

Основы построения защищенных компьютерных сетей

Лекция 3
Сетевые атаки

Семён Новосёлов

2021



Стадии сетевой атаки

- 1. **Сбор информации**
 - информация о ПО и его версиях, компонентах, доменах, узлах
- 2. **Анализ уязвимостей**
 - поиск по базам известных уязвимостей и эксплойтов
 - самостоятельное нахождение уязвимостей в ПО
- 3. **Эксплуатация**
 - разработка/выбор эксплойта и его использование
- 4. **Постэксплуатация**
 - выполнение вредоносной нагрузки (рассылка спама, сбор личной информации)
 - эскалация привилегий, продвижение по сети дальше
- 5. **Подготовка отчёта (при аудите)**

По уровню модели TCP/IP

1. Канальный уровень
2. Сетевой уровень
3. Транспортный уровень
4. Прикладной уровень

→ Ethernet/WLAN

ARP-poisoning

MAC-spoofing

DHCP-spoofing

По уровню модели TCP/IP

1. Канальный уровень
2. Сетевой уровень
3. Транспортный уровень
4. Прикладной уровень

→ IP, ICMP

IP-spoofing

Ping-flood

IP fragmentation

Пересекающиеся
фрагменты

Переполнение
буфера пакетов

Превышение
макс. размера IP-
пакета

По уровню модели TCP/IP

1. Канальный уровень
2. Сетевой уровень
3. Транспортный уровень
4. Прикладной уровень

→ TCP, UDP, TLS

SYN-flood

UDP-flood

TCP-reset/hijacking

Атаки на TLS

Heartbleed

Poodle

По уровню модели TCP/IP

1. Канальный уровень
2. Сетевой уровень
3. Транспортный уровень
4. Прикладной уровень

→ HTTP/SSH/DNS

Brute-Force

Атаки на
уязвимости в ПО

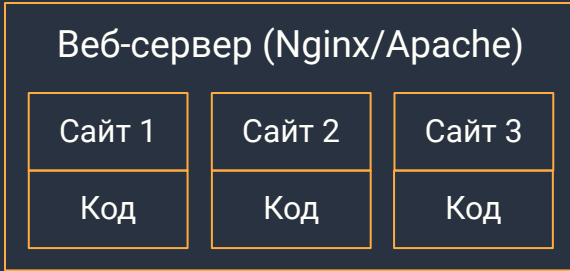
Web-атаки

DNS-spoofing

Атаки на веб-приложения

- На серверное ПО
- На серверную часть приложения (backend)
- На клиентскую часть приложения (frontend)

Схема работы веб-приложений и сайтов. 1/5



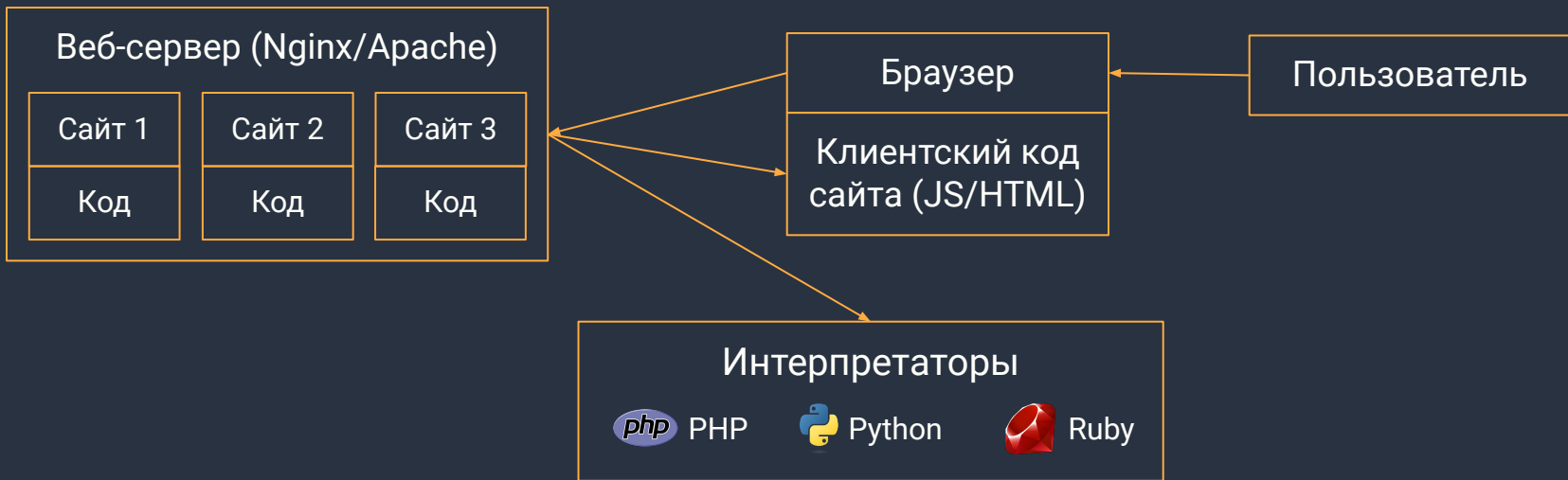
Сайты располагаются на веб-сервере.

Схема работы веб-приложений и сайтов. 2/5



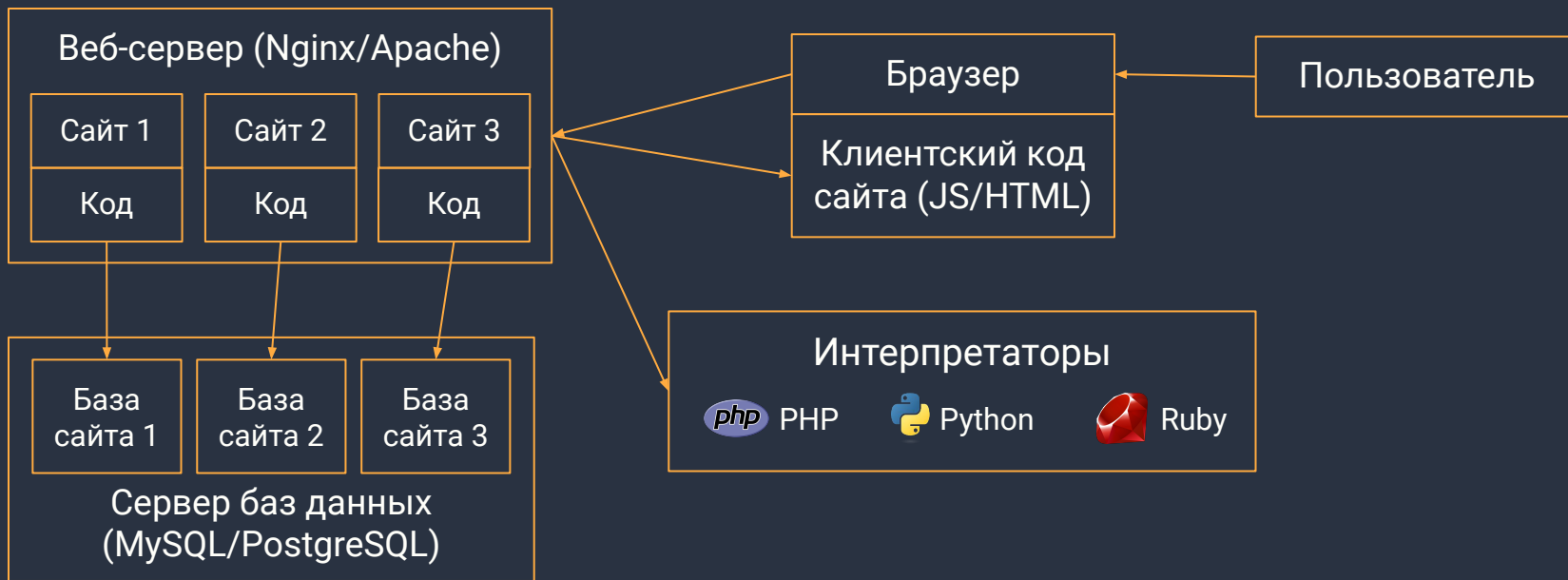
Код сайта состоит из **серверной части (backend)** и **клиентской части (frontend)**

Схема работы веб-приложений и сайтов. 3/5



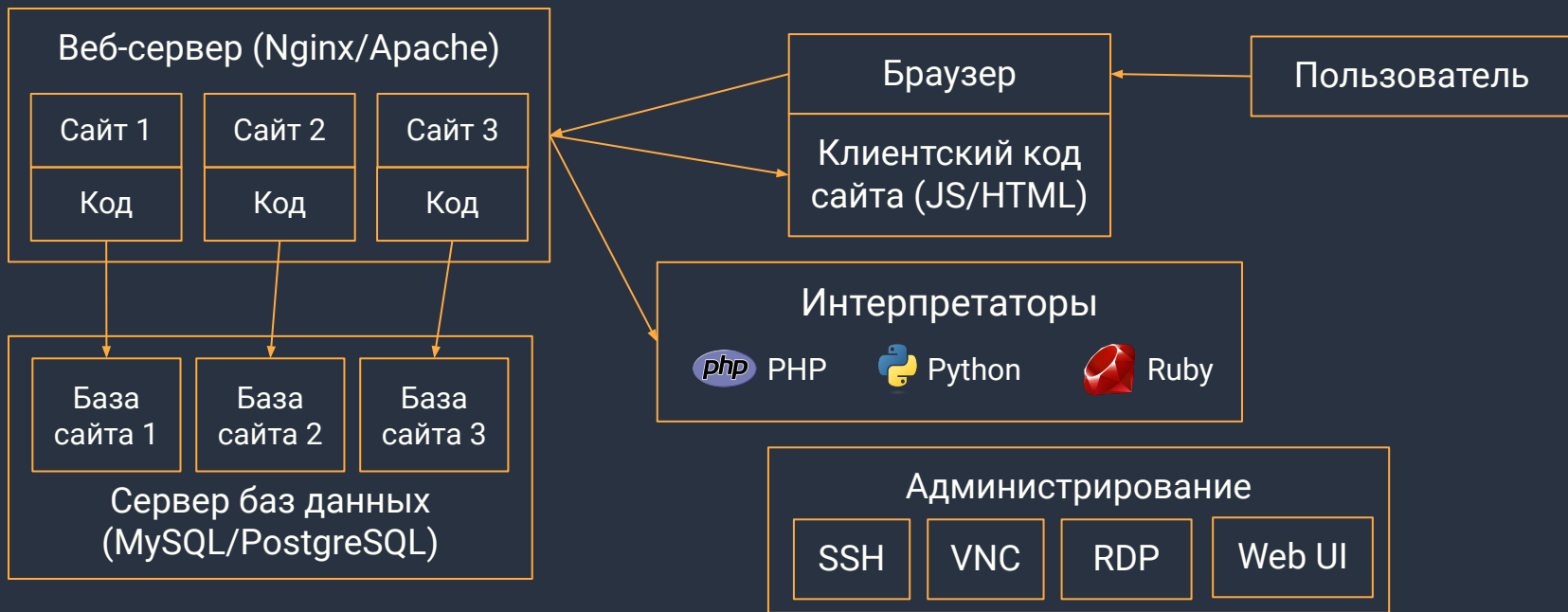
Для выполнения серверной части кода сервер обращается к интерпретаторам языков программирования.

Схема работы веб-приложений и сайтов. 4/5



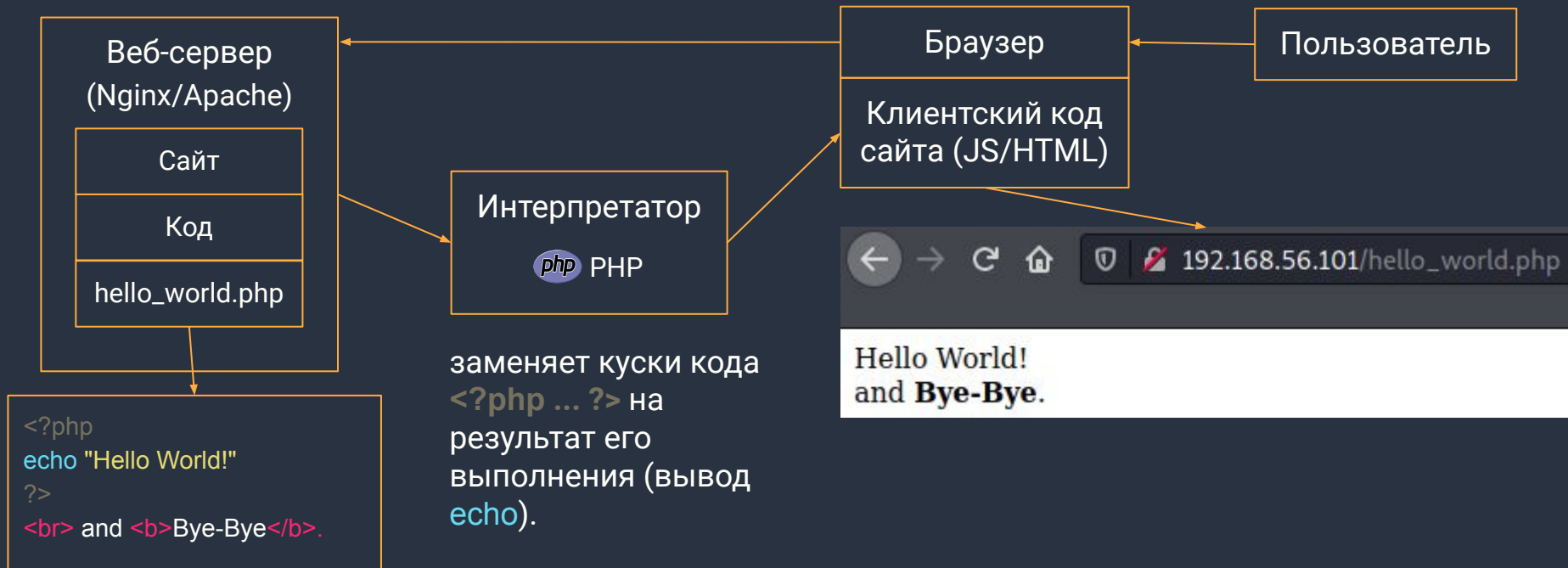
Серверный код может подключаться к БД с учетными данными пользователей или другой информацией

Схема работы веб-приложений и сайтов. 5/5



Администраторы подключаются к отдельно запущенным программам-серверам.

Простой пример



Серверное программное обеспечение

Веб-сервер (Nginx/Apache)

Сайт 1

Сайт 2

Сайт 3

Интерпретаторы



PHP



Python



Ruby

База
сайта 1

База
сайта 2

База
сайта 3

Сервер баз данных
(MySQL/PostgreSQL)

Администрирование

SSH

VNC

RDP

Web UI

Серверное программное обеспечение выполняется на спец. компьютере-сервере (или нескольких)

Уязвимости серверного ПО

Веб-сервер (Nginx/Apache)

Сайт 1

Сайт 2

Сайт 3

База
сайта 1

База
сайта 2

База
сайта 3

Сервер баз данных
(MySQL/PostgreSQL)

Сервера обычно разрабатываются на C/C++ для скорости

Интерпретаторы



PHP



Python



Ruby

Администрирование

SSH

VNC

RDP

Web UI

Уязвимости серверного ПО

Веб-сервер (Nginx/Apache)

Сайт 1

Сайт 2

Сайт 3

База
сайта 1

База
сайта 2

База
сайта 3

Сервер баз данных
(MySQL/PostgreSQL)

Сервера обычно разрабатываются на C/C++ для скорости

Уязвимости: переполнения буфера, утечки памяти, ошибки конфигурации

Интерпретаторы



PHP



Python



Ruby

Администрирование

SSH

VNC

RDP

Web UI

Уязвимости серверного ПО

Веб-сервер (Nginx/Apache)

Сайт 1

Сайт 2

Сайт 3

База
сайта 1

База
сайта 2

База
сайта 3

Сервер баз данных
(MySQL/PostgreSQL)

Сервера обычно разрабатываются на C/C++ для скорости

Уязвимости: переполнения буфера, утечки памяти, ошибки конфигурации

Масштаб: затрагивают большое количество сайтов

Интерпретаторы



PHP



Python



Ruby

Администрирование

SSH

VNC

RDP

Web UI

Уязвимости серверного ПО

Веб-сервер (Nginx/Apache)

Сайт 1

Сайт 2

Сайт 3

База
сайта 1

База
сайта 2

База
сайта 3

Сервер баз данных
(MySQL/PostgreSQL)

Сервера обычно разрабатываются на C/C++ для скорости

Уязвимости: переполнения буфера, утечки памяти, ошибки конфигурации

Масштаб: затрагивают большое количество сайтов

Детектирование: сканирование портов и определение версий (Nmap), тестирование отправкой некорректных данных (фаззинг)

Интерпретаторы



PHP



Python



Ruby

Администрирование

SSH

VNC

RDP

Web UI

Уязвимости серверного ПО

Веб-сервер (Nginx/Apache)

Сайт 1

Сайт 2

Сайт 3

База
сайта 1

База
сайта 2

База
сайта 3

Сервер баз данных
(MySQL/PostgreSQL)

Интерпретаторы



PHP



Python



Ruby

Администрирование

SSH

VNC

RDP

Web UI

Сервера обычно разрабатываются на C/C++ для скорости

Уязвимости: переполнения буфера, утечки памяти, ошибки конфигурации

Масштаб: затрагивают большое количество сайтов

Детектирование: сканирование портов и определение версий (Nmap), тестирование отправкой некорректных данных (фаззинг)

Частые атаки: перебор паролей, исп. эксплойтов к старым версиям программ

Определение версий ПО с помощью Nmap

```
Nmap scan report for 192.168.1.45
Host is up (0.000015s latency).
Not shown: 977 filtered ports
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 2.3.4
22/tcp    open  ssh          OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp    open  telnet       Linux telnetd
25/tcp    open  smtp         Postfix smtpd
53/tcp    open  domain       ISC BIND 9.4.2
80/tcp    open  http         Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp   open  rpcbind      2 (RPC #100000)
139/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp   open  exec         netkit-rsh rexecd
513/tcp   open  login?
514/tcp   open  tcpwrapped
1099/tcp  open  rmiregistry  GNU Classpath grmiregistry
1524/tcp  open  bindshell    Metasploitable root shell
2049/tcp  open  nfs          2-4 (RPC #100003)
2121/tcp  open  ftp          ProFTPD 1.3.1
3306/tcp  open  mysql        MySQL 5.0.51a-3ubuntu5
```

Определение версий сервисов на машине с Metasploitable 2.

```
nmap -sV 192.168.1.45
```

здесь есть ssh, можно попытаться подобрать пароль:

```
nmap -p 22 --script ssh-brute 192.168.1.45
```

Код сайта. Серверная часть (backend)

Веб-сервер
(Nginx/Apache)

Сайт

Код

Серверная часть веб-приложения пишется на интерпретируемых языках:



PHP



Python



Ruby

Для упрощения задачи используются фреймворки:



Laravel



django

Код сайта. Серверная часть (backend)

Веб-сервер
(Nginx/Apache)

Сайт

Код

Серверная часть веб-приложения пишется на интерпретируемых языках:



PHP



Python



Ruby

Для упрощения задачи используются фреймворки:



Laravel



django

Уязвимости: инъекции кода, отсутствие аутентификации, ошибки конфигурации и логики программы

Код сайта. Серверная часть (backend)



Веб-сервер
(Nginx/Apache)

Сайт

Код

Серверная часть веб-приложения пишется на интерпретируемых языках:



PHP



Python



Ruby

Для упрощения задачи используются фреймворки:



Laravel

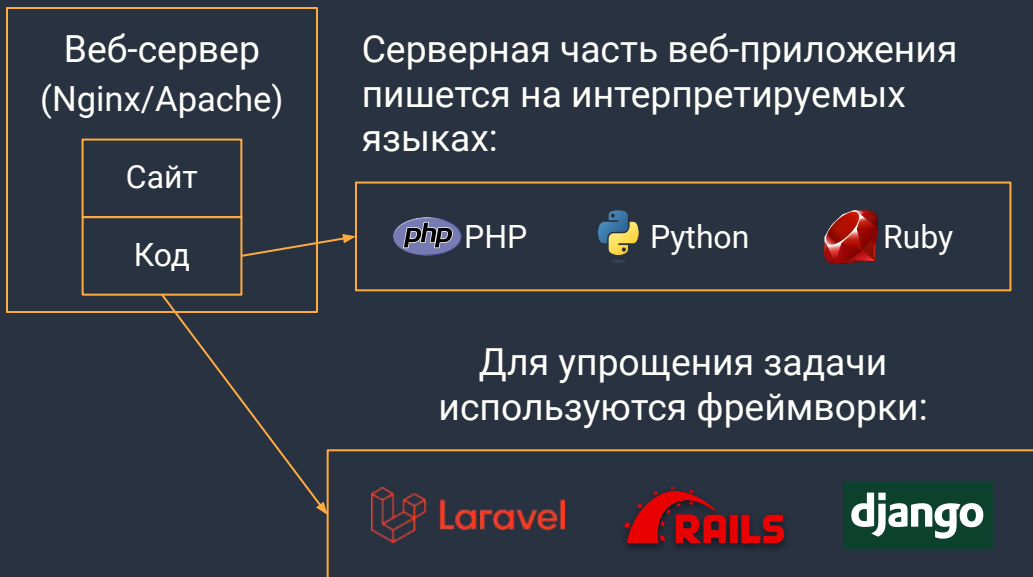


django

Уязвимости: инъекции кода, отсутствие аутентификации, ошибки конфигурации и логики программы

Масштаб: специфичны для сайта, для уязвимостей в фреймворке - все сайты на нём

Код сайта. Серверная часть (backend)

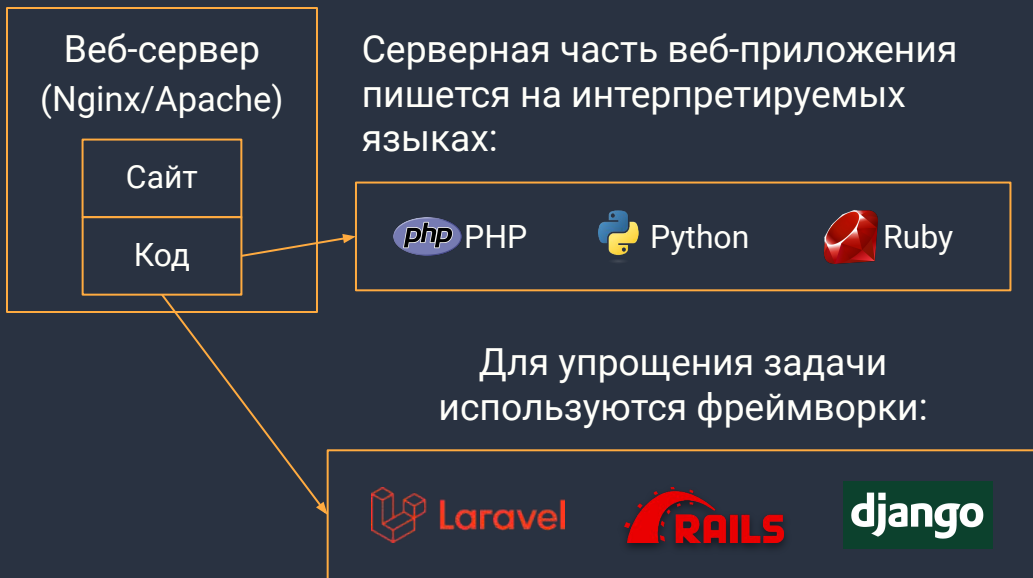


Уязвимости: инъекции кода, отсутствие аутентификации, ошибки конфигурации и логики программы

Масштаб: специфичны для сайта, для уязвимостей в фреймворке - все сайты на нём

Частые атаки: перебор паролей, исп. эксплойтов к старым версиям компонентов, инъекции

Код сайта. Серверная часть (backend)



Уязвимости: инъекции кода, отсутствие аутентификации, ошибки конфигурации и логики программы

Масштаб: специфичны для сайта, для уязвимостей в фреймворке - все сайты на нём

Частые атаки: перебор паролей, исп. эксплойтов к старым версиям компонентов, инъекции

Детектирование: сканирование на типичные ошибки конфигурации (Nikto), спец. сканеры под фреймворки и классы уязвимостей (sqlmap), ручной поиск

Пример. Сканирование сайта на ошибки конфигурации

```
L$ nikto -host 192.168.56.101
- Nikto v2.1.6

+ Target IP:      192.168.56.101
+ Target Hostname: 192.168.56.101
+ Target Port:    80
+ Start Time:    2021-04-06 20:54:05 (GMT-4)

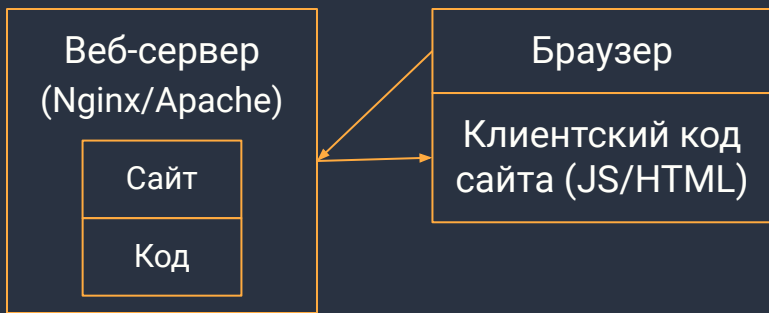
+ Server: Apache/2.2.8 (Ubuntu) DAV/2
+ Retrieved x-powered-by header: PHP/5.2.4-2ubuntu5.10
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type
+ Uncommon header 'tcn' found, with contents: list
+ Apache mod_negotiation is enabled with MultiViews, which allows attackers to easily brute force file names. See http://www.wisec.it/sectou.php?id=4698ebdc59d15. The following alternatives for 'index' were found: index.php
+ Apache/2.2.8 appears to be outdated (current is at least Apache/2.4.37). Apache 2.2.34 is the EOL for the 2.x branch.
+ Web Server returns a valid response with junk HTTP methods, this may cause false positives.
+ OSVDB-877: HTTP TRACE method is active, suggesting the host is vulnerable to XST
+ /phpinfo.php: Output from the phpinfo() function was found.
+ OSVDB-3268: /doc/: Directory indexing found.
+ OSVDB-48: /doc/: The /doc/ directory is browsable. This may be /usr/doc.
+ OSVDB-12184: /?=PHPB8B5F2A0-3C92-11d3-A3A9-4C7B08C10000: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings.
+ OSVDB-12184: /?=PHPE9568F36-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings.
+ OSVDB-12184: /?=PHPE9568F34-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings.
```

Вывод сканера **Nikto**.

Большинство ошибок не ведут к практическим атакам.

Доступен файл **phpinfo.php** с полной информацией о PHP и его расширениях

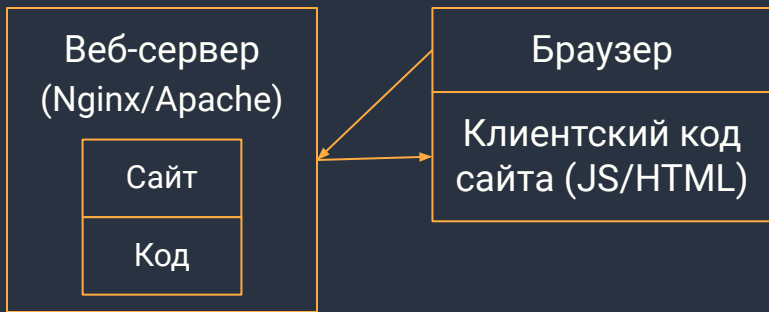
Код сайта. Клиентская часть (frontend)



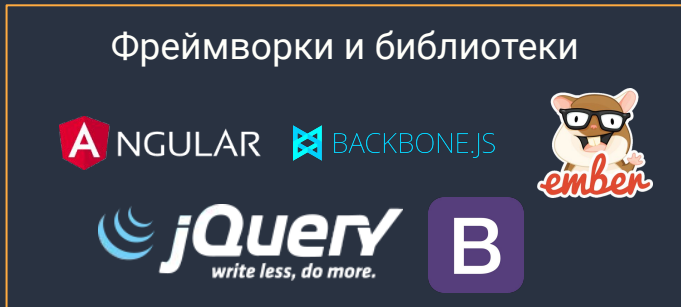
Клиентская часть выполняется в браузере пользователя.



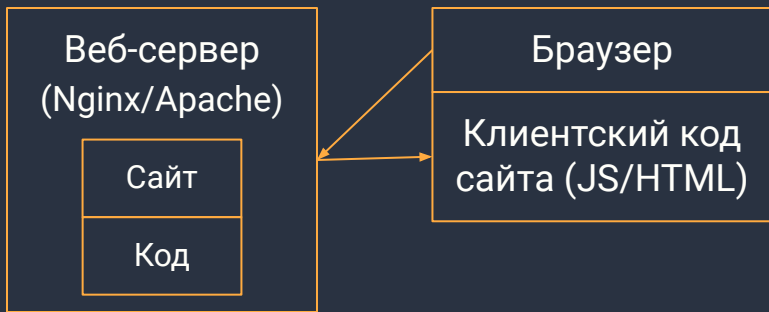
Код сайта. Клиентская часть (frontend)



Уязвимости: инъекции JS-кода, внедрение шаблонов, XSS

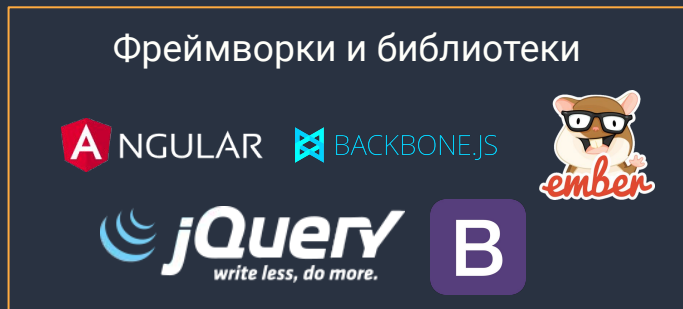


Код сайта. Клиентская часть (frontend)

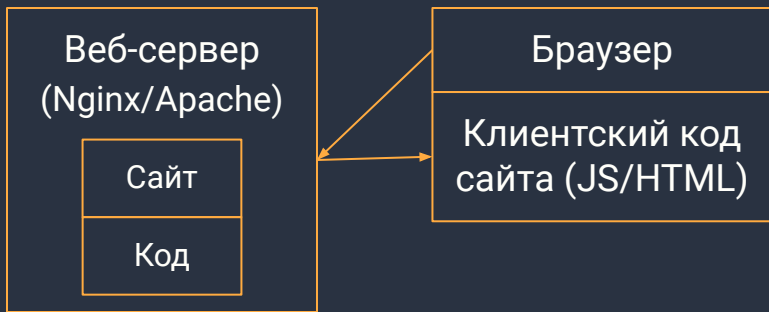


Уязвимости: инъекции JS-кода, внедрение шаблонов, XSS

Масштаб: специфичные для сайта, для уязвимостей в фреймворке - все сайты на нём



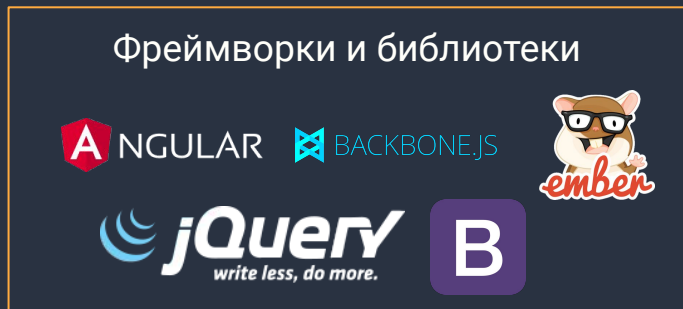
Код сайта. Клиентская часть (frontend)



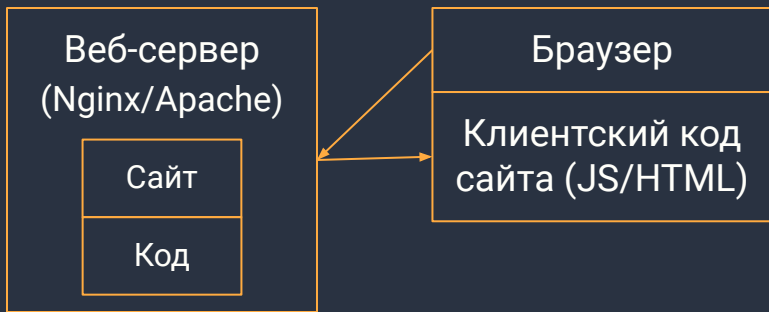
Уязвимости: инъекции JS-кода, внедрение шаблонов, XSS

Масштаб: специфичные для сайта, для уязвимостей в фреймворке - все сайты на нём

Частые атаки: кража аккаунтов через XSS, фишинг



Код сайта. Клиентская часть (frontend)



Уязвимости: инъекции JS-кода, внедрение шаблонов, XSS

Масштаб: специфичные для сайта, для уязвимостей в фреймворке - все сайты на нём

Частые атаки: кража аккаунтов через XSS, фишинг

Детектирование: сканеры XSS, ручной поиск подстановкой спец. символов шаблонов, например: `{{5*5}}` для Angular.



Литература и ссылки

- Дуглас Э. Камер - Сети TCP IP. Принципы, протоколы и структура (2003)
- Metasploitable 2 и 3 - системы для тренировки
 - <https://sourceforge.net/projects/metasploitable/>
 - <https://github.com/brimstone/metasploitable3/releases>
- Gordon Lyon. Nmap Network Scanning. The Official Nmap Project Guide to Network Discovery and Security Scanning (2011) (<http://nmap.org/book/>)
- Список скриптов Nmap:
<https://nmap.org/nsedoc/>