Л38. Автоматическое сканирование уязвимостей - Nessus, OpenVAS, Lynis, SPARTA. Развертывание Eve-ng.

Сканирование уязвимостей — процесс выявления и анализа критических недостатков безопасности в целевой среде. Иногда эту операцию называют оценкой уязвимости. Сканирование уязвимостей — одна из основных задач программы выявления и устранения этих недостатков. С его помощью можно проанализировать все элементы управления безопасностью ИТ-инфраструктуры. Сканирование уязвимостей производится после того, как мы обнаружили, собрали и перечислили информацию об инфраструктуре целевой системы. Информация, полученная после сканирования системы на уязвимости, может привести к компрометации целевой системы, нарушению ее целостности и конфиденциальности.

В.этой.главе.излагаются.следующие.темы.

- Понятия двух общих типов уязвимостей: локальные и удаленные.
- Классификация уязвимостей, указывающая на отраслевой стандарт, который может быть использован для систематизации любой уязвимости и распределения в соответствии с определенными признаками.
- Знакомство с несколькими инструментами для поиска и анализа уязвимостей, присутствующих в целевой среде. Представленные инструменты распределены в соответствии с их основными функциями, связанными с оценкой безопасности. Это такие инструменты, как Nessus, Cisco, SMB, SNMP и средства анализа веб-приложений.

Обратите внимание, что независимо от того, тестируем мы внешнюю или внутреннюю сеть, ручные и автоматизированные процедуры оценки уязвимостей должны использоваться поровну. Если задействовать только автоматический режим, можно получить большое количество ложных срабатываний и отрицаний. Кроме того, очень важна теоретическая подготовка испытателя на проникновение, а также то, насколько хорошо он знает инструменты, с помощью которых будет выполняться тест. Аудитору постоянно нужно совершенствовать свои знания и навыки.

И еще один очень важный момент: автоматизированная оценка уязвимости не является окончательным решением. Бывают ситуации, когда автоматизированные

средства не могут определить логические ошибки, скрытые уязвимости, неопубликованные уязвимости программного обеспечения и человеческий фактор, влияющий на безопасность. Поэтому рекомендуется использовать комплексный подход, предусматривающий применение как автоматизированных, так и ручных методов оценки уязвимости. Это повысит успешность тестов на проникновение и предоставит наиболее объективную информацию для исправления уязвимостей.

Технические требования

Ноутбук или настольный компьютер с объемом оперативной памяти не менее 6 Гбайт, четырехъядерный процессор и 500 Гбайт места на жестком диске. В качестве операционной системы мы используем Kali Linux 2018.2 или 2018.3 (как виртуальную машину или систему, установленную на жестком диске, SD-карте или USB-накопителе).

Типы уязвимостей

Существует три основных категории уязвимостей, которые, в свою очередь, можно разделить на локальные и удаленные. Это уязвимости, допущенные при разработке программного обеспечения, ошибки при реализации программного обеспечения и уязвимости, обнаруживаемые при эксплуатации системы.

- *Уязвимости при разработке* — обнаруживаются из-за недостатков в технических требованиях к программному обеспечению.
- *Уязвимости реализации* — технические ошибки безопасности, найденные в коде системы.
- *Эксплуатационные уязвимости* — уязвимости, которые могут возникнуть из-за неправильной настройки и разворачивания системы в целевой среде.

На основе этих трех классов у нас есть два общих типа уязвимостей: локальные и удаленные, которые могут появиться в любой категории описанных уязвимостей.

Локальные уязвимости

Если злоумышленник получает доступ, выполняя часть кода, это называется *локальной уязвимостью*. Воспользовавшись данной уязвимостью, злоумышленник может повысить свои права доступа и получить неограниченный доступ к компьютеру.

Рассмотрим пример, в котором злоумышленник Боб имеет локальный доступ к системе, работающей под управлением Windows Server 2008 (32-разрядная платформа x86). Доступ ему был ограничен администратором при реализации политики безопасности, в результате чего Бобу стало недоступно определенное приложение. В экстремальных условиях Боб обнаружил, что с помощью вредоносного фраг-

мента кода он может получить доступ к компьютеру на уровне системы или ядра. Воспользовавшись хорошо известной уязвимостью (например, CVE-2013-0232, GP Trap Handler nt!KiTrap0D), он повысил свои права доступа, что позволило ему выполнять все административные задачи и получать неограниченный доступ к приложению. Это ясно показывает нам, как злоумышленник воспользовался уязвимостью для получения несанкционированного доступа к системе.

Дополнительные сведения об уязвимости CVE-2013-0232 MS и повышении прав доступа в Windows можно найти по адресу http://www.exploit-db.com/exploits/11199/.

Удаленная уязвимость

Удаленная уязвимость — это состояние, при котором злоумышленник еще не имеет доступа, но может его получить, запустив вредоносную часть кода через сеть. Этот тип уязвимости позволяет злоумышленнику получить удаленный доступ к компьютеру без каких-либо физических или локальных барьеров.

Например, Боб и Алиса подключены к Интернету с разных устройств. У них разные IP-адреса, да и живут они в разных местах. Предположим, компьютер Алисы работает под управлением операционной системы Windows XP и содержит секретную биотехнологическую информацию, а Бобу известен IP-адрес машины Алисы и то, какая операционная система установлена на этом компьютере. Боб ищет решение, которое позволит ему получить удаленный доступ к компьютеру Алисы. Со временем он узнает, что уязвимость службы Windows Server MS08-067 может быть легко использована удаленно на компьютере под управлением операционной системы Windows XP. Затем он запускает эксплойт против компьютера Алисы и получает к машине полный доступ.

Дополнительные сведения об уязвимости службы MS08-067 MS Windows Server можно найти по адресу http://www.exploit-db.com/exploits/6841/.

Систематизация уязвимостей

С увеличением количества доступных технологий за последние несколько лет предпринимались различные попытки ввести наиболее удобную классификацию, чтобы распределить по категориям все возможные уязвимости. Тем не менее не все распространенные ошибки кодирования, которые могут повлиять на безопасность системы, удалось классифицировать. Это связано с тем, что каждая уязвимость может относиться к нескольким категориям или классам. Кроме того, каждая системная платформа имеет собственную подключаемую базу уязвимостей, сложности с расширением и в результате — сложности взаимодействия с внешней средой.

В следующей таблице мы представляем вам стандарты таксономии (классификации и систематизации), которые помогут вам по возможности определить большинство распространенных сбоев безопасности. Обратите внимание: почти все из этих стандартов уже реализованы в ряде инструментов оценки безопасности. Данные инструменты позволяют изучать проблемы безопасности программного обеспечения в режиме реального времени.

Стандарт безопасности	Ссылка на ресурс
Seven pernicious kingdoms (Семь пагубных царств)	http://www.cigital.com/papers/download/bsi11-taxonomy.pdf
Common weakness enumeration (Перечисление общих недостатков)	http://cwe.mitre.org/data/index.html
OWASP Top 10	http://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project
Klocwork (Часовой механизм)	http://www.klocwork.com/products/documentation/Insight-.1/Taxonomy
WASC threat classification (Классификация угроз WASC)	http://projects.webappsec.org/Threat-Classification

Основная функция каждого из этих стандартов безопасности (таксономий) заключается в систематизации категорий и классов уязвимостей, которые могут использовать специалисты по безопасности и разработчики программного обеспечения для выявления конкретных ошибок. Обратите внимание: ни один такой стандарт безопасности не может считаться точным и полным.

Автоматическое сканирование уязвимостей

Испытатели на проникновение очень осторожно относятся к автоматическому сканированию уязвимостей и иногда говорят, что это просто мошенничество. Хотя, если мало времени, автоматические сканеры уязвимостей могут помочь получить большой объем информации о целевой сети.

Nessus 7

Tenable's Nessus был разработан два десятилетия назад и до сих пор остается очень популярным инструментом оценки уязвимостей. На Nessus можно подписаться на год. Однако хорошие люди в Tenable создали седьмую версию Nessus Professional и предлагают пробную версию всем, кто желает с нею ознакомиться.

Перед установкой вам необходимо узнать, какая версия Kali Linux установлена на вашем компьютере. Это поможет вам скачать ту версию Nessus, которая будет без сбоев работать с вашей операционной системой.

Введите в командную строку терминала команду `uname –a` (рис. 6.1).

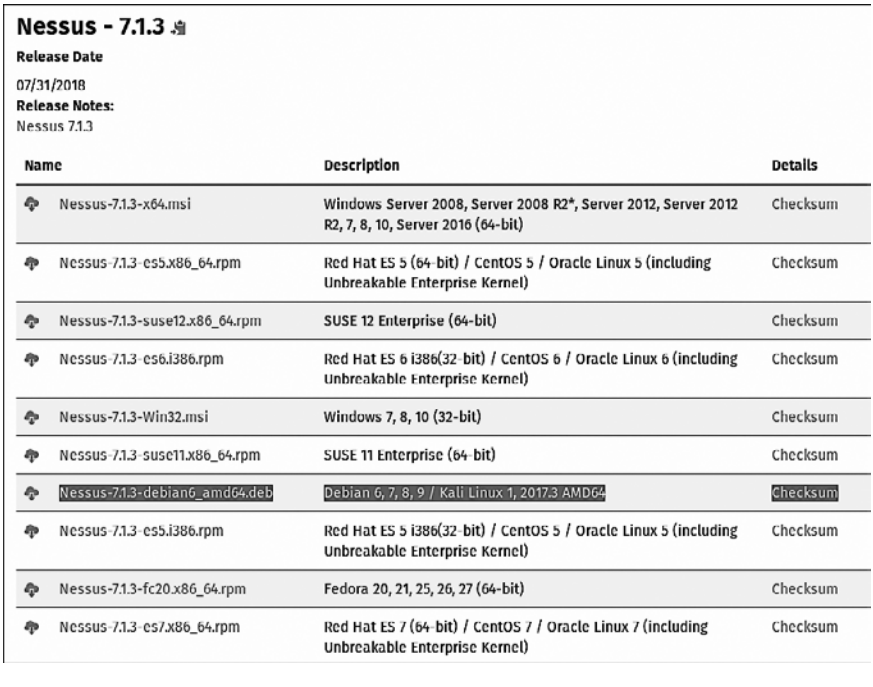
Рис. 6.1. Проверка версии Kali Linux

На рис. 6.1 видно, что я использую 64-разрядную версию (amd64) Kali Linux на основе Debian. Таким образом, мне нужно будет загрузить 64-разрядную версию Nessus, предназначенную для сборок Debian.

Установка сканера уязвимостей Nessus. Чтобы установить Nessus в Kali Linux, откройте браузер и перейдите на страницу Nessus (https://www.tenable.com/try).

Ознакомительная версия поставляется со всеми функциями полной версии, за исключением ограничений 16-IP.

Чтобы получить пробную версию, вам потребуется зарегистрироваться в Tenable. По электронной почте вы получите код подтверждения. После получения письма с кодом подтверждения вы можете скачать нужную версию Nessus в Kali Linux (рис. 6.2).



Nessus - 7.1.3

Release Date

07/31/2018

Release Notes:

Nessus 7.1.3

Name	Description	Details
Nessus-7.1.3-x64.msi	Windows Server 2008, Server 2008 R2*, Server 2012, Server 2012 R2, 7, 8, 10, Server 2016 (64-bit)	Checksum
Nessus-7.1.3-es5.x86_64.rpm	Red Hat ES 5 (64-bit) / CentOS 5 / Oracle Linux 5 (including Unbreakable Enterprise Kernel)	Checksum
Nessus-7.1.3-suse12.x86_64.rpm	SUSE 12 Enterprise (64-bit)	Checksum
Nessus-7.1.3-es6.i386.rpm	Red Hat ES 6 i386(32-bit) / CentOS 6 / Oracle Linux 6 (including Unbreakable Enterprise Kernel)	Checksum
Nessus-7.1.3-Win32.msi	Windows 7, 8, 10 (32-bit)	Checksum
Nessus-7.1.3-suse11.x86_64.rpm	SUSE 11 Enterprise (64-bit)	Checksum
Nessus-7.1.3-debian6_amd64.deb	Debian 6, 7, 8, 9 / Kali Linux 1, 2017.3 AMD64	Checksum
Nessus-7.1.3-es5.i386.rpm	Red Hat ES 5 i386(32-bit) / CentOS 5 / Oracle Linux 5 (including Unbreakable Enterprise Kernel)	Checksum
Nessus-7.1.3-fc20.x86_64.rpm	Fedora 20, 21, 25, 26, 27 (64-bit)	Checksum
Nessus-7.1.3-es7.x86_64.rpm	Red Hat ES 7 (64-bit) / CentOS 7 / Oracle Linux 7 (including Unbreakable Enterprise Kernel)	Checksum

Рис. 6.2. Предлагаемые для скачивания версии Nessus

Выберите версию Nessus, соответствующую вашей операционной системе. Чтобы согласиться с условиями использования, нажмите кнопку Accept (Принять). Далее в появившемся диалоговом окне нажмите кнопку Save File (Сохранить файл). Файл будет сохранен на вашем компьютере в папке `Downloads`. Мы для этого примера загрузили 64-битную версию Nessus (`Nessus-7.1.3-debian6_amd64.deb`).

После того как загрузка будет завершена, запустите новый терминал и перейдите в каталог загрузок. Для этого введите в командную строку команду `cd Downloads`. Далее просмотрите содержимое каталога, введя команду `ls`. С помощью этого действия вы можете убедиться, что файл действительно скачан и сохранен в целевой папке. Кроме того, вы можете скопировать имя установочного файла, чтобы вставить его в следующую команду. Далее, чтобы установить Nessus, введите команду `dpkg –i Nessus-7.1.3-debian6_amd64.deb`, как показано на рис. 6.3.

```
root@kali:~# cd Downloads
root@kali:~/Downloads# ls
Nessus-7.1.3-debian6_amd64.deb
root@kali:~/Downloads# dpkg -i Nessus-7.1.3-debian6_amd64.deb
```

Рис. 6.3. Установка Nessus

Если будут доступны новые версии Nessus, для выполнения команды dpkg -i скопируйте имя скачанного файла загрузки и его версию.

Не выходя из каталога `Downloads`, запустите службу Nessus. Для этого введите команду `service nessusd start`. При появлении следующего запроса укажите пароль для Kali Linux (рис. 6.4).

```
root@kali:~/Downloads# service nessusd start
Enter Auth Password: ****
root@kali:~/Downloads#
```

Рис. 6.4. Запуск службы Nessus

Для работы с Nessus откройте браузер, введите в адресную строку https://localhost:8834 и нажмите клавишу Enter. Когда появится баннер с предупреждением об опасности, нажмите кнопку Advanced (Дополнительно), далее нажмите кнопку Add Exception (Добавить исключение) и в конце — кнопку Confirm Security Exception (Подтвердить исключение безопасности) (рис. 6.5).

Для продолжения запуска службы выполните следующие шаги.

1. Создайте сначала учетную запись, указав имя пользователя и свой аккаунт, после чего нажмите кнопку Continue (Продолжить).

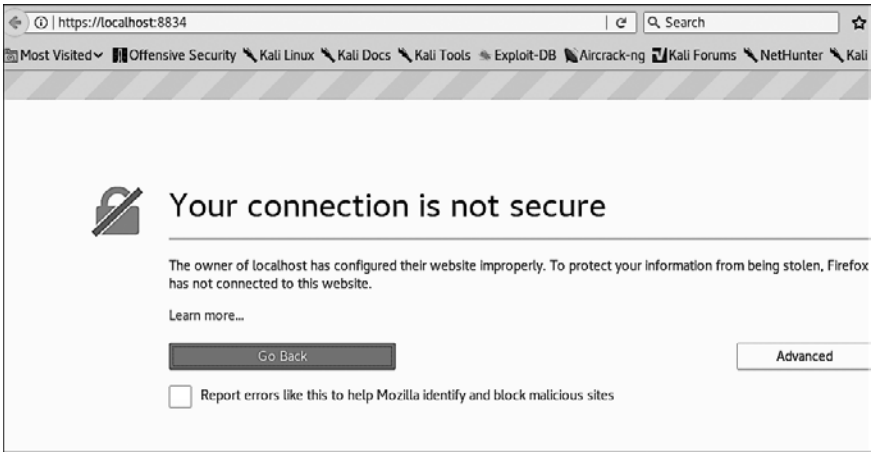


Рис. 6.5. Добавление исключения

2. Оставьте предлагаемые по умолчанию настройки Home, Professional или Manager, введите в поле ввода Activation Code (Код активации) полученный по электронной почте код активации и нажмите кнопку Continue (Продолжить).

   Если все пройдет удачно, Nessus начнет инициализацию, загрузит и скомпилирует необходимые плагины (рис. 6.6).

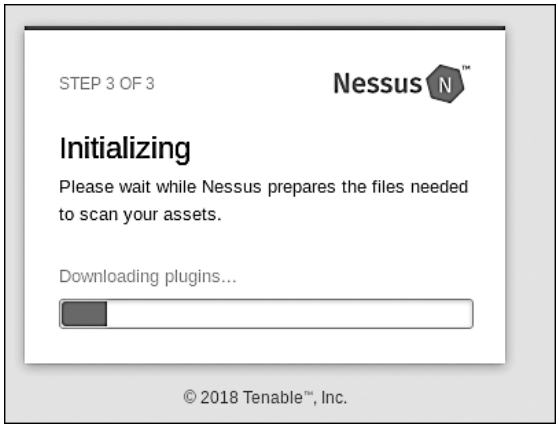


Рис. 6.6. Инициализация Nessus

В зависимости от скорости вашего соединения с Интернетом данная процедура может продлиться некоторое время. Пока будет происходить установка, можете зайти на сайт www.packtpub.com и посмотреть еще книги от Packt Publishing по Kali Linux и по тестированию на проникновение.

После завершения всех обновлений будет загружен интерфейс Nessus. Нажмите расположенную в правом верхнем углу кнопку New Scan (Новое сканирование), чтобы просмотреть все доступные типы сканирования (рис. 6.7).

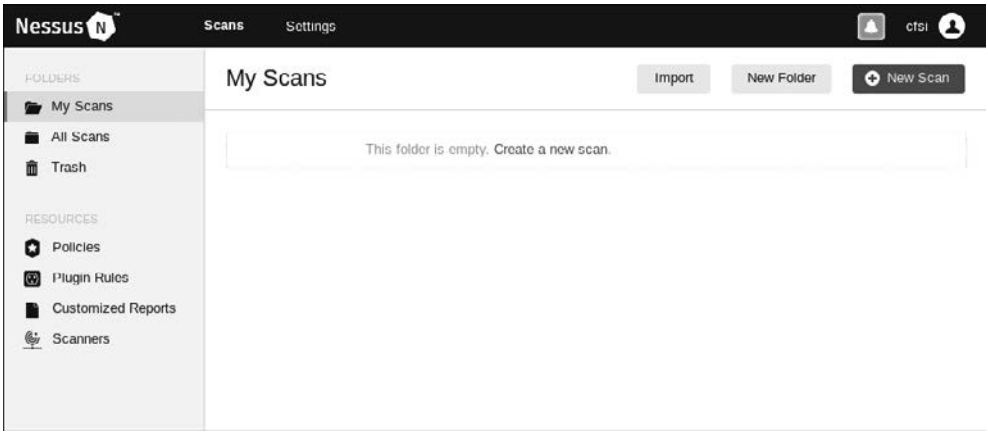


Рис. 6.7. Доступные виды сканирования

Здесь предлагается для использования большое количество шаблонов сканирования. Есть несколько шаблонов, которые доступны только по платной подписке. Кроме обнаружения узлов и расширенного сканирования, Nessus проводит расширенное сканирование уязвимостей, в том числе следующих типов.

- Сканирование облачной инфраструктуры.
- Локальное и удаленное сканирование обнаруженных поврежденных оболочек.
- Сканирование внутренней сети PCI.
- Сканирование вредоносных программ Linux и Windows.
- Сканирование Meltdown и Spectre.
- Сканирование программ-вымогателей WannaCry.
- Сканирование веб-уязвимостей.

Некоторые из них показаны на рис. 6.8.

Чтобы продемонстрировать обнаружение уязвимостей, воспользуемся уязвимым веб-сервером Linux. В главе 2 мы рассказывали, как настроить Metasploitable 2, Metasploitable 3, очень уязвимую систему Linux и BadStore.

В окне сканера щелкните на шаблоне Advanced Scan (Расширенное сканирование) и в разделе BASIC (Основное) заполните поля ввода. В поле Targets (Цели) укажите IP-адрес целевой машины или диапазон IP-адресов целевых машин, которые должны быть просканированы с помощью шаблона расширенного сканирования (рис. 6.9).

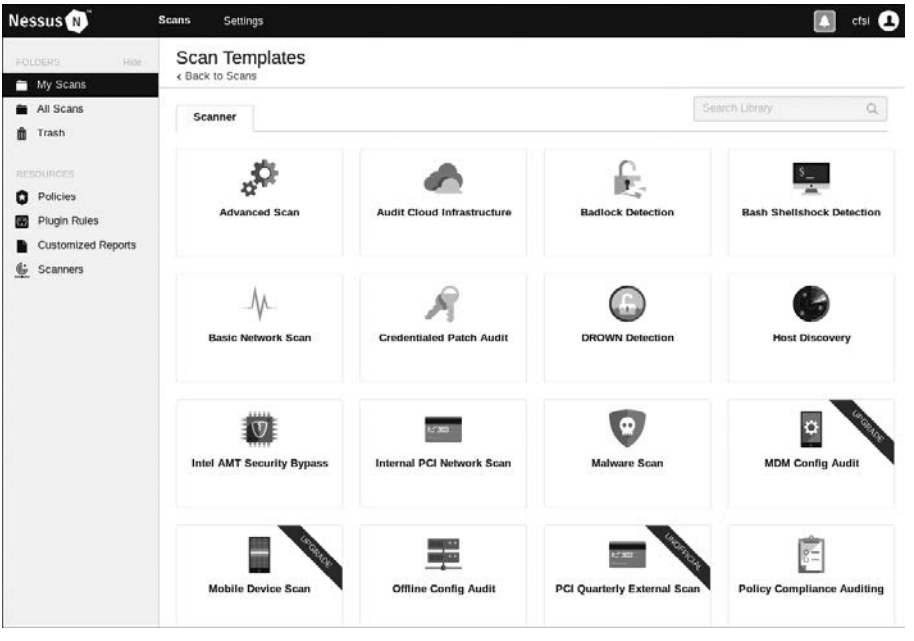


Рис. 6.8. Некоторые из видов сканирования

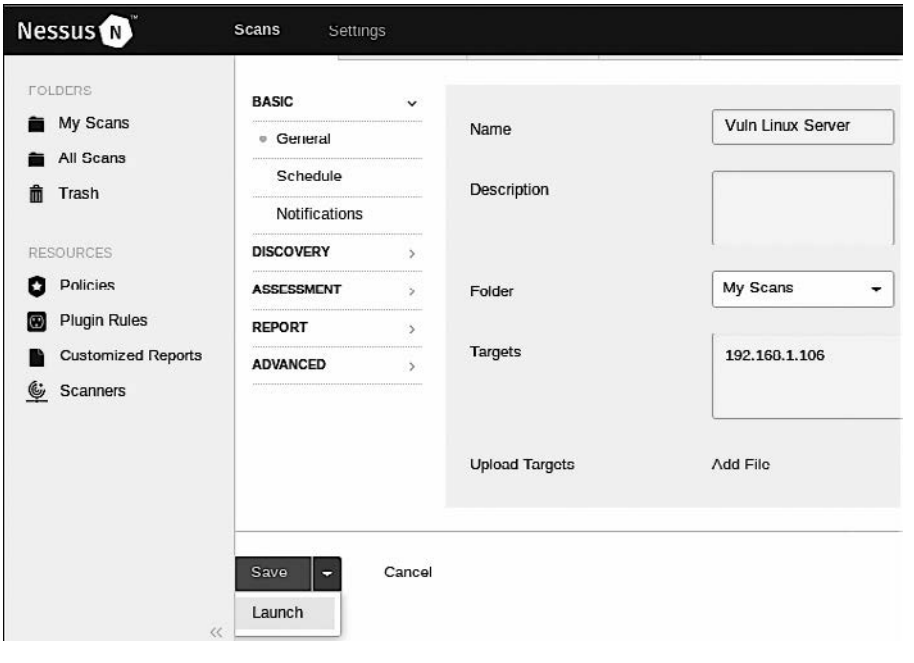


Рис. 6.9. Указываем цели

Поскольку предлагается несколько различных настроек, изучите другие разделы левого столбца. Каждый из этих разделов позволяет настроить сканирование в соответствии с конкретными требованиями.

- DISCOVERY (Открытие). Nessus использует ряд различных методов для обнаружения действующих в данное время хостов. Здесь можно задать определенные параметры для их обнаружения.
- ASSESSMENT (Оценивание). Здесь вы можете задать тип и глубину сканирования.
- REPORTING (Отчетность). При подготовке отчета о тестировании на проникновение важно иметь подробную информацию о проверке уязвимостей. Эта функция позволяет задать параметры отчетов.
- ADVANCED (Дополнительно). Расширенные настройки позволяют изменять не только количество сканируемых хостов, но и другие параметры синхронизации.

После настройки сканирования можно выбрать команду Save (Сохранить) или Launch (Запустить). Вы увидите список My Scan (Мои сканирования).

Щелкните кнопкой мыши на значке Play (Воспроизведение), который расположен справа от имени шаблона сканирования. Будет запущено сканирование. Если щелкнуть на имени шаблона сканирования во время проведения теста, на экране вы увидите общую информацию о сканируемой целевой машине и об уязвимости (рис. 6.10).

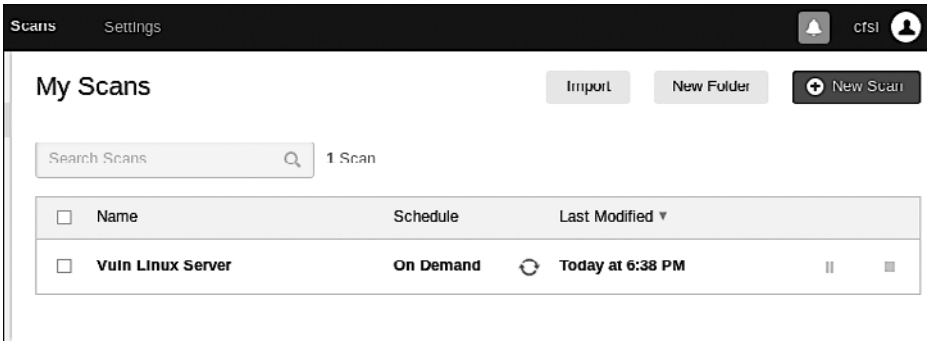


Рис. 6.10. Общая информация о сканировании

Если щелкнете кнопкой мыши на сканируемом целевом компьютере, то увидите более подробный список обнаруженных уязвимостей. Уязвимости имеют цветовую маркировку:

- красный — критический уровень;
- оранжевый — высокий;
- желтый — средний;
- зеленый — низкий;
- синий — содержит информацию.

Как видно на рис. 6.11, при сканировании в общей сложности было обнаружено 70 уязвимостей, из которых шесть являются критическими и 17 — высокого уровня. Это значит, что машина очень уязвима.

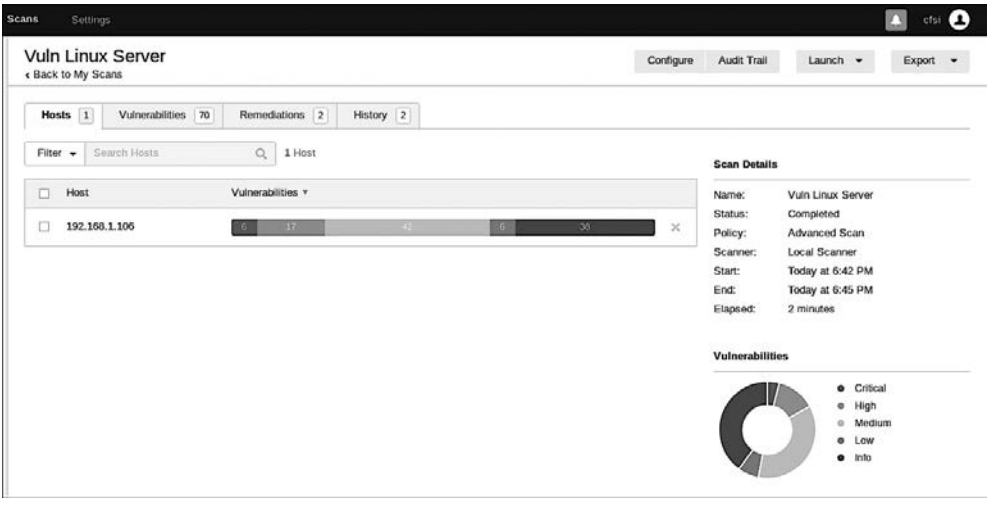


Рис. 6.11. Отчет о найденных уязвимостях

Если щелкнуть кнопкой мыши на цветных категориях уязвимостей, обнаруженные уязвимости отобразятся в порядке от наиболее уязвимых (то есть критических) до наименее уязвимых (информационных) (рис. 6.12).

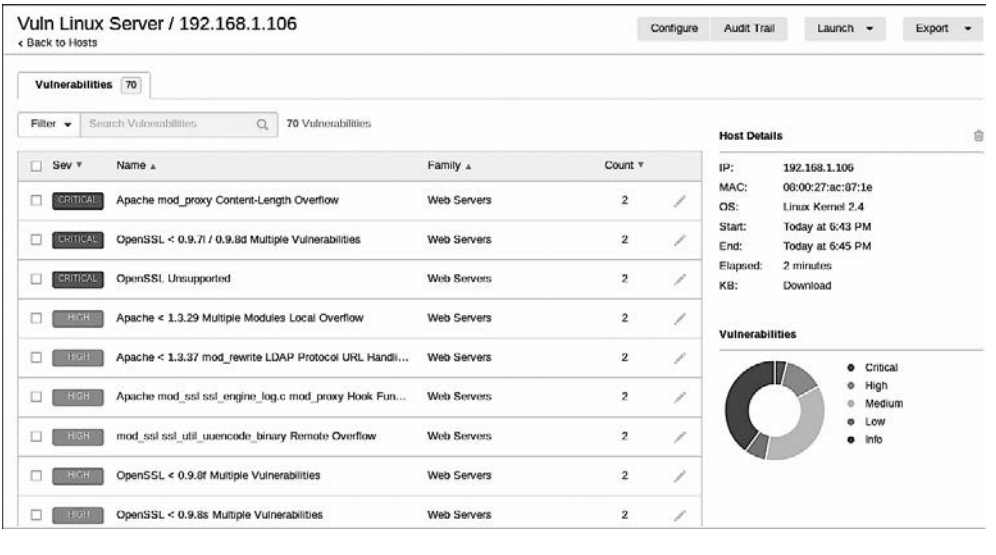


Рис. 6.12. Отображение найденных уязвимостей в порядке от критических до информационных

Полученная информация включает в себя сведения не только об уязвимости, но и об эксплойтах. Она позволяет испытателю запланировать и осуществить дополнительные атаки на эти уязвимости (рис. 6.13).

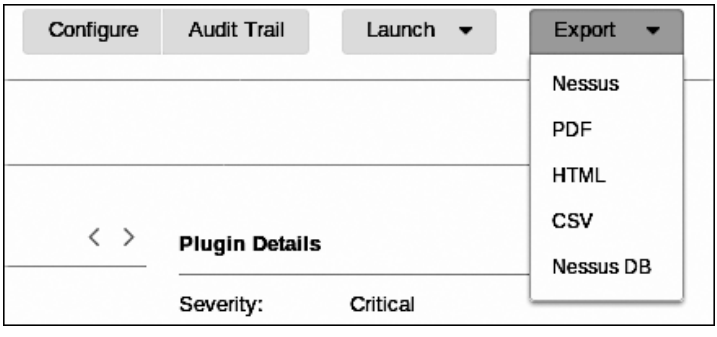


Рис. 6.13. Создание отчетности

Nessus — мощный инструмент с большим количеством функциональных возможностей, который можно использовать при тестировании на проникновение. Он предоставляет большой объем информации. К сожалению, в этом разделе мы не сможем рассмотреть весь функционал программы, но рекомендуем вам потратить некоторое время на самостоятельное изучение доступных функций. Обратите внимание, что Tenable предлагает бесплатно протестировать и домашнюю версию. Если же вы желаете протестировать внешние IP-адреса или применяете Nessus для клиента, вам придется воспользоваться платной версией.

OpenVAS

Open Vulnerability Assessment System (открытая система оценки уязвимостей, OpenVAS) — фреймворк, состоящий из нескольких сервисов и утилит. OpenVAS — это сканер с открытым исходным кодом. Он прост в установке и имеет удобный интерфейс, позволяющий выполнять активный мониторинг (с активными действиями в сети). Согласно сайту http://www.openvas.org/about.html при работе OpenVAS использует коллекцию уязвимостей, состоящую из 50 000 тестов (NVTs). OpenVAS является основой линейки профессиональных устройств Greenbone Secure Manager.

Чтобы установить OpenVAS, откройте терминал и введите команду `apt-get install openvas` (рис. 6.14).

Когда установка OpenVAS будет завершена, для запуска конфигурации введите в командную строку терминала команду `openvas-setup`. Процесс конфигурации займет некоторое время, в зависимости от загрузки сети и скорости подключения к Интернету (рис. 6.15).

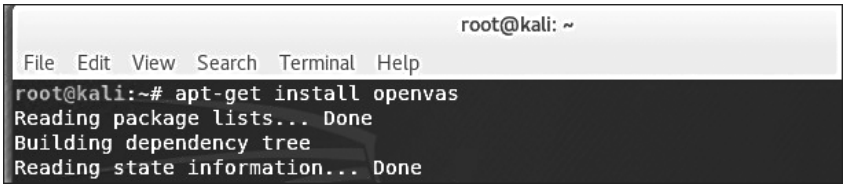


Рис. 6.14. Установка OpenVAS

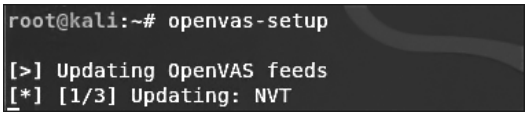


Рис. 6.15. Конфигурация приложения

В конце процесса установки и настройки OpenVAS сгенерирует пароль, который потребуется при запуске OpenVAS (рис. 6.16).

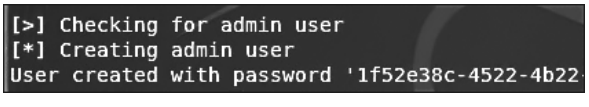


Рис. 6.16. Пароль сгенерирован

Для запуска сервиса OpenVAS введите команду `openvas-start`. Далее запустите браузер и введите в адресную строку https://127.0.0.1:9392 или https://localhost:9392.

При повторном использовании OpenVAS откройте терминал и введите команду openvas-start. Новую установку запускать не следует.

Вам после ввода предыдущего URL-адреса снова придется добавить исключение безопасности. Для этого нажмите кнопку Advanced (Дополнительно), далее — кнопку Add Exception (Добавить исключение), а затем кнопку Confirm Security Exception (Подтвердить исключение безопасности) (рис. 6.17).

При появлении запроса войдите в систему, введя имя пользователя admin и пароль, сгенерированный в процессе установки. Убедитесь, что логин и пароль надежно сохранены, так как вам при работе с OpenVAS неоднократно потребуется входить в систему (рис. 6.18).

Чтобы запустить сканирование, щелкните на ярлыке вкладки Scans (Сканирования), а затем на строке Tasks (Задачи). Откроется информационное окно, в котором нужно выбрать мастер задач. Он представлен в виде фиолетового значка, расположенного в левом верхнем углу экрана (рис. 6.19).

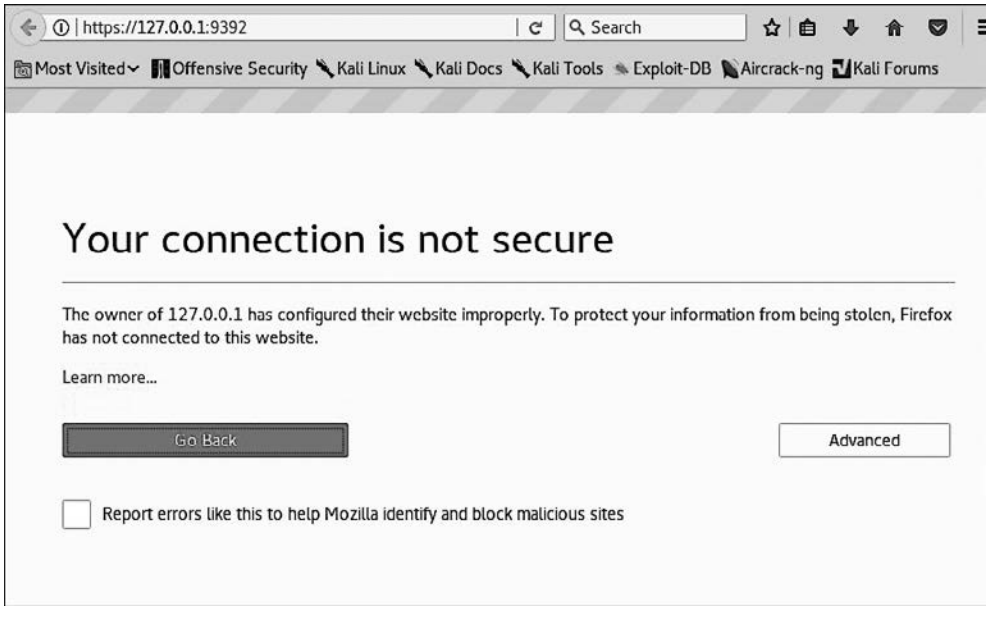


Рис. 6.17. Подтверждение добавления исключения безопасности

Рис. 6.18. Запуск OpenVAS

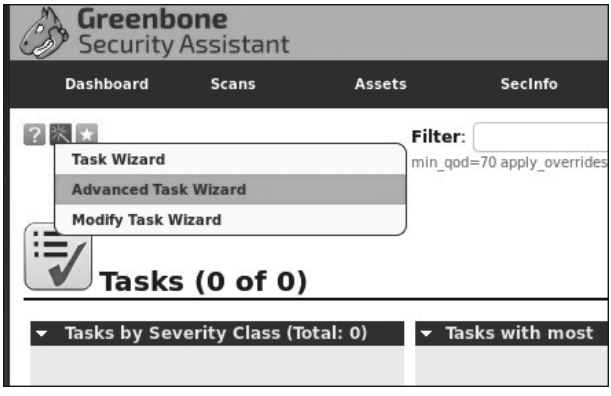


Рис. 6.19. Запуск сканирования

В открывшемся меню щелкните на строке Advanced Task Wizard (Мастер расширенных задач). В появившихся полях введите соответствующую информацию (рис. 6.20). Обратите внимание: поле Scan Config (Конфигурация сканирования) имеет на выбор несколько типов сканирования, включая Discovery (Обнаружение), Full and fast (Полное и быстрое), Full and fast ultimate (Полное и быстрое окончательное) и Full and very deep ultimate (Полное и очень глубокое окончательное) (наиболее трудоемкий и затратный по времени вариант).

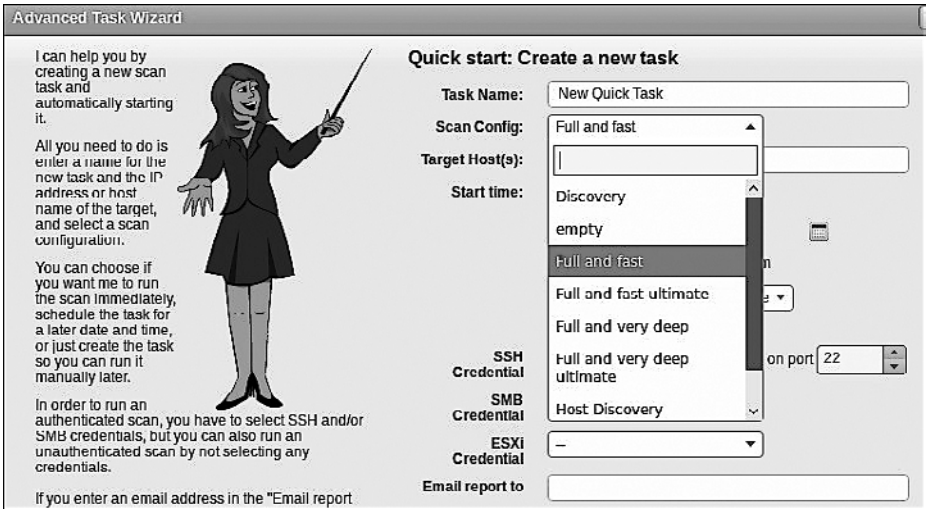


Рис. 6.20. Создаем новую задачу

Параметр Start time (Время начала) позволяет испытателю на проникновение запланировать сканирование. Это очень полезная функция. Сканирование может

нарушать работу сети, поэтому его лучше выполнять тогда, когда сеть не сильно загружена, то есть в нерабочее время или в выходные дни.

После того как все поля будут заполнены, прокрутите страницу вниз и нажмите кнопку Create (Создать). В результате запустится сканирование и на экране появится сводка сведений о сканировании и состоянии (рис. 6.21).

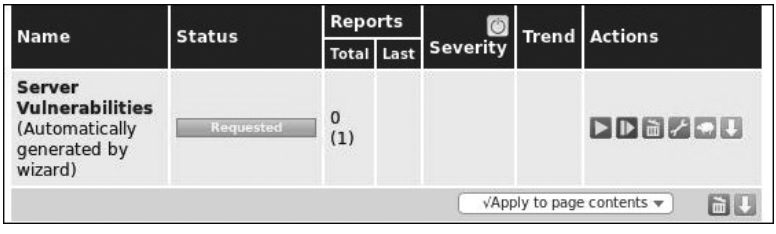


Рис. 6.21. Сканирование запущено

Чтобы просмотреть дополнительные сведения о задаче, в поле Name (Имя) щелкните на имени задачи (рис. 6.22).

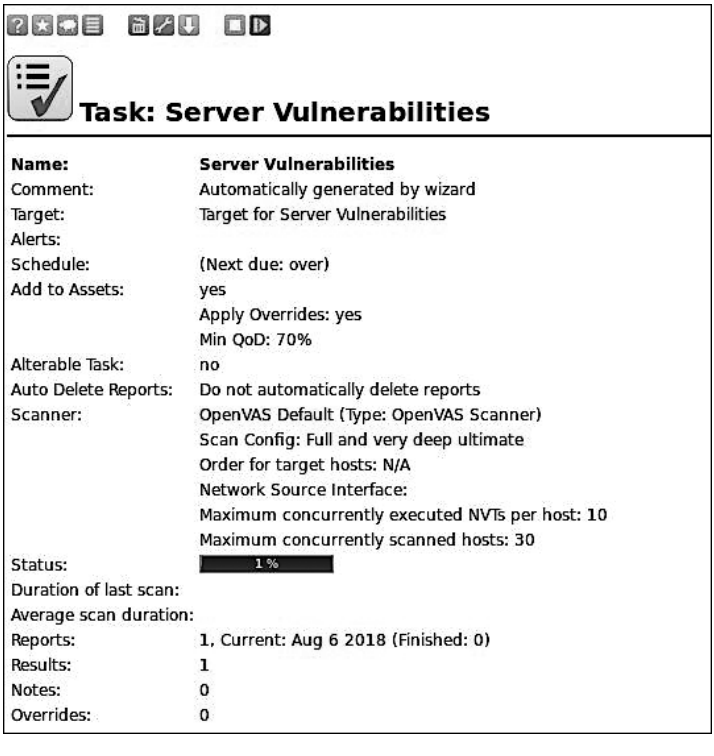


Рис. 6.22. Дополнительные сведения о задаче

По завершении сканирования нажмите кнопку Done (Готово). При этом будет создан отчет со списком обнаруженных уязвимостей и оценкой степени угрозы для каждой из них (рис. 6.23).

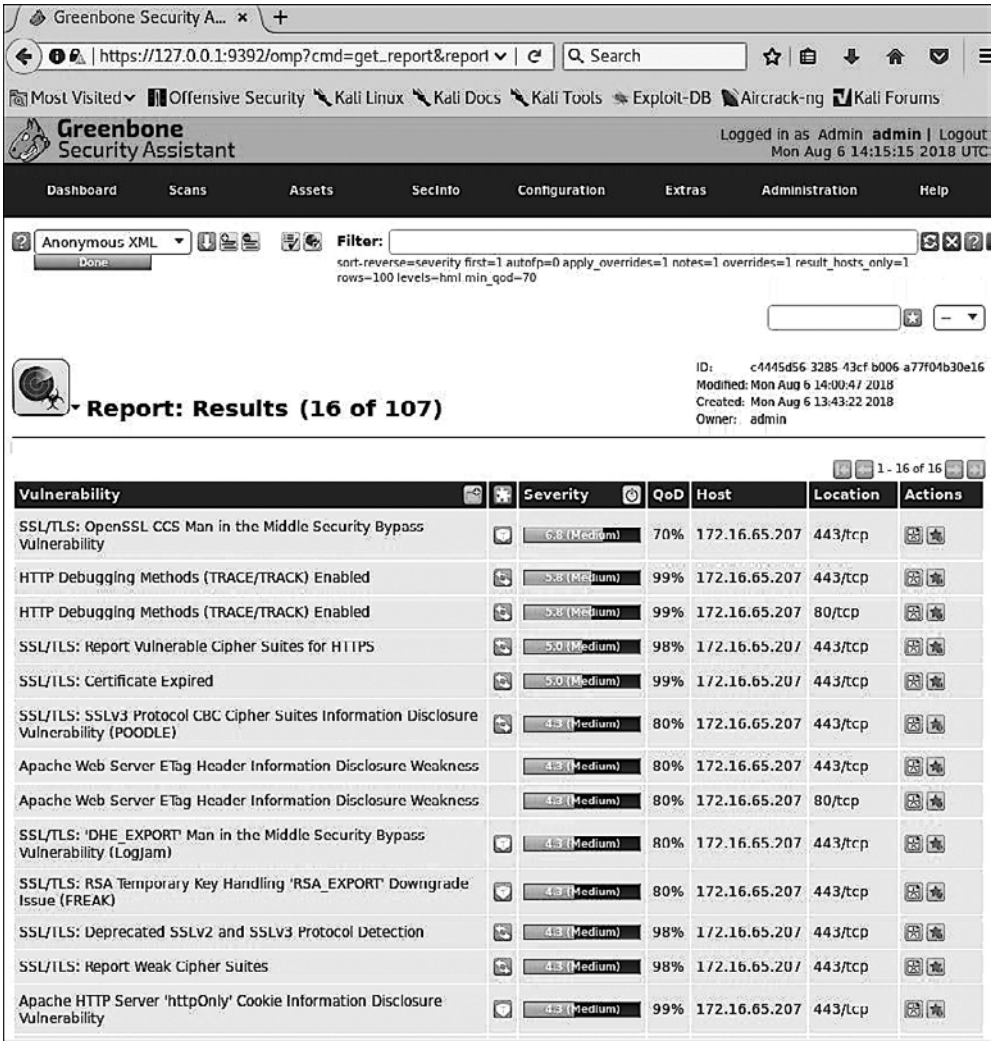
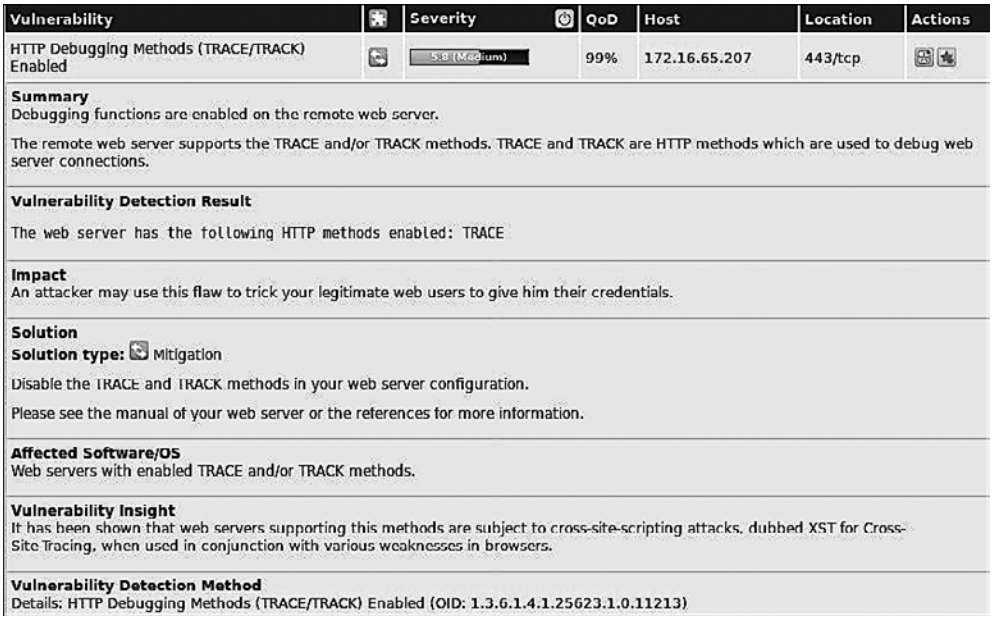


Рис. 6.23. Отчет о сканировании

Если щелкнуть кнопкой мыши на любой из перечисленных уязвимостей, отобразятся дополнительные сведения, такие как Summary (Сводка), Impact (Влияние), Solution (Решение), Affected Software/OS (Уязвимое программное обеспечение) и др. (рис. 6.24).



Vulnerability	Severity	QoD	Host	Location	Actions
HTTP Debugging Methods (TRACE/TRACK) Enabled	5.8 (Medium)	99%	172.16.65.207	443/tcp	

Summary
Debugging functions are enabled on the remote web server.

The remote web server supports the TRACE and/or TRACK methods. TRACE and TRACK are HTTP methods which are used to debug web server connections.

Vulnerability Detection Result

The web server has the following HTTP methods enabled: TRACE

Impact
An attacker may use this flaw to trick your legitimate web users to give him their credentials.

Solution
Solution type: Mitigation

Disable the TRACE and TRACK methods in your web server configuration.

Please see the manual of your web server or the references for more information.

Affected Software/OS
Web servers with enabled TRACE and/or TRACK methods.

Vulnerability Insight
It has been shown that web servers supporting this methods are subject to cross-site-scripting attacks, dubbed XST for Cross-Site Tracing, when used in conjunction with various weaknesses in browsers.

Vulnerability Detection Method
Details: HTTP Debugging Methods (TRACE/TRACK) Enabled (OID: 1.3.6.1.4.1.25623.1.0.11213)

Рис. 6.24. Дополнительные сведения

Сканирование уязвимостей Linux с помощью Lynis

Разработанное компанией Cisofy (www.cisofy.com) приложение *Lynis* — это инструмент проверки безопасности, который управляется из командной строки Kali Linux. Это бесплатная программа, но доступна и корпоративная версия. Lynis используется для автоматизированной оценки безопасности и сканирования уязвимостей в различных версиях операционных систем Linux, macOS X и Unix.

В отличие от других приложений такого типа Lynis специализируется на проверке безопасности законодательных актов о передаче и защите сведений учреждений здравоохранения (HIPAA), защите стандарта безопасности платежных карт PCI DSS, систем внутреннего контроля SOX и GLBA. Это приложение позволит предприятиям, использующим различные стандарты, обеспечить безопасность своих систем.

Lynis можно загрузить и установить самостоятельно. Установка приложения в целевой системе сэкономит трафик, по сравнению с установкой на удаленном компьютере.

Lynis входит в пакет Kali Linux, но также может быть клонирован с GitHub (https://github.com/CISOfy/lynis) или скачан с официального сайта (https://cisofy.com/documentation/lynis/get-started/#installation).

Для запуска Lynis в Kali выберите команду меню Applications ▸ Vulnerability Analysis ▸ Lynis (Приложения ▸ Анализ уязвимостей ▸ Lynis). Для запуска приложения из командной строки введите в терминале команду `lynis`. Она отобразит установленную версию Lynis (в данном случае 2.6.2) и инициализирует программу. Вы также увидите список всех параметров команды (рис. 6.25).

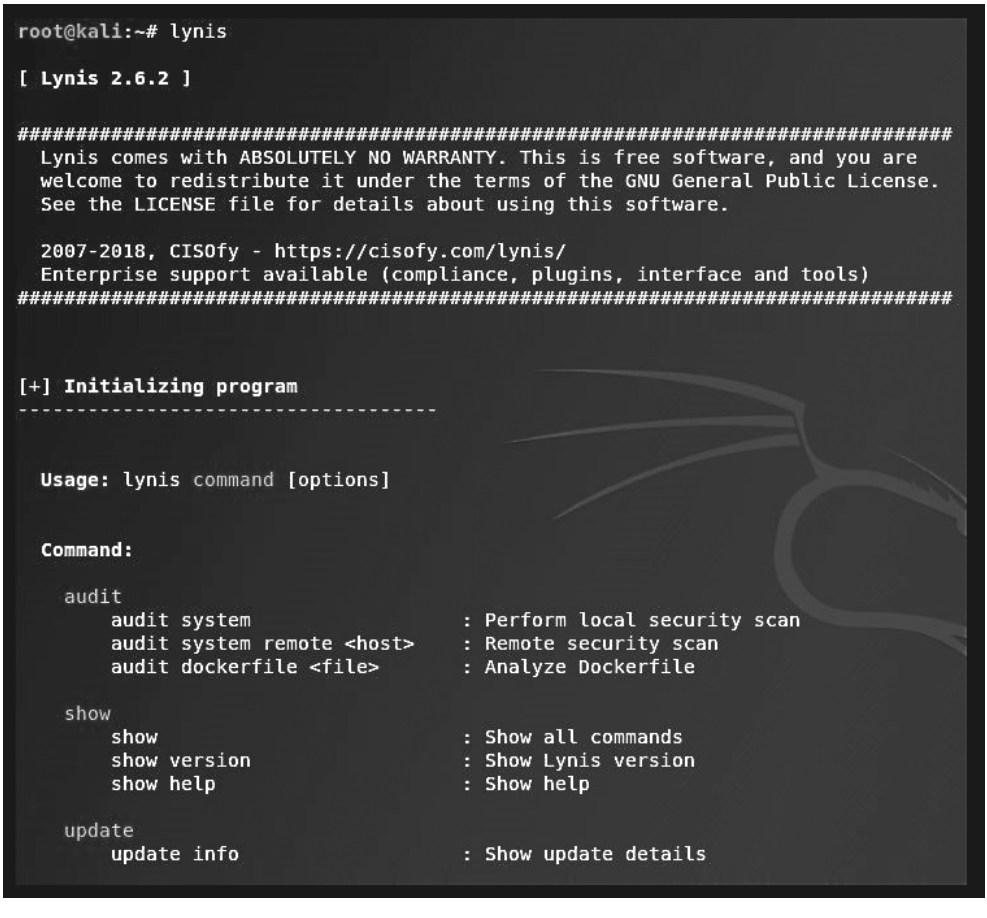


Рис. 6.25. Список параметров команды Lynix

Если вы подзабыли нужную команду, введите `lynis show commands` (рис. 6.26).

Lynis — полностью механизированный инструмент проверки безопасности, у которого есть минимальный набор команд. Чтобы проверить вашу машину Kali Linux, просто введите `lynis audit system`. Время, которое займет проверка, зависит от характеристик проверяемой машины Kali Linux. Обычно проверка длится от 15 до 30 минут. Результат проверки показан на рис. 6.27.

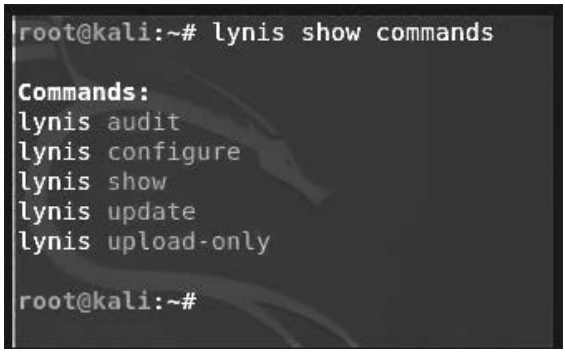


Рис. 6.26. Отображение команд Lynix

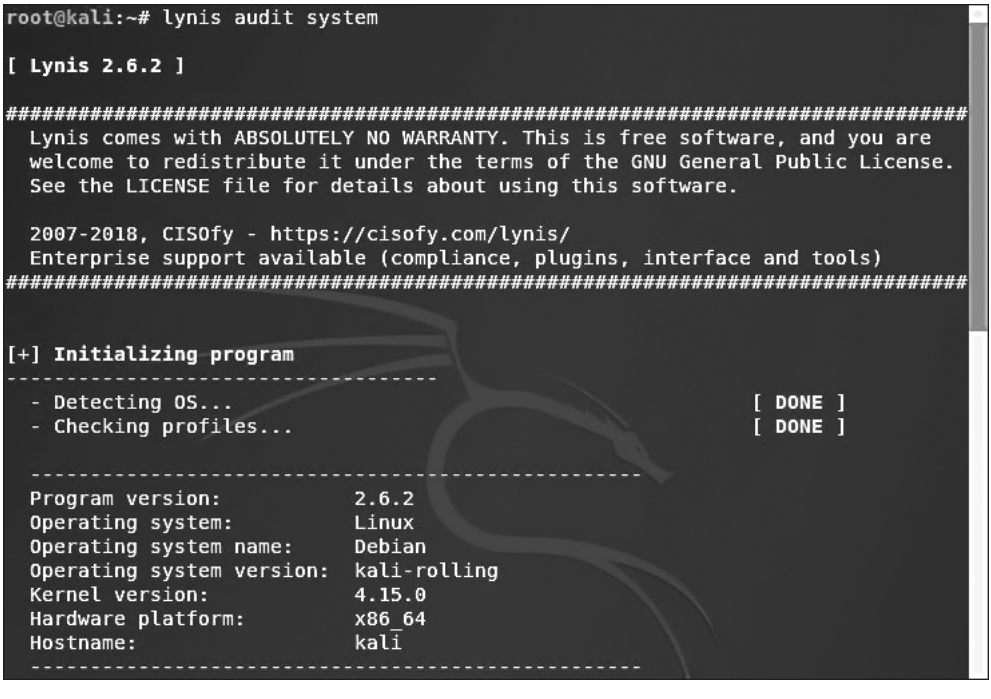


Рис. 6.27. Результат проверки машины Kali Linux

Результаты проверки включают в себя сведения:

- о версии Debian;
- загрузке и службах;
- ядре;
- памяти и процессоре;

- пользователях, группах и проверке подлинности;
- оболочке;
- файловой системе;
- USB-устройствах;
- сети и брандмауэрах;
- портах и принтерах;
- надежности ядра.

```
[+] Networking
------------------------------------
  - Checking IPv6 configuration                          [ ENABLED ]
      Configuration method                               [ AUTO ]
      IPv6 only                                          [ NO ]
  - Checking configured nameservers
    - Testing nameservers
        Nameserver: 10.2.0.24                            [ OK ]
    - Minimal of 2 responsive nameservers                [ WARNING ]
  - Checking default gateway                             [ DONE ]
  - Getting listening ports (TCP/UDP)                    [ DONE ]
      * Found 1 ports
  - Checking promiscuous interfaces                      [ OK ]
  - Checking waiting connections                         [ OK ]
  - Checking status DHCP client                          [ RUNNING ]
  - Checking for ARP monitoring software                 [ NOT FOUND ]

[+] Printers and Spools
------------------------------------
  - Checking cups daemon                                 [ NOT FOUND ]
  - Checking lp daemon                                   [ NOT RUNNING ]

[+] Software: e-mail and messaging
------------------------------------

[+] Software: firewalls
------------------------------------
  - Checking iptables kernel module                      [ FOUND ]
    - Checking iptables policies of chains               [ FOUND ]
    - Checking for empty ruleset                         [ WARNING ]
    - Checking for unused rules                          [ OK ]
  - Checking host based firewall                         [ ACTIVE ]

[+] Software: webserver
------------------------------------
  - Checking Apache (binary /usr/sbin/apache2)           [ FOUND ]
      Info: Configuration file found (/etc/apache2/apache2.conf)
      Info: No virtual hosts found
    * Loadable modules                                   [ FOUND (116) ]
        - Found 116 loadable modules
          mod_evasive: anti-DoS/brute force              [ NOT FOUND ]
          mod_reqtimeout/mod_qos                         [ FOUND ]
```

Рис. 6.28. Сведения, полученные с помощью Lynix

На рис. 6.29 показан фрагмент результатов проверки Lynis с четырьмя предупреждениями и 40 предложениями.

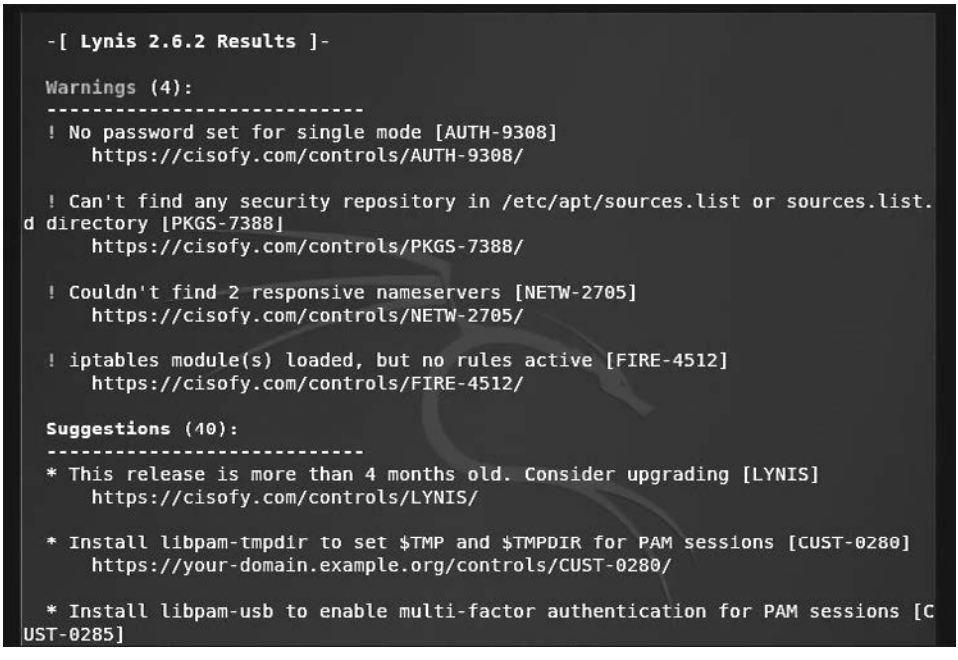


Рис. 6.29. Фрагмент результатов проверки

Прокрутив результаты теста до конца, мы увидим общую сводку по проверке (рис. 6.30).

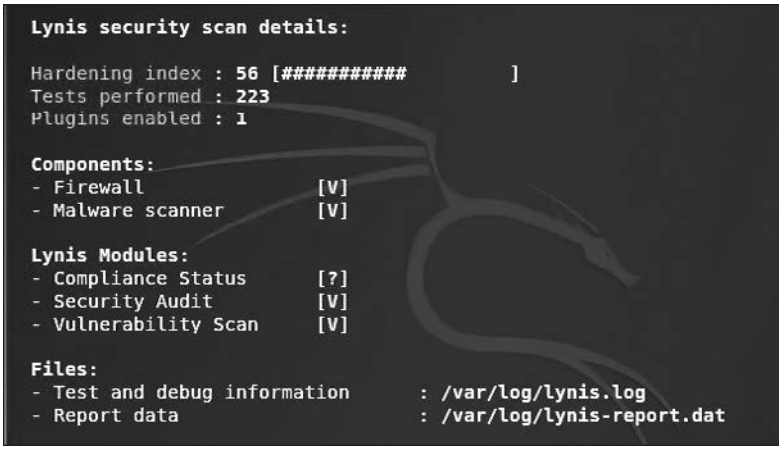


Рис. 6.30. Общая сводка по проверке компьютера

Сканирование и перечисление уязвимостей с помощью SPARTA

SPARTA — это инструмент с пользовательским интерфейсом для тестирования на проникновение сети. Авторы приложения — Антонио Кин (Antonio Quina) и Леонидас Ставлиотис (Leonidas Stavliotis) из компании SECFORCE. SPARTA — стандартное приложение Kali Linux. В рамках одного инструмента оно автоматизирует процессы сканирования, перечисления и оценки уязвимостей. Кроме функций сканирования и перечисления, в SPARTA также встроено средство для взлома паролей с помощью грубой силы.

Последние версии SPARTA можно загрузить из GitHub и клонировать на локальную машину. Для этого достаточно ввести команду git clone https://github.com/secforce/sparta.git.

Для запуска SPARTA в Kali Linux 2018 выберите команду меню Applications ▸ Vulnerability Analysis ▸ SPARTA (Приложения ▸ Анализ уязвимостей ▸ SPARTA). Чтобы добавить в область проверки целевую машину или группу целевых машин, в графическом интерфейсе SPARTA 1.0.3 щелкните кнопкой мыши на левой панели. Далее добавьте в поле ввода IP Range (IP-диапазон) диапазон проверяемых IP-адресов (рис. 6.31).

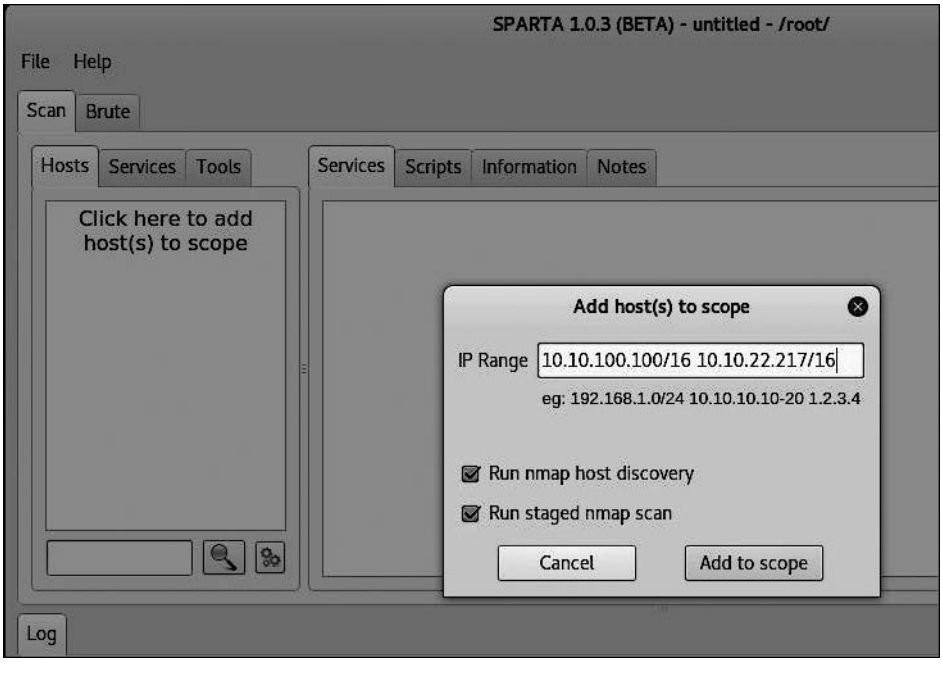


Рис. 6.31. Диапазон IP-адресов для проверки введен

После того как IP адреса узлов сети будут добавлены, нажмите кнопку Add to scope (Добавить в область). Начнется поэтапное сканирование целевых объектов (рис. 6.32).

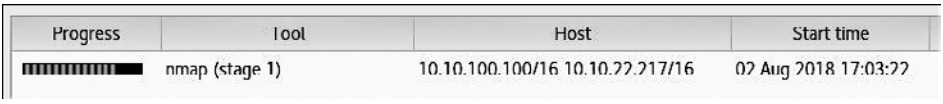


Рис. 6.32. Процесс сканирования узла сети

После завершения сканирования карты сети в основном окне SPARTA появятся следующие вкладки: Services (Службы), Scripts (Сценарии), Information (Информация), Notes (Заметки), Nikto и Screenshot (Скриншоты), на которых вы найдете очень полезную информацию.

По умолчанию сначала откроется вкладка Services (Службы) со списком открытых портов и служб (рис. 6.33).

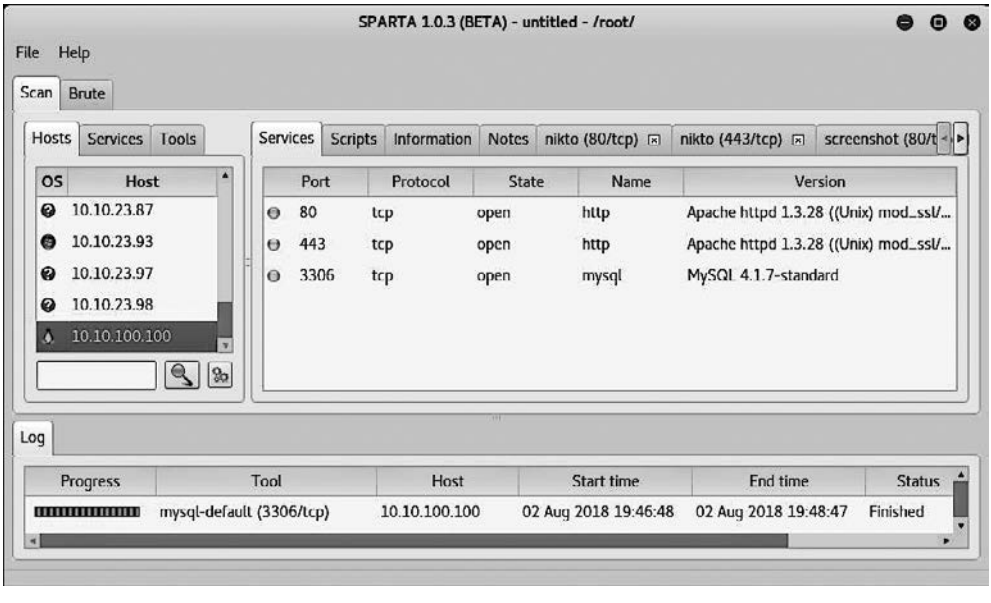


Рис. 6.33. Вкладка Services (Службы) со списком открытых портов и служб

Если откроете вкладку Information (Информация), то увидите собранную информацию о целевой машине. Там будет указан IP, количество открытых, закрытых и отфильтрованных портов (если таковые имеются), а также операционная система вместе с ее версией и значение точности определения информации (рис. 6.34).

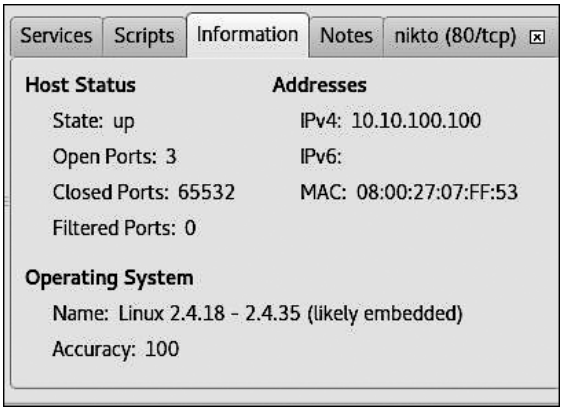


Рис. 6.34. Вкладка Information (Информация)

Поскольку в нашем случае целевой машиной был выбран Linux-сервер, применен инструмент сканирования Nikto. Чтобы просмотреть список найденных уязвимостей, откройте вкладку nikto (80/tcp) (рис. 6.35).

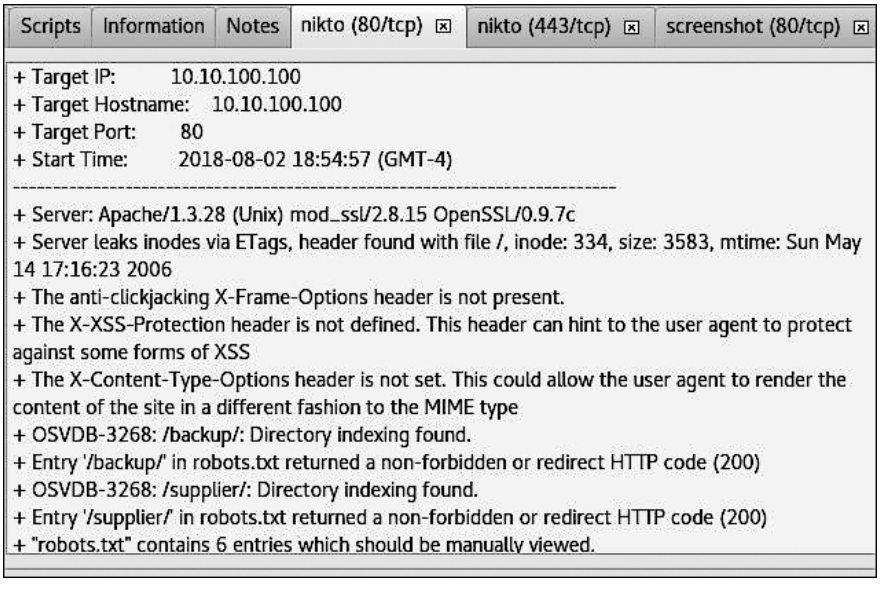


Рис. 6.35. Вкладка nikto (80/tcp)

Многие из обнаруженных уязвимостей имеют префикс OSVBD, который указывает на то, что информацию о них можно искать в базах данных сайтов *Common Vulnerabilites and Exposures (CVE)* и *Open Source Vulnerabilities Database (OSVDB)*.

Для получения подробной информации испытатель на проникновение может воспользоваться простым поиском Google, введя в поисковую строку название уязвимости, например OSVDB-3268, которая была выявлена при предыдущем сканировании. Впоследствии выявленные уязвимости могут быть использованы, например, инструментом Metasploit. Но об этом поговорим в следующих главах.

Теперь посмотрим на другую сканируемую машину под управлением операционной системы Windows. В области Hosts (Хосты) слева найдите IP-адрес 10.10.22.217 и щелкните на нем кнопкой мыши. Далее откройте вкладку Services (Службы) (рис. 6.36). Здесь вы увидите список открытых портов.

Services | Scripts | Information | Notes | smbenum (445/tcp) | nikto (5357/tcp) | screenshot (5357/tcp)

Port	Protocol	State	Name	Version
135	tcp	open	msrpc	Microsoft Windows RPC
137	udp	open	netbios-ns	Microsoft Windows netbios-ns (workgroup: WORKGROUP)
139	tcp	open	netbios-ssn	Microsoft Windows netbios-ssn
445	tcp	open	microsoft-ds	Microsoft Windows 7 - 10 microsoft-ds (workgroup: WORKGROUP)
5357	tcp	open	http	Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
49152	tcp	open	msrpc	Microsoft Windows RPC
49153	tcp	open	msrpc	Microsoft Windows RPC
49154	tcp	open	msrpc	Microsoft Windows RPC
49155	tcp	open	msrpc	Microsoft Windows RPC
49156	tcp	open	msrpc	Microsoft Windows RPC
49157	tcp	open	msrpc	Microsoft Windows RPC

Рис. 6.36. Вкладка Services (Службы) с информацией о портах машины 10.10.22.217

Из этой информации видно, что исследуемая нами машина работает под управлением Windows. Для обследования портов данной машины в SPARTA был запущен инструмент smbenum. С его помощью мы проверили наличие нулевых сессий и просмотрели этот компьютер. В ходе этих операций мы, в частности, искали сведения о пользователях и общих ресурсах (рис. 6.37).

SPARTA осуществляет сканирование, перечисление и оценку уязвимости, позволяя испытателю выполнять различные функции тестирования на проникновение в сеть. Для этого на вкладке Services (Сервисы) щелкните правой кнопкой мыши на любом из открытых портов и выберите в появившемся меню нужную команду.

На рис. 6.38 вы видите контекстное меню, появившееся после того, как мы щелкнули правой кнопкой мыши на строке open port 3306. Используя команды этого контекстного меню, вы можете попробовать открыть порт с помощью Telnet, Netcat, или клиента MySQL. Для клиента MySQL вам потребуются права root. Вы также можете попытаться взломать пароли с применением грубой силы.

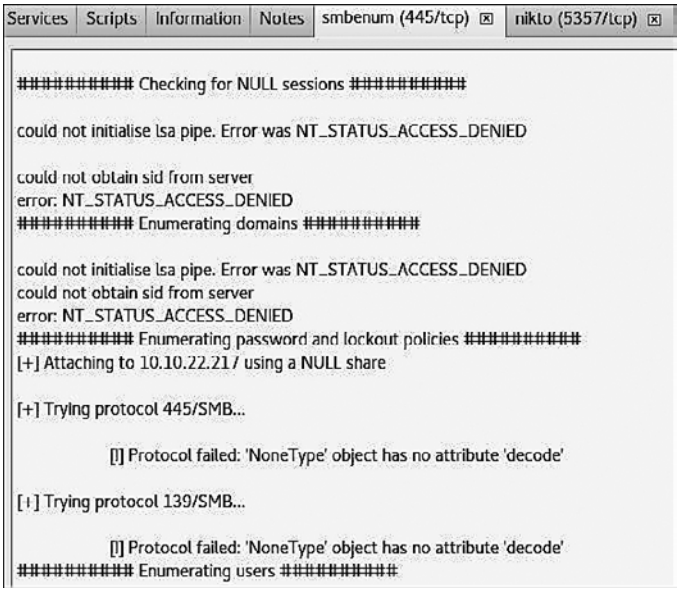


Рис. 6.37. Отчет об исследовании с помощью инструмента smbenum (445/tcp)

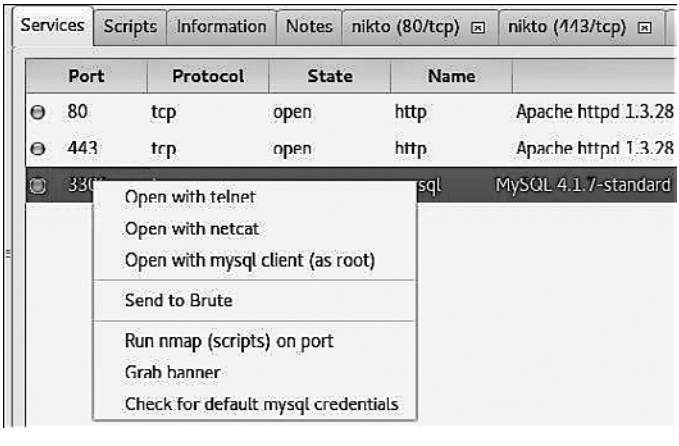


Рис. 6.38. Контекстное меню для порта 3306

Если в контекстном меню выбрать команду Send to Brute (Отправить в Brute), то через выбранный порт будет предпринята попытка атаки с помощью инструмента взлома пароля THC Hydra. Наряду с другими вариантами для получения нужных данных можно применять списки с именами пользователей и паролей. Указав необходимые параметры, нажмите кнопку Run (Выполнить), чтобы предпринять попытку атаки.

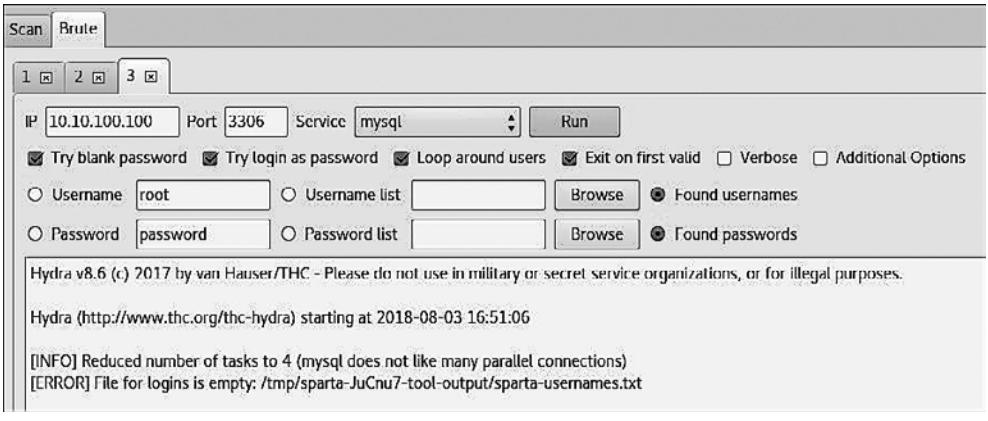


Рис. 6.39. Вкладка инструмента Brute

Это далеко не единственные инструменты, доступные в SPARTA. Если, допустим, щелкнуть правой кнопкой мыши на open port 445 компьютера под управлением операционной системы Windows, вы увидите большой список инструментов (рис. 6.40).

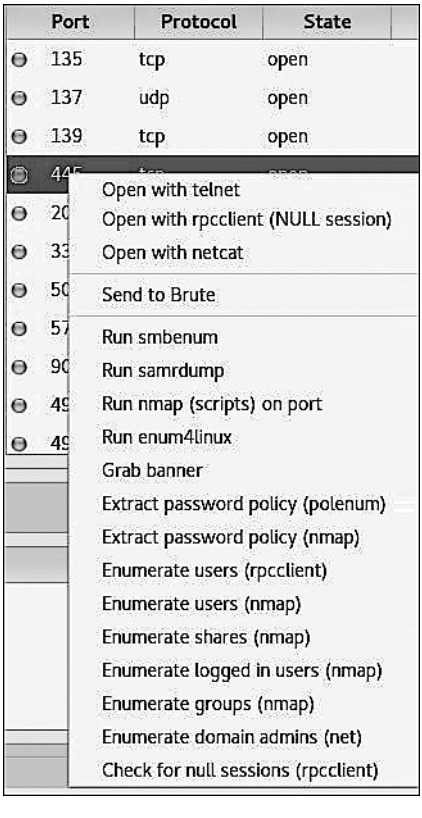


Рис. 6.40. Список команд контекстного меню для компьютера под управлением Windows