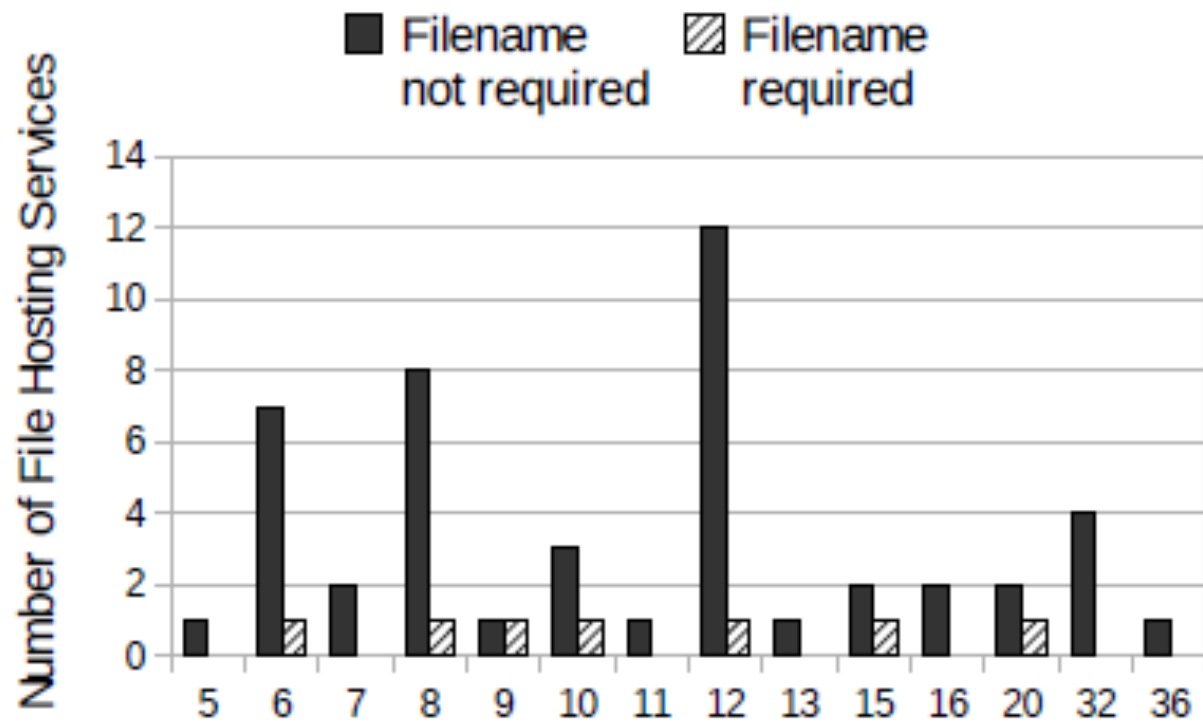


Figure 1: Length of the Identifier

Non-Sequential IDs

- 54 FHSs adopt non-sequential identifiers
- `len(C_SET)`

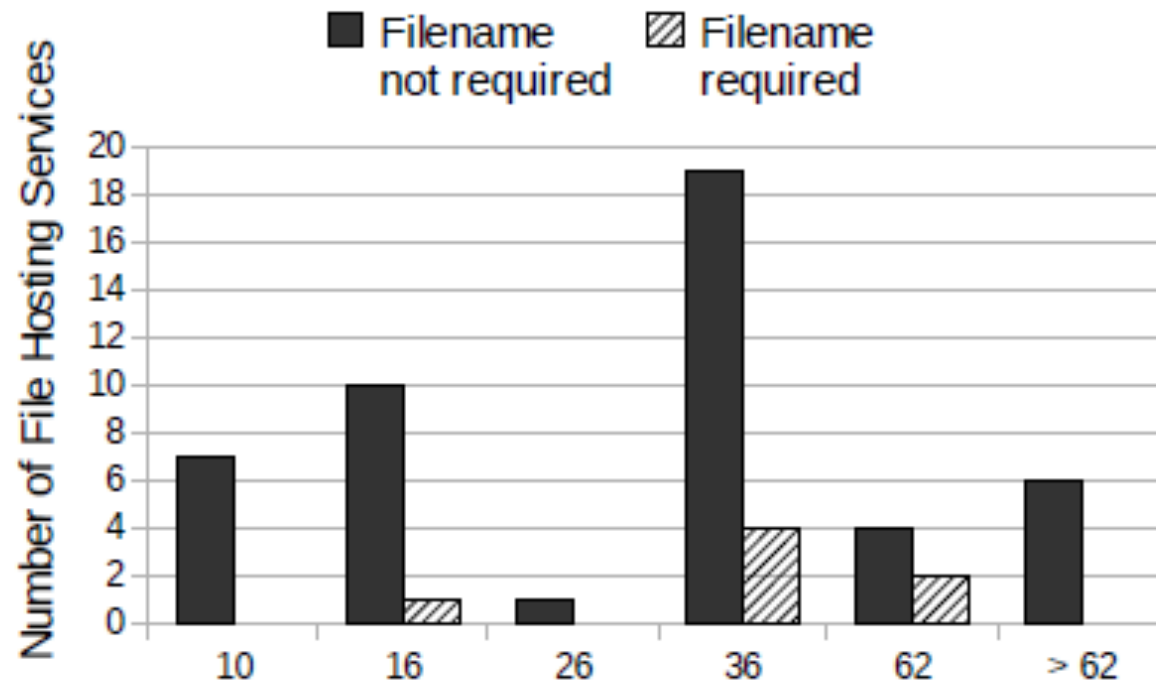


Figure 2: Size of the Identifier's Character Set

Random but short

- Brute-force short random identifiers

Length	Charset	#Tries	#Files Found
6	Numeric	617,169	728
6	Alphanumeric	526,650	586
8	Numeric	920,631	332

Design & Implementation errors



- Security audit of a popular FHS software product
 - ▣ Used in 13% of FHSs
 - ▣ Directory traversal vulnerability
 - ▣ De-randomization attack for deletion code
 - Report-link contained the first 10 characters of the 14-character delete code
 - $16^{14} \rightarrow 16^4$ combinations

Status...



- File hosting services are not privacy-aware
 - ▣ Sequential identifiers
 - ▣ Weak non-sequential identifiers
 - ▣ Bugs in their source code
- Do attackers know about this?
 - ▣ How do we found out?

Honey-pot experiment

- Honeyfiles promising valuable content
 - ▣ Phished_paypal_details.html
 - ▣ Paypal_account_gen.exe
 - ▣ Sniffed_email1.doc

- Each file connects back to our monitor server when opened
 - ▣ in HTML files
 - ▣ embedded HTML in doc files
 - ▣ TCP socket in executables
 - ▣ Attempt to open page in pdf files



Carding forum

- One of the decoy files contained valid credentials for our fake forum
 - ▣ `card3rz_reg_details.html`
- Fake underground carding community
 - ▣ `card3rz.co.cc`
- Reasons:
 - i. Hide our monitors
 - ii. Do attackers use data that they find in illegally obtained files?





Card3rz Login

Username

Password

Login

This website is for similarly minded people. Unless you have a valid
username/password combination, you are adviced to leave...

NOW

Honeypot experiment: results



- Monitoring sequential FHSs for 30 days
- 275 honeyfiles accesses
- More than 80 unique IP addresses
- 7 different sequential FHSs
 - ▣ 1 had a catalogue functionality
 - ▣ 2 had a search functionality
 - ▣ 4 had neither!
- Accesses from all around the world

Geo-location of the bad-guys



HoneyFiles results

□ Download ratio of each file:

Claimed content	Download ratio
Credentials to PayPal accounts	40.36%
Credentials for card3rz.co.cc	21.81%
PayPal account Generator	17.45%
Leaked customer list	9.09%
Sniffed email	6.81%
List of emails for spamming purposes	5.09%

card3rz.co.cc results



- 93 successful logins
 - ▣ 43 different IP addresses
 - ▣ 32% came back at a later time
- Attacks against the monitor and the login-form
 - ▣ SQL-injection & file-inclusion attacks
- Attackers do in-fact use data from illegally obtained files

Status...



- File hosting services are vulnerable
 - ▣ Sequential identifiers
 - ▣ Weak non-sequential identifiers
 - ▣ Bugs in their source code
- Attackers are abusing them
 - ▣ They are using the data found in other user's files

SecureFS

- A client must protect himself
- Encryption is a good way
 - ▣ Do people know how to?
 - ▣ If they do know, does their OS assist them?
- SecureFS
 - ▣ Encryption to protect a user's data
 - ▣ Steganography to mislead potential attackers
- Project site: <http://www.securitee.org/sfs/>



SecureFS

- Browser-plugin monitoring uploads and downloads
- Protects uploads on-the-fly



- Browser-plugin monitoring uploads and downloads
- Rewrites download links to include the random key
 - <http://unsafefhs.com/12345>
 - [http://unsafefhs.com/12345/sfs_key/\[RND_KEY\]](http://unsafefhs.com/12345/sfs_key/[RND_KEY])

Conclusion



- Large percentage of FHSs fail to provide the user with adequate privacy
 - ▣ Hundreds of thousands of files ready to be misused
- Attackers know & exploit this fact
- A user must protect himself:
 - ▣ *SecureFS*

Questions

