

# Крокодил

---

Подготовил: One-nine9, ilinor

---

## Вступление

---

Уровень I — это все векторы эксплуатации, которые соединяются вместе, чтобы дать вам возможность закрепиться на цели от одного сервиса к другому. Учетные данные могут быть потеряны где-то в общедоступной папке, что позволит вам войти в систему через удаленную оболочку, оставленную без присмотра и контроля. Неправильно настроенная служба может привести к утечке информации, которая позволит вам выдать себя за цифровую личность жертвы. В реальном мире существует любое количество возможностей. Однако начнем с более простых.

Рассматривая пример, сшитый из двух других предыдущих целей, мы рассмотрим небезопасную конфигурацию доступа на FTP и административный вход для веб-сайта. Приступим к деконструкции этого вектора и анализу его составляющих.

---

## перечисление

---

Мы начнем с перечисления цели. Наш первый шаг — это, как всегда, тщательное сканирование nmap. Используя следующие два переключателя для сканирования, мы гарантируем, что наш сценарий nmap анализирует службу, запущенную на любом порту, найденном в открытым состоянии и возвращает в основном точное значение версии службы на выходе, а также то, что все сценарии анализа по умолчанию запускаются для цели, поскольку мы не ограничены в том, насколько навязчивыми мы можем быть при сканировании. Запустив сканирование, как уже упоминалось, мы можем получить результаты, как показано ниже, с фрагментами каталогов, которые сканирование даже нашло для нас!

- sC: выполняет сканирование сценария с использованием набора сценариев по умолчанию. Это эквивалентно -- script=default. Некоторые скрипты в этой категории считаются навязчивыми и не должны запускаться в целевой сети без разрешения.

- sV: включает определение версии, которое будет определять, какие версии работают на каком порту.



```
$ sudo nmap -sC -sV {target_IP}
```

```
Starting Nmap 7.91 ( https://nmap.org ) at 2021-07-09 11:48 CEST
```

```
Nmap scan report for {target_IP}
```

```
Host is up (0.21s latency).
```

```
Not shown: 998 closed ports
```

```
PORT      STATE SERVICE VERSION
```

```
21/tcp open  ftp      vsftpd 3.0.3
```

```
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
```

```
| -rw-r--r--  1 ftp      ftp          33 Jun 08 10:58 allowed.userlist
```

```
|_rw-r--r--  1 ftp      ftp          62 Apr 20 11:32 allowed.userlist.passwd
```

```
| ftp-syst:
```

```
|  STAT:
```

```
|  FTP server status:
```

```
|    Connected to ::ffff:{user_IP}
```

```
|    Logged in as ftp
```

```
|    TYPE: ASCII
```

```
|    No session bandwidth limit
```

```
|    Session timeout in seconds is 300
```

```
|    Control connection is plain text
```

```
|    Data connections will be plain text
```

```
|    At session startup, client count was 2
```

```
|    vsFTPD 3.0.3 - secure, fast, stable
```

```
|_End of status
```

```
80/tcp open  http      Apache httpd 2.4.41 ((Ubuntu))
```

```
|_http-server-header: Apache/2.4.41 (Ubuntu)
```

```
|_http-title: Smash - Bootstrap Business Template
```

```
Service Info: OS: Unix
```

```
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
```

```
Nmap done: 1 IP address (1 host up) scanned in 11.67 seconds
```

У нас есть два открытых порта: 21 и 80. Порт 21 — это порт, предназначенный для FTP (протокола передачи файлов), что означает, что его основное использование — передача файлов между хостами в одной сети.

Согласно Википедии, краткое напоминание:

Протокол передачи файлов (FTP) — это стандартный протокол связи, используемый для передачи компьютерных файлов с сервера на клиент в компьютерной сети. Пользователи FTP могут аутентифицировать себя с помощью протокола входа в открытый текст, обычно используя имя пользователя и пароль. Однако они могут подключаться анонимно, если сервер настроен на это.

Пользователи могли подключаться к FTP-серверу анонимно, если сервер настроен на это, а это означает, что мы могли использовать его, даже если у нас не было действительных учетных данных. Если мы вернемся к результатам сканирования nmap, то увидим, что FTP-сервер действительно настроен на анонимный вход в систему:

ftp-anon: разрешен анонимный FTP-вход (FTP-код 230)

Если вам нужна переподготовка, `фТП -ч` Команда поможет вам разобраться с доступными командами для FTP сервис на вашем локальном хосте.

```
$ ftp -h

Usage: { ftp | pftp } [-46pinegvtd] [hostname]
  -4: use IPv4 addresses only
  -6: use IPv6, nothing else
  -p: enable passive mode (default for pftp)
  -i: turn off prompting during mget
  -n: inhibit auto-login
  -e: disable readline support, if present
  -g: disable filename globbing
  -v: verbose mode
  -t: enable packet tracing [nonfunctional]
  -d: enable debugging
```

Чтобы подключиться к удаленному FTP-серверу, вам необходимо указать целевой IP-адрес (или имя хоста), как показано на странице лаборатории Starting Point. Затем в приглашении будут запрошены наши учетные данные для входа, где мы можем заполнить `анонимный` имя пользователя. В нашем случае FTP-сервер не запрашивает пароль, и ввод `анонимный` имени пользователя достаточно, чтобы мы получили код 230, `Авторизация успешна`.

```
$ ftp {target_IP}

Connected to {target_IP}.
220 (vsFTPd 3.0.3)
Name ({target_IP}:{username}): anonymous
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.

ftp>
```

После входа в систему вы можете ввести `помощь` команда для проверки доступных команд.



```
ftp> help
```

Commands may be abbreviated. Commands are:

!	dir	mdelete	qc	site
\$	disconnect	mdir	sendport	size
account	exit	mget	put	status
append	form	mkdir	pwd	struct
ascii	get	mls	quit	system
bell	glob	mode	quote	sunique
binary	hash	modtime	recv	tenex
bye	help	mput	reget	tick
case	idle	newer	rstatus	trace
cd	image	nmap	rhel	type
cdup	ipany	nlist	rename	user
chmod	ipv4	ntrans	reset	umask
close	ipv6	open	restart	verbose
cr	lcd	prompt	rmdir	?
delete	ls	passive	runique	
debug	macdef	proxy	send	

Мы будем использовать `директор` и `получить` для просмотра каталогов и управления файлами, хранящимися на FTP-сервере. С

`директор` Команда, мы можем проверить содержимое нашего текущего каталога на удаленном хосте, где внимание привлекают два интересных файла. Кажется, это файлы, оставшиеся от конфигурации другой службы на хосте, скорее всего, веб-сервера HTTPD. Их имена описательные, намекающие на возможный список имен пользователей и связанных с ними паролей.



```
ftp> dir
```

```
200 PORT command successful. Consider using PASV.
```

```
150 Here comes the directory listing.
```

```
-rw-r--r--  1 ftp      ftp          33 Jun 08 10:58 allowed.userlist
-rw-r--r--  1 ftp      ftp          62 Apr 20 11:32 allowed.userlist.passwd
226 Directory send OK.
```

Оба файла можно легко загрузить с помощью `получить` команда. Служба FTP сообщит о загрузке завершение статуса обратно к вам на этом этапе. Это не должно занять много времени, чтобы они оба уютно расположились на вашей атакующей виртуальной машине.

```
ftp> get allowed.userlist

local: allowed.userlist remote: allowed.userlist
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for allowed.userlist (33 bytes).
226 Transfer complete.
33 bytes received in 0.00 secs (257.8125 kB/s)

ftp> get allowed.userlist.passwd

local: allowed.userlist.passwd remote: allowed.userlist.passwd
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for allowed.userlist.passwd (62 bytes).
226 Transfer complete.
62 bytes received in 0.00 secs (126.6671 kB/s)
```

Завершение FTP-соединения можно выполнить, вернув вкладку `выход` команда. Это вернет текущий терминала в предыдущее состояние.

```
ftp> exit

221 Goodbye.
```

Сразу после выхода из оболочки службы FTP мы можем ввести `ЛС` команда, чтобы проверить, являются ли наши файлы присутствуют в каталоге, в котором мы находились в последний раз. Чтобы прочитать их содержимое и обнаружить имена пользователей и пароли внутри, мы можем использовать `Кот` команда, за которой следует имя файла, который мы хотим открыть.



```
$ cat allowed.userlist

aron
pwnmeow
egotisticalsw
admin

$ cat allowed.userlist.passwd

root
Supersecretpassword1
@BaASD&9032123sADS
rKXM59ESxesUFHAd
```

---

## плацдарм

---

После получения учетных данных следующим шагом будет проверка, используются ли они в службе FTP для повышенного доступа или на веб-сервере, работающем на порту 80, обнаруженном во время сканирования nmap. Попытка войти с любыми учетными данными на FTP-сервере возвращает код ошибки 530 Этот FTP-сервер является анонимным

Только . Здесь не повезло, поэтому мы можем выйти из оболочки службы FTP.



```
$ ftp {target_IP}

Connected to {target_IP}.
220 (vsFTPd 3.0.3)
Name ({target_IP}:{username}): aron
530 This FTP server is anonymous only.
Login failed.

ftp> exit
221 Goodbye.
```

Однако у нас остается один вариант. Во время сканирования nmap служба, работающая на порту 80, была зарегистрирована как Апач httpd 2.4.41 , HTTP-сервер Apache. Введите IP-адрес цели в поле нашего браузера.

Панель поиска URL приводит к этой веб-странице. Кажется, это витрина для компании, занимающейся хостингом серверов.

[HOME](#)[SERVICES](#)[PORTFOLIO](#)[PRICING](#)[ABOUT](#)[TEAM](#)[CONTACT](#)[DOWNLOAD](#)

# Business is Now Digital

We blend insights and strategy to create digital products for forward-thinking organisations.

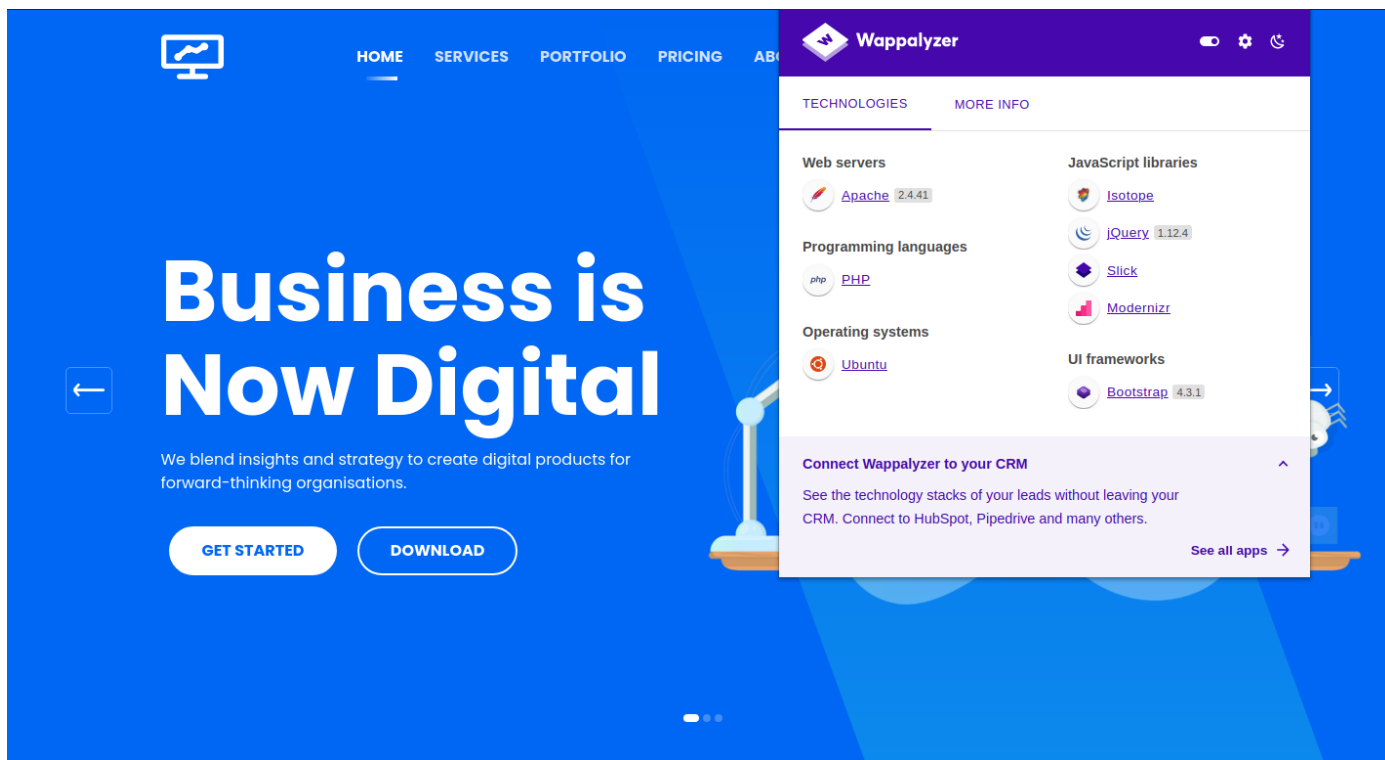
[GET STARTED](#)[DOWNLOAD](#)

Чтение о цели полезно, но только на поверхностном уровне. Чтобы лучше понять технологию, которую они использовали для создания своего веб-сайта, и, возможно, найти любые связанные с ними уязвимости, мы можем использовать удобный плагин для браузера под названием [Банпал уЗер](#). Этот подключаемый модуль анализирует код веб-страницы и возвращает все различные технологии, использованные для ее создания, такие как тип веб-сервера, библиотеки JavaScript, языки программирования и многое другое. Вы можете нажать на ссылки ниже, чтобы добавить подключаемый модуль в выбранный вами браузер.

[Добавьте меня в Chrome!](#)

[Добавьте меня в Firefox!](#)

После установки вы можете получить доступ к Wappalyzer, нажав на его значок в правом верхнем углу окна браузера. Ниже приведены результаты для нашей текущей цели.



Из вывода Wappalyzer мы можем отметить некоторые из наиболее интересных элементов, в частности, язык программирования PHP, используемый для создания веб-страницы. Однако пока ничто не дает нам прямого плана атаки. Между тем, навигация по странице с помощью вкладок и кнопок, представленных на ней, никуда не приводит. Ссылаясь на предыдущие рецензии, упоминается другой, более прямой способ навигации по любым скрытым или труднодоступным каталогам и страницам, а именно через удаление каталогов. Используя gobuster в качестве нашего предпочтительного инструмента, мы можем использовать следующие переключатели для скрипта, чтобы получить самые быстрые и точные результаты.

dir : использует режим перечисления каталогов/файлов.

-- url : целевой URL.

--wordlist : Путь к списку слов.

- x : Расширения файлов для поиска.

Для `- Икс` переключатель, мы можем указать `php` и `HTML` чтобы отфильтровать весь ненужный беспорядок, который не заинтересовать нас. Файлы PHP и HTML чаще всего являются страницами. Возможно, нам повезет, и мы найдем страницу входа в административную панель, которая поможет нам найти рычаги против цели в сочетании с учетными данными, которые мы извлекли с FTP-сервера.



```

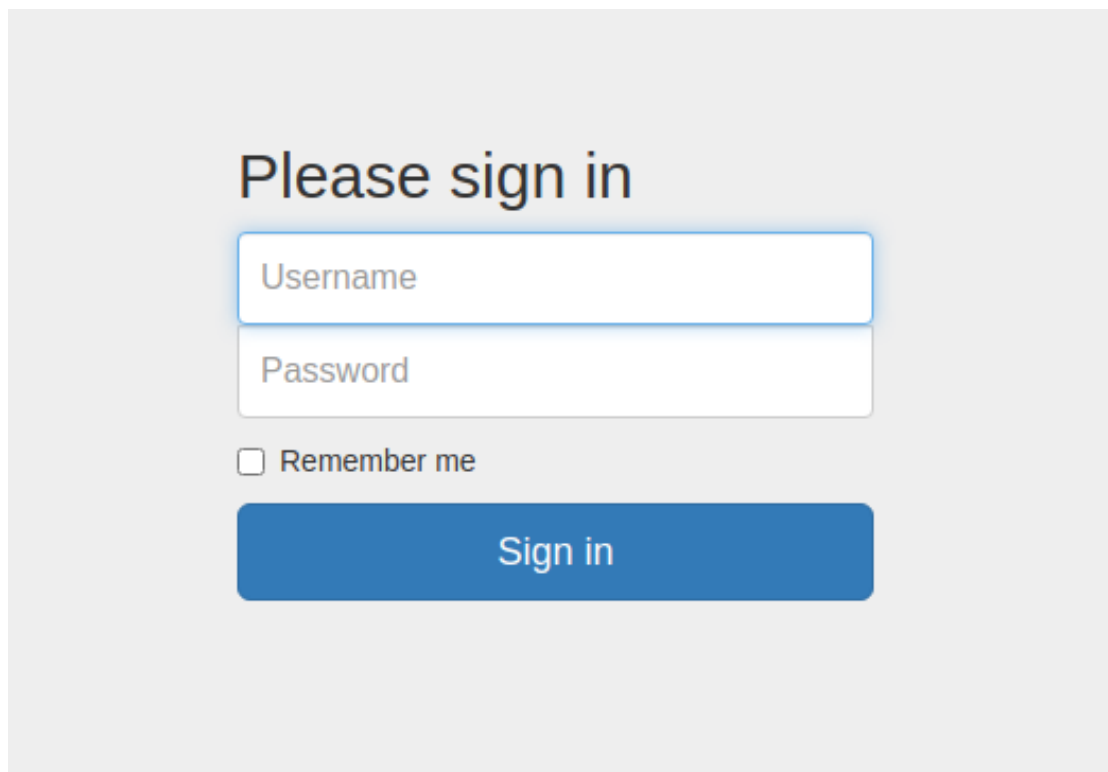
$ gobuster dir --url http://{target_IP}/ --wordlist /usr/share/wordlists/dirbuster
/directory-list-2.3-small.txt -x php,html

=====
Gobuster v3.1.0
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
=====
[+] Url: http://{target_IP}/
[+] Method: GET
[+] Threads: 10
[+] Wordlist: /usr/share/wordlists/dirbuster/directory-list-2.3-small.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.1.0
[+] Extensions: php,html
[+] Timeout: 10s
=====
2021/07/09 12:56:23 Starting gobuster in directory enumeration mode
=====
/index.html (Status: 200) [Size: 58565]
/login.php (Status: 200) [Size: 1577]
/assets (Status: 301) [Size: 311] [--> http://{target_IP}/assets/]
/css (Status: 301) [Size: 308] [--> http://{target_IP}/css/]
/js (Status: 301) [Size: 307] [--> http://{target_IP}/js/]
/logout.php (Status: 302) [Size: 0] [--> login.php]
/config.php (Status: 200) [Size: 0]
/fonts (Status: 301) [Size: 310] [--> http://{target_IP}/fonts/]
/dashboard (Status: 301) [Size: 314] [--> http://{target_IP}/dashboard/]

=====
2021/07/09 13:22:08 Finished
=====

```

Одним из наиболее интересных файлов, извлеченных gobuster, является `/login.php` страница. Переход вручную к URL-адрес в виде `http://{целевой_IP}/login.php`, нас встречает страница входа с запросом комбинация имени пользователя/пароля.



Please sign in

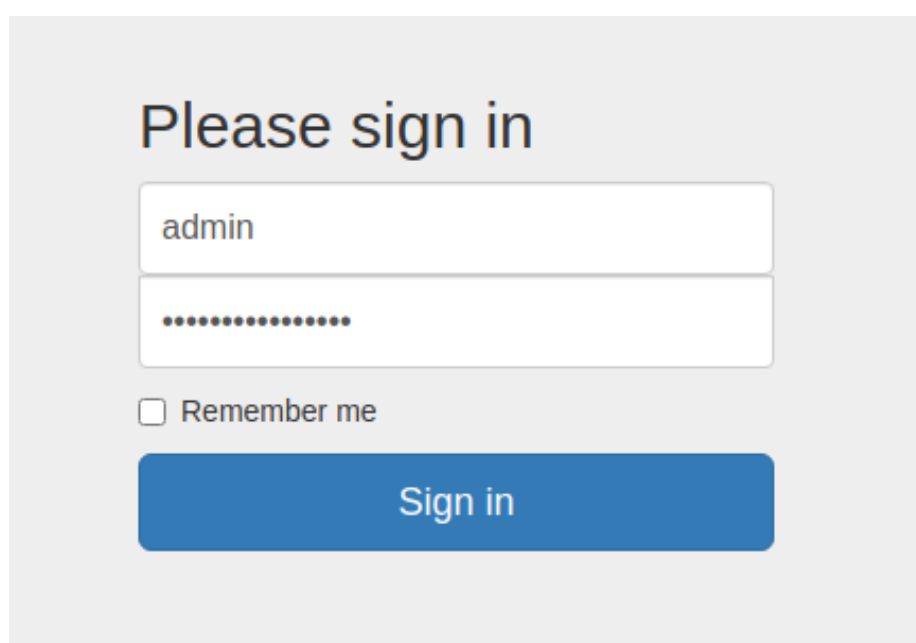
Username

Password

☐ Remember me

Sign in

Если бы списки учетных данных, которые мы нашли, были бы длиннее, мы могли бы использовать модуль Metasploit или скрипт грубой силы для входа в систему, чтобы выполнять комбинации из обоих списков быстрее, чем вручную. Однако в этом случае списки относительно малы, что позволяет нам попытаться войти в систему вручную.



Please sign in

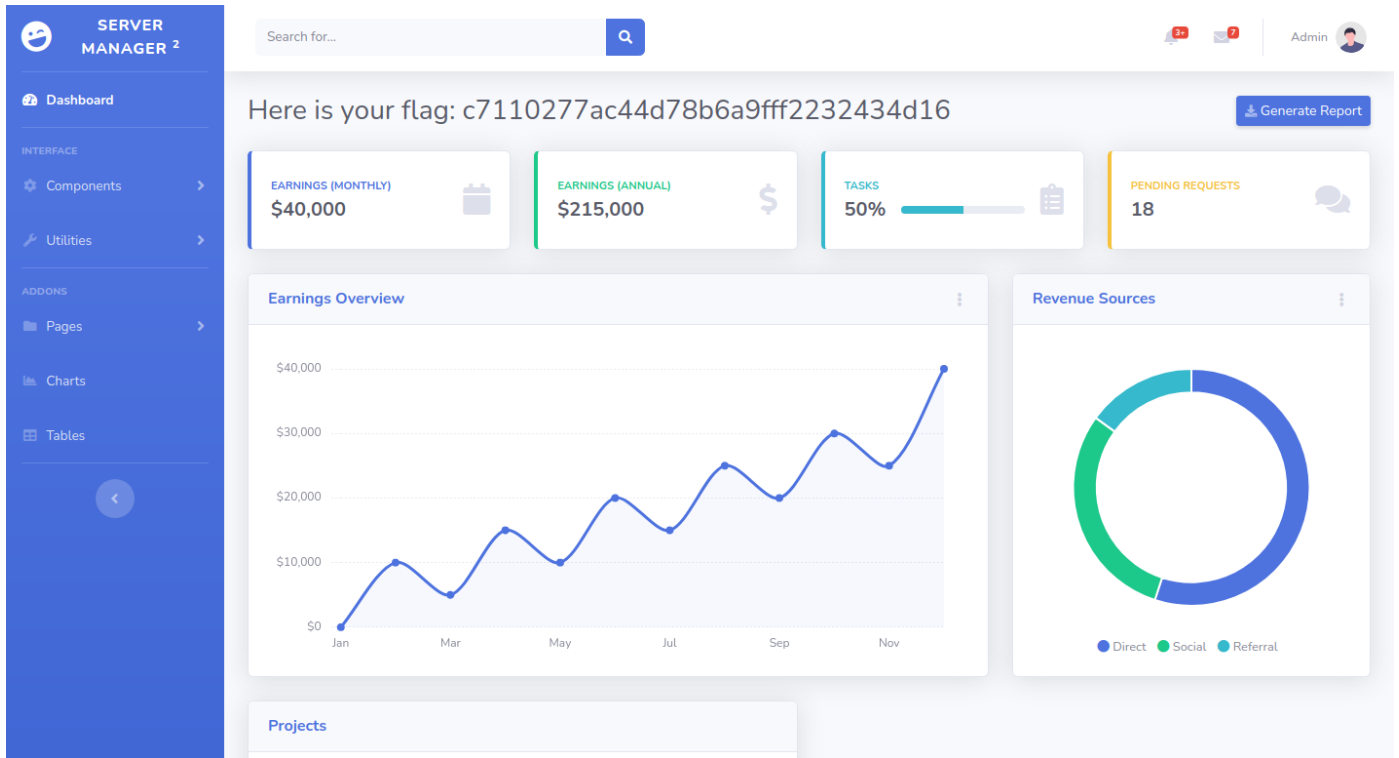
admin

.....

☐ Remember me

Sign in

После попытки нескольких комбинаций имени пользователя и пароля нам удалось войти в систему, и мы встретились с административной панелью диспетчера серверов. Оказавшись здесь, злоумышленник может манипулировать веб-сайтом любым удобным для него способом, нанося ущерб пользовательской базе и владельцам или извлекая дополнительную информацию, которая поможет им закрепиться на серверах, на которых размещена веб-страница.



Мы успешно получили флаг! Он отображается у нас в верхней части админки.

Отличная работа!