Введение

Курс посвящен оценке безопасности и защите компьютерных сетей. Разбору сетевых атак, уязвимостей, способов их поиска и эксплуатации. Носит практическую направленность. Из инструментов изучаются такие инструменты как metasploit, nmap, wireshark, iptables, snort. Имеется некоторое количество заданий в формате ctf.

Содержимое курса

1. Введение
2. Базовые сетевые протоколы и их безопасность
3. Сетевые атаки
4. Сетевые сканеры. Nmap
5. Системы текстового вторжения. Metasploit
6. Анализ трафика. Wireshark
7. Уязвимости Web-приложений
8. Межсетевые экраны. Iptables
9. Системы обнаружения и предотвращения вторжений. Snort
10. Криптографические протоколы. TLS
11. Виртуальные частные сети. OpenVPN
12. Безопасность беспроводных сетей. Aircrack
13. Песочницы. AppArmor

Основные понятия

Уязвимость представляет собой недостаток системы, который ведёт к нарушению её безопасности. Например, выполнить произвольных код или получить доступ к информации в обход системы защиты.

А **эксплойт** - это программа, которая позволяет реализовать атаку, используя уязвимость.

Аудит представляет собой процесс оценки безопасности системы. Это может быть тестирование на проникновение (**пентест**) или менее жёсткие способы проверки безопасности. Он отличается от взлома легальностью происходящего, так как выполняется с разрешения владельца ресурса.

Стадии сетевой атаки

Выделяют 4 основные стадии сетевой атаки:

1. сбор информации
2. анализ уязвимостей

3. эксплуатация
4. постэксплуатация.

При аудите добавляется составление отчёта.

Стадия **сбора информации** включает в себя поиск информации из различных источников таких как поисковые системы, соц.сети, DNS. Также может использоваться сетевое сканирование. Как правило на данном шаге определяется используемое на системе программное обеспечение и его версии, что упрощает последующую задачу анализа уязвимостей.

На стадии **анализа уязвимостей** производится поиск уязвимостей на цели. Для этого может осуществляться поиск по различным базам с уязвимостями и эксплойтами, разбор информации которая там содержится. Также на данном этапе может осуществляется самостоятельное нахождение уязвимостей, если система не содержит известных уязвимостей.

На этапе **эксплуатации** производится разработка / выбор эксплойта под уязвимость и осуществляется непосредственно атака на систему.

Постэксплуатация включает в себя продвижение далее по сети с использованием полученного доступа к системе. Т.е. идёт переход к стадии 1, но уже с повышенными привилегиями. Хакеры на данном этапе выполняют вредоносную нагрузку, например рассылку спама или сбор личной информации.

При аудите или пентесте, на последнем этапе составляется **отчёт** с описанием всех найденных уязвимостей.

Весь процесс может повторяться циклично, в зависимости от того, насколько хакер или пентестер желает проникнуть внутрь сети.

Базы уязвимостей

Множество примеров уязвимостей в программах можно найти в специальных базах, в которых ведётся их учёт.

Наиболее полная и известная база - это **CVE** (Common Vulnerabilities and Exposures) от корпорации MITRE. Доступна по адресу: cve.mitre.org. Все найденные уязвимости в программах регистрируются в данной базе с уникальным номером и списком ссылок с дополнительной информацией.

В базе **NVD** (National Vulnerabilities Database) собрана воедино в единой стандартизированной форме информация из различных классификаторов (CVE, CWE, CPE и другие). Поиск по базе осуществляется по адресу:

https://nvd.nist.gov/vuln/search

Вся база доступна в формате JSON (https://nvd.nist.gov/vuln/data-feeds) и может быть использована для написания собственных инструментов безопасности.

База NVD позволяет производить поиск имени программы и с недавних по процессорам и другому аппаратному обеспечению. Пример поиска уязвимостей в для браузера Google Chrome приведён на Рис.1, на котором содержится список найденных уязвимостей.

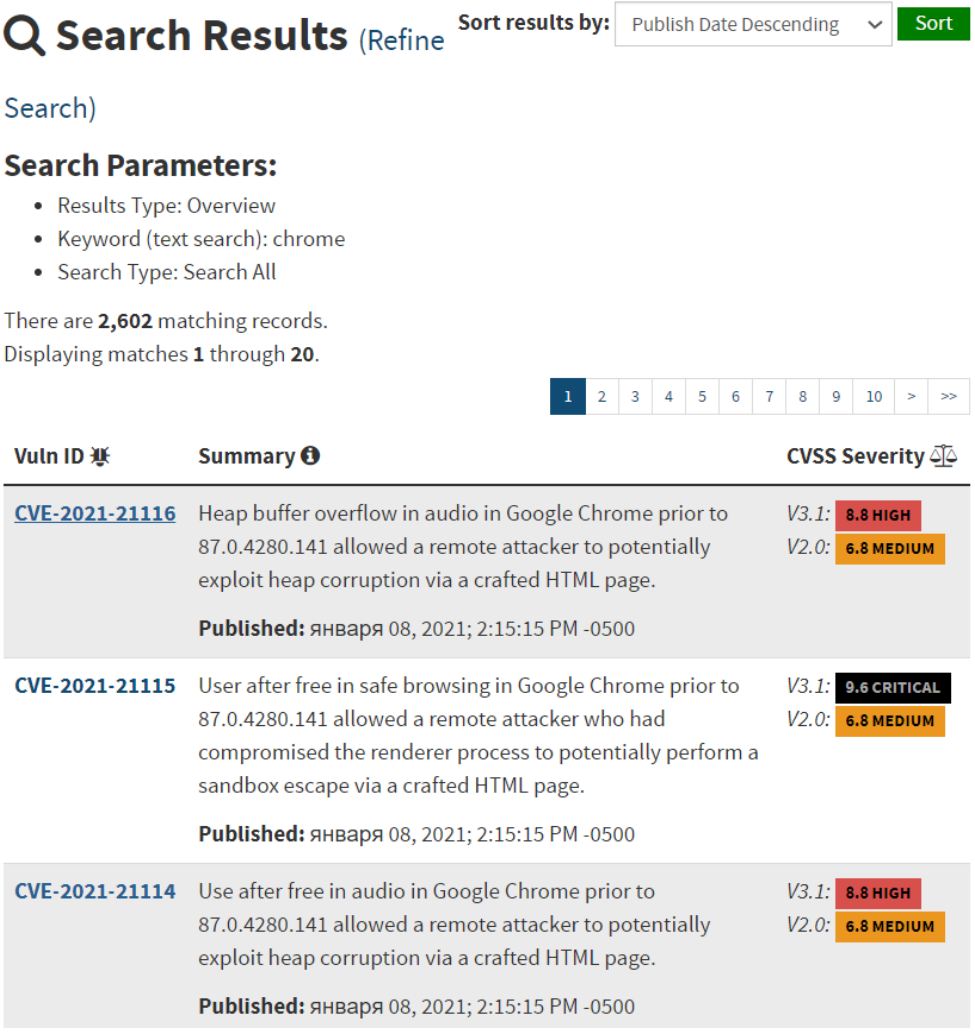


Рис. 1. Пример поиска уязвимостей для браузера Google Chrome.

К каждому результату имеется **краткое описание** вида:

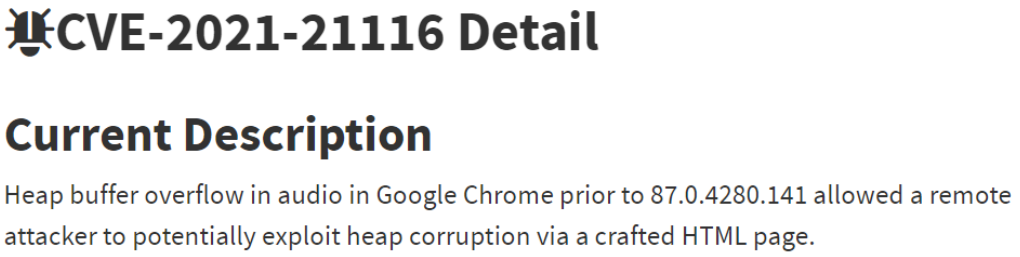


Рис.2. Краткое описание уязвимости

В данном случае имеем переполнение буфера в куче. Как видно из описания хакер может составить вредоносную страницу с воспроизведением звука и заманить на неё пользователя с помощью фишинга или другим способом. При этом переполнение буфера потенциально ведёт к выполнению произвольного кода, так позволяет перезаписывать адреса функций, по которым переходит программа по мере своего выполнения.

Для каждой уязвимости база содержит также **оценку опасности**:

Рис.3. Оценка уязвимости по стандарту CVSS.

Данная оценка учитывает наличие эксплойта в открытом доступе, простоту эксплуатации и другие факторы. Подробнее можно посмотреть в стандарте CVSS, по которому производится оценка.

Также для каждой уязвимости доступен **список ссылок** с дополнительной информацией.

Рис. 4. Список ссылок с дополнительной информацией в базе CVE/NVD.

Рабочие эксплойты для последних версий ПО выкладываются редко, поэтому для эксплуатации нужно разбираться в уязвимости подробнее. Полезным источником информации здесь являются системы контроля версий (Git, SVN,

CVN, Mercurial), в которых хранится исходный код программ со всей историей изменений. Здесь можно посмотреть как разработчики исправили уязвимость, используя для поиска соответствующего исправления информацию из базы CVE (например, номер ошибки в багтрекере может содержаться в описании коммита Git). Часто случается, что разработчики не закрывают уязвимости полностью, либо при исправлении добавляют другие ошибки, ведущие к уязвимостям. В качестве примера можно назвать серию уязвимостей Shellshock.

Кроме того, в базе NVD уязвимости группируются по классам с помощью **классификатора CWE**.

CWE-ID	CWE Name	Source
CWE-787	Out-of-bounds Write	NIST

Рис. 5. Информация о классе для уязвимости CVE-2021-21116 в Google Chrome.

Для нашей уязвимости в Google Chrome имеем переполнение буфера, которое представляет собой запись за границы массива. Заметим, что база CWE содержит в себе достаточно подробную информацию о классах уязвимостей с примерами (об этом далее), позволяющими разобраться в данном классе уязвимостей.

И последнее, что мы рассмотрим из содержимого базы NVD — это информацию о **версии или конфигурации ПО**.

cpe:2.3:a:google:chrome:*:*:*:*:*:*:*:* Show Matching CPE(s)	Up to (excluding) 87.0.4280.141

Рис. 6. Информация об уязвимых версиях ПО.

Данная информация представлена в едином формате (CPE), который позволяет производить поиск уязвимостей по базе с учётом (диапазона) версий. Соответственно, если мы знаем версию программы или в случае сайтов версию, скажем, PHP, мы можем использовать эту информацию для поиска незакрытых уязвимостей. Аналогичным образом мы можем проверить на уязвимости любую программу, а не только Google Chrome.

Базы эксплойтов

Как только мы нашли уязвимость в базе, мы можем попытаться найти эксплойт. Ссылки на эксплойты могут содержаться (редко) дополнительных ссылках базы CVE. Но наиболее полную информацию по эксплойтам чаще

всего можно найти в сторонних источниках (социальные сети, специализированные базы).

Наиболее полная и известная база уязвимостей — Exploit Database от Offensive Security — от разработчика Kali Linux. Доступна по адресу: www.exploit-db.com. Всю базу можно скачать себе на компьютер и пользоваться утилитой SearchSploit для локального поиска по ней. Большинство эксплойтов либо старые, либо из категории Proof-Of-Concept, т.е. они предназначенны для демонстрации наличия уязвимости, но для эксплуатирования уязвимости требуется доработка. Это и понятно, потому что распространение и использование работоспособных эксплойтов может подпадать под законодательные запреты многих стран, включая Россию (для данного случая смотри далее).

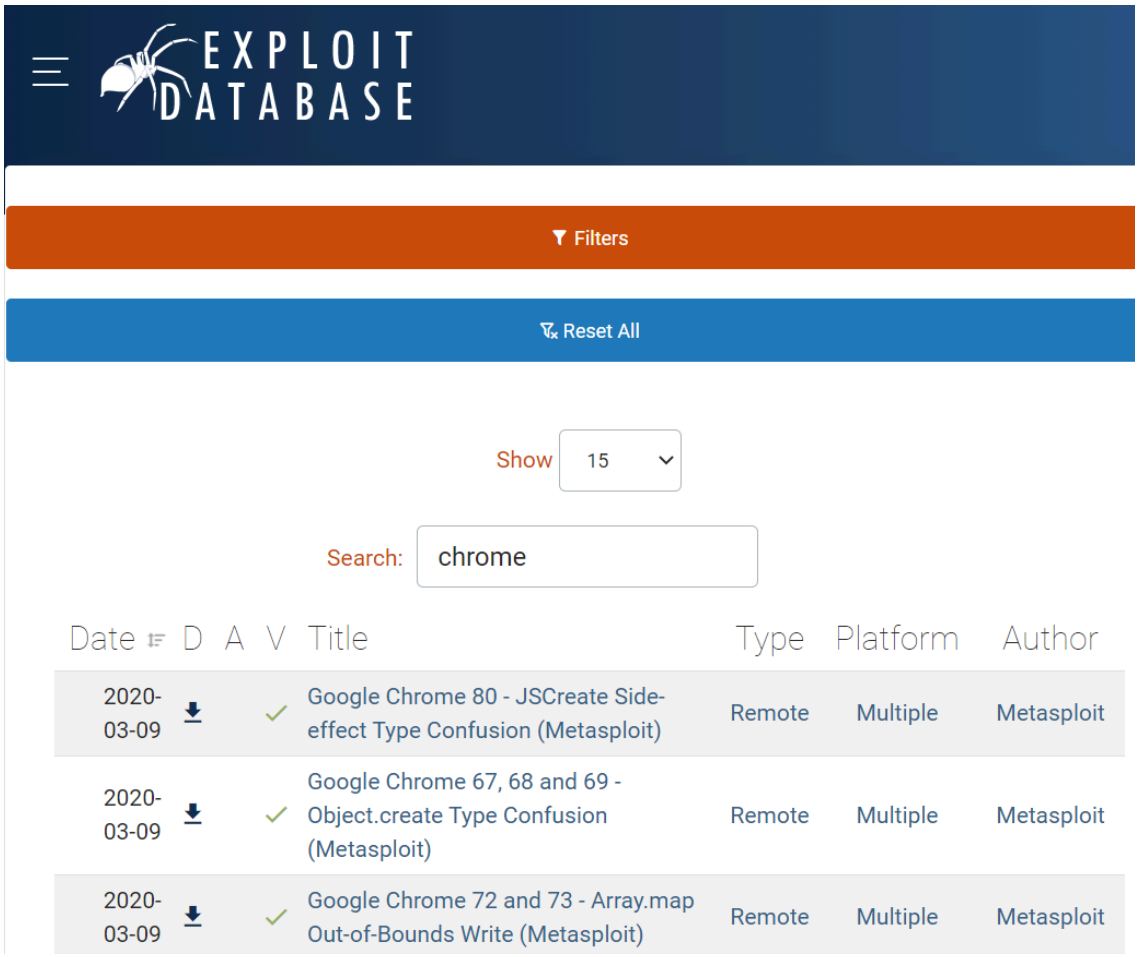


Рис. 7. Поиск по базе эксплойтов от Offensive Security.

Вероятность успеха атаки готовыми эксплойтами

Многие компании работают по принципу “работает не трогай” и могут не обновлять свои системы годами. Поэтому они часто становятся жертвами неопытных хакеров, которые запускают уже готовые эксплойты к старым уязвимостям и общедоступные сканеры безопасности для определения версий. Такие как Nmap или такой мощный комбайн как OpenVAS

Виды уязвимостей

Есть несколько классификаторов видов уязвимостей. Мы рассмотрим две из них — уже упоминавшуюся базу CWE и специализированную базу OWASP.

База база OWASP (Open Web Application Security Project) посвящена классам уязвимостей в веб-приложениях. Самые опасные и часто встречаемые виды веб-уязвимостей по версии OWASP Top-10 - это инъекции кода, некорректная аутентификация (как правило встречается её отсутствие для отдельных страниц). Также распространены утечки данных и ошибки конфигурации.

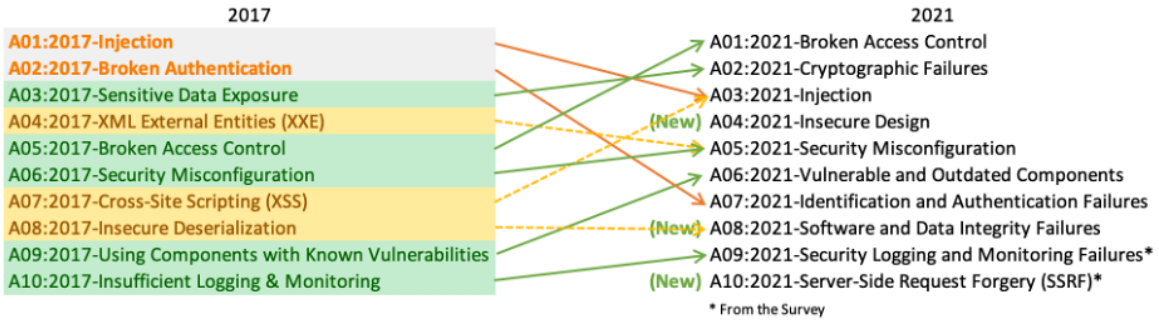


Рис. 8. Наиболее опасные классы уязвимостей по версии OWASP.
Источник: https://owasp.org/www-project-top-ten/

Наиболее полная общая база с классами уязвимостей — это CWE (Common Weaknes Enumeration). Список наиболее опасных классов уязвимостей представлен на Рис.9.

Большинство указанных классов уязвимостей как в базе OWASP, так и в из базы CWE мы будем подробно изучать в отдельном курсе по внешнему аудиту компьютерных сетей. А пока что необходимую краткую информацию о классах уязвимостей можно найти в данных двух базах.

Рассмотрим в качестве примера инъекции кода (CWE-94). По каждому классу уязвимостей есть краткое и подробное описание (Рис. 10), примеры уязвимого кода программ и скриптов (Рис. 11). А также примеры атаки. В данном случае у нас приложение содержит часть которая генерирует код по информации

предоставленной извне. При этом из-за ошибочной нейтрализации спецсимволов можно внедрить свой код. В примере из базы таким можно внедрить на выполнение любой php-код.

2024 CWE Top 25

Rank	ID	Name	Score	CVEs in KEV	Rank Change vs. 2023
1	CWE-79	Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')	56.92	3	+1
2	CWE-787	Out-of-bounds Write	45.20	18	-1
3	CWE-89	Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')	35.88	4	0
4	CWE-352	Cross-Site Request Forgery (CSRF)	19.57	0	+5
5	CWE-22	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal')	12.74	4	+3
6	CWE-125	Out-of-bounds Read	11.42	3	+1
7	CWE-78	Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection')	11.30	5	-2
8	CWE-416	Use After Free	10.19	5	-4
9	CWE-862	Missing Authorization	10.11	0	+2
10	CWE-434	Unrestricted Upload of File with Dangerous Type	10.03	0	0
11	CWE-94	Improper Control of Generation of Code ('Code Injection')	7.13	7	+12
12	CWE-20	Improper Input Validation	6.78	1	-6
13	CWE-77	Improper Neutralization of Special Elements used in a Command ('Command Injection')	6.74	4	+3
14	CWE-287	Improper Authentication	5.94	4	-1
15	CWE-269	Improper Privilege Management	5.22	0	+7
16	CWE-502	Deserialization of Untrusted Data	5.07	5	-1
17	CWE-200	Exposure of Sensitive Information to an Unauthorized Actor	5.07	0	+13
18	CWE-863	Incorrect Authorization	4.05	2	+6
19	CWE-918	Server-Side Request Forgery (SSRF)	4.05	2	0
20	CWE-119	Improper Restriction of Operations within the Bounds of a Memory Buffer	3.69	2	-3
21	CWE-476	NULL Pointer Dereference	3.58	0	-9
22	CWE-798	Use of Hard-coded Credentials	3.46	2	-4
23	CWE-190	Integer Overflow or Wraparound	3.37	3	-9
24	CWE-400	Uncontrolled Resource Consumption	3.23	0	+13
25	CWE-306	Missing Authentication for Critical Function	2.73	5	-5

Рис. 9. Список наиболее опасных классов уязвимостей по версии MITRE.

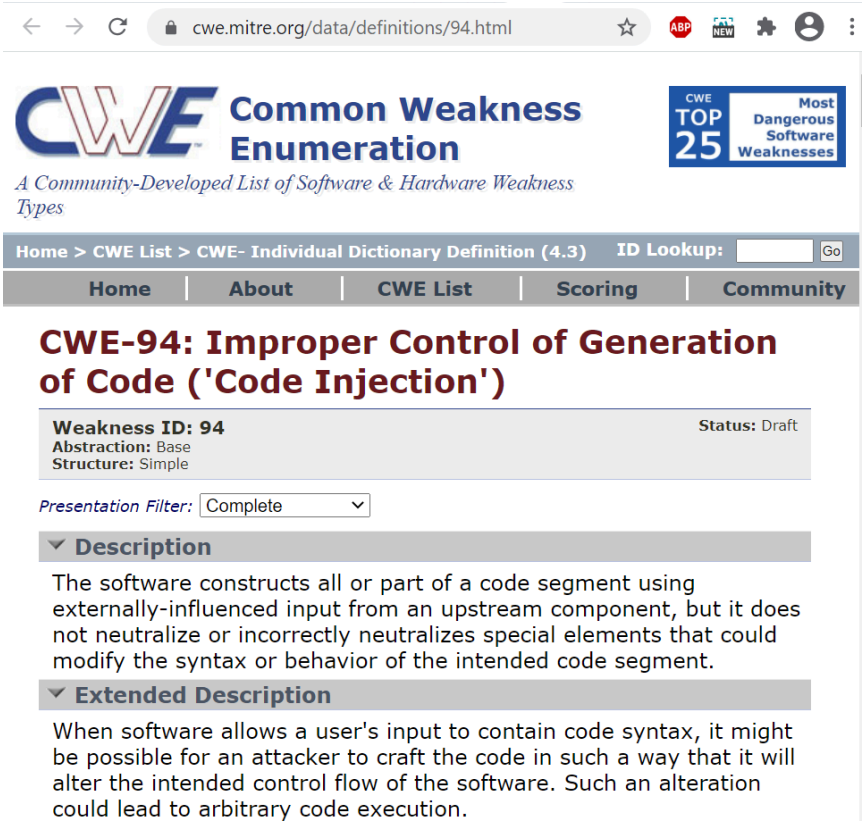


Рис. 10. Описание класса уязвимости.

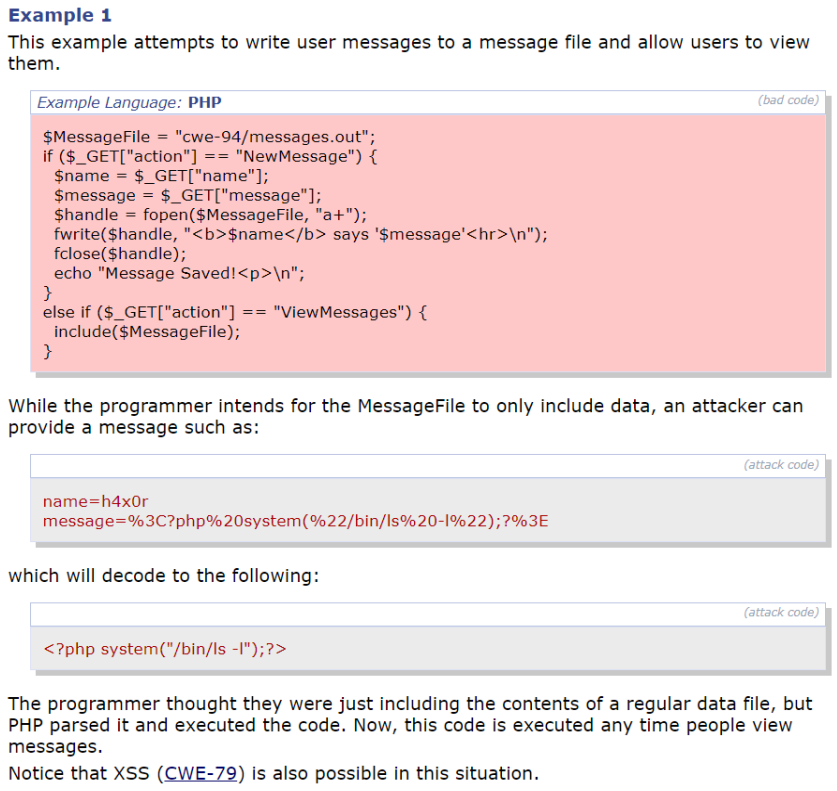


Рис. 11. Примеры кода с разбором.

Инструментарий

В дальнейшей работе будем использовать следующие инструменты.

Metasploit - система для проведения тестовых вторжений и разработки эксплойтов.

Nmap - сетевой сканер, который может использоваться для определения версий сервисов с последующим поиском по базе уязвимостей.

Wireshark в качестве анализатора сетевого трафика, в котором можно посмотреть, какие пакеты проходят через ваш компьютер.

Все эти инструменты и многие другие есть в составе Kali Linux. Это дистрибутив Linux с большим количеством инструментов для тестирования безопасности. Рекомендуется скачать виртуальную машину с ним и использовать. Заметим, что этот дистрибутив предназначен для тестирования безопасности и для использования в качестве основной системы для работы не предназначен.

Поисковые системы

Из поисковых систем для поиска информации о целевой системе может использоваться Google для этого можно составлять специальные поисковые запросы, которые называются Google Dorks. База таких запросов есть доступна по адресу:

- https://www.exploit-db.com/google-hacking-database

Также есть специализированные поисковые системы, такие как Shodan. Которые собирают информацию об устройствах в сети. Например, на на Рис. 12 справа представлена карта устройств в интернете (включая сервера, веб-камеры), которые отвечают на ping-запрос.

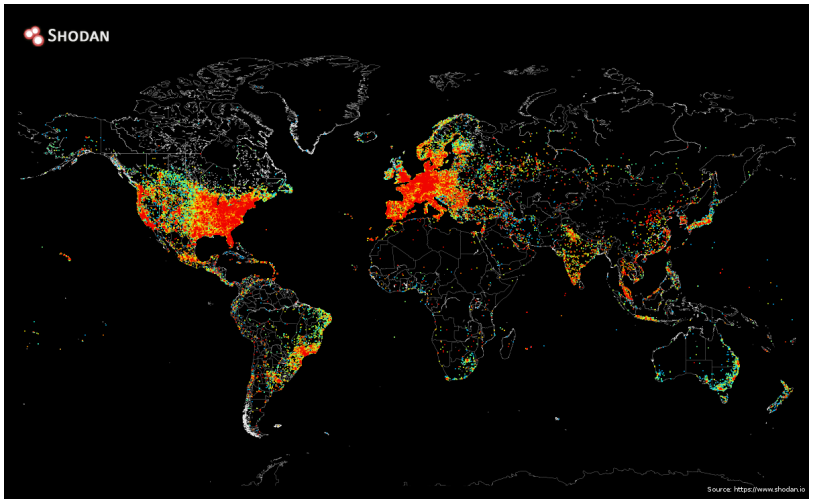


Рис. 12. Ping-карта интернета

В Shodan также доступна фильтрация по разным параметрам.

Filter Name	Description	Example
city	Name of the city	Devices in San Diego
country	2-letter Country code	Open ports in the United States
http.title	Title of the website	"Hacked" Websites
net	Network range or IP in CIDR notation	Services in the range of 8.8.0.0 to 8.8.255.255
org	Name of the organization that owns the IP space	Devices at Google
port	Port number for the service that is running	SSH servers
product	Name of the software that is powering the service	Samsung Smart TVs
screenshot.label	Label that describes the content of the image	Screenshots of Industrial Control Systems
state	U.S. State	Devices in Texas

Рис. 13. Shodan. Основные фильтры

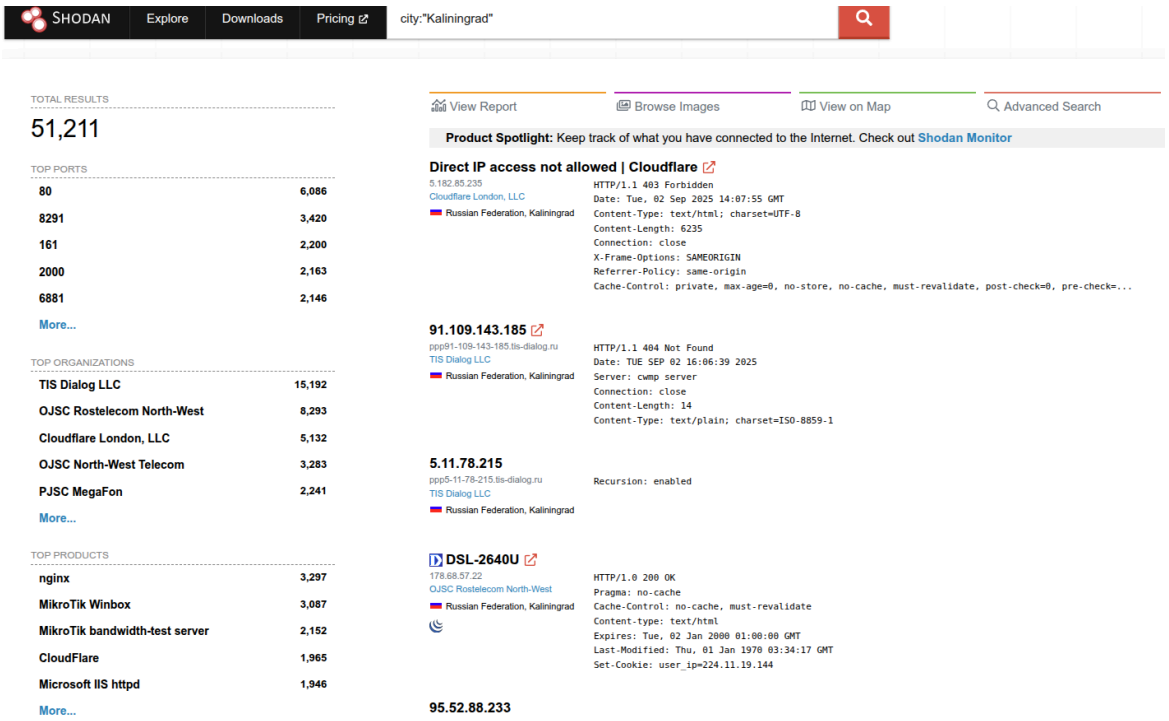


Рис. 14. Shodan. Пример. Список устройств в Калининграде

Ресурсы для тренировки

Для тренировки можно использовать машины с уязвимостями с ресурсов HackTheBox или VulnHub.

HackTheBox (https://www.hackthebox.com/). Ресурс содержит в себе множество как платных так и бесплатных виртуальных машин для тренировки. Причём данные машины не нужно разворачивать на своём компьютере - ресурс запускает их сам на своих серверах, предоставляя доступ для тренировки. По-началу для регистрации на HackTheBox требовалось решить небольшую задачу на взлом, но затем такую возможность убрали. Машины отсортированы по уровню сложности.

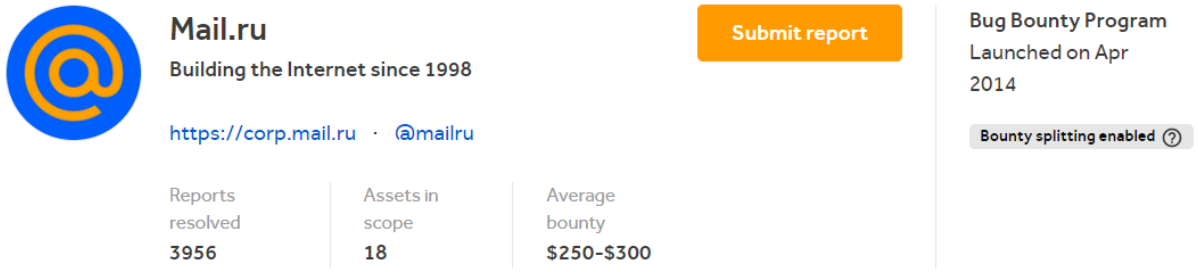
VulnHub (https://www.vulnhub.com/). Машины с VulnHub нужно скачивать на свой компьютер и запускать через VirtualBox. Здесь также есть сортировка по уровню сложности.

Программы Bug Bounty

В качестве варианта для легального применения навыков пентеста на практике есть программы Bug Bounty, предназначенные для оплаты за уязвимости. Крупнейшим агрегатором таких программ является ресурс

HackerOne, на котором представлена информация по программам со всего мира и который предоставляет платформу для сообщения об уязвимостях и оплаты за них. Из российских компаний представлены Mail.ru, Ozon, Yandex, VK. В настоящее время (2023 г.) выплаты исследователям безопасности из России не платформе HackerOne не доступны. Поэтому следует использовать альтернативные площадки, например bugbounty.standoff365.com от Positive Technologies, либо программы от крупных компаний, таких как Яндекс (https://yandex.ru/bugbounty/index).

Одна из самых крупных программ - у компании Mail.ru (https://bugbounty.standoff365.com/programs/mail_vk). Всего по ней выплачено наград более чем на 124 миллиона рублей.



Самые дорогие найденные уязвимости - это удаленное выполнение кода, SQL-инъекции и доступ к локальным файлам в обход ограничений. Самые дешёвые - XSS.

Mail.ru authentication center, mail, messaging, cloud services, portal, content and news projects:

Vulnerability	Main Scope	MCS	ICQ	Content
Remote code execution (RCE)	$35000	$25000	$15000	$20000
Injections (SQLi or equivalent)	$25000	$20000	$10000	$10000
Local files access and manipulation (LFR, RFI, XXE) without jail/chroot/file type restrictions	$25000	$20000	$10000	$10000
RCE in standalone isolated / virtualized single-purpose process (e.g. image conversion)	$5000	$15000	$5000	$5000
SSRF, non-blind (with ability to read reply text), except dedicated proxies	$10000	$15000	$5000	$7500
SSRF, blind, except dedicated proxies	$2000	$2000	$2000	$1500

Правовые вопросы

⚠ С точки зрения закона многое из деятельности тестировщика безопасности подпадает под статьи как в России так и за рубежом. Причём сами статьи довольно размыты. Тестирование на безопасность без разрешения карается статьёй 272. Неправомерный доступ к компьютерной информации. Для тренировки используйте специально предназначенные ресурсы по типу HackTheBox или VulnHub.

В программах поиска уязвимостей следует внимательно читать условия и не выходить за границы правил.

Разработка и распространение эксплойтов подпадает под статью 273 создание и распространение вредоносных программ. Похоже, что PoC-эксплойты, которые предназначены только для демонстрации уязвимости, более-менее безопасно разрабатывать, при условии, что они не позволяют хакеру собственно взломать систему.

Литература и ссылки

1. Яворски П. - Ловушка для багов. Полевое руководство по веб-хакингу (2020)
2. Яворски П. - Основы веб-хакинга (2016)
3. Эриксон Д. - Хакинг: искусство эксплойта. 2 изд. (2018)
4. Курс по Metasploit (https://www.offensive-security.com/metasploit-unleashed/)
5. Журнал Хакер (https://xakep.ru/)