Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Комсомольский-на-Амуре государственный университет»

УТВЕРЖДАЮ
Декан
факультета компьютерных технологий
(наименование факультета)
Я.Ю. Григорьев
(подпись, ФИО)
«04» 06 20 21 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
Форензика

Направление подготовки	*10.05.03 "Информационная безопасность автоматизированных систем"*
Направленность (профиль) образовательной программы	*Анализ безопасности информационных систем*
Квалификация выпускника	*специалист по защите информации*
Год начала подготовки *(по учебному плану)*	*2021*
Форма обучения	*очная*
Технология обучения	*традиционная*

Курс	Семестр	Трудоемкость, з.е.
5	*А*	*3*

Вид промежуточной аттестации	Обеспечивающее подразделение
Зачет с оценкой	*Кафедра ИБАС - Информационная безопасность автоматизированных систем*

Комсомольск-на-Амуре 2021

Разработчик рабочей программы:

Доцент ИБАС к.э.н		Обласов А.А.
(должность, степень, ученое звание)	(подпись)	(ФИО)

СОГЛАСОВАНО:

Заведующий кафедрой ИБАС		Лошманов А.Ю.
(наименование кафедры)	(подпись)	(ФИО)

1 Общие положения

Рабочая программа дисциплины «Форензика» составлена в соответствии с требованиями федерального государственного образовательного стандарта, утвержденного приказом Министерства образования и науки Российской Федерации № 1509 от 01.12.2016, и основной профессиональной образовательной программы подготовки «Обеспечение информационной безопасности распределенных информационных систем» по направлению 10.05.03 "Информационная безопасность автоматизированных систем".

Практическая подготовка реализуется на основе:

Профессиональный стандарт утвержденного приказом Министерства труда и социальной защиты от 15 сентября 2016 года N 522н №843 **"Специалист по защите информации в автоматизированных системах"** зарегистрированного в Министерстве юстиции Российской Федерации 28 сентября 2016 года, регистрационный N 43857. Обобщенная трудовая функция: B/03.6 Управление защитой информации в автоматизированных системах.

Задачи дисциплины	Приобретение обучаемыми необходимого объёма знаний и практических навыков в области обеспечения информационной безопасности конфиденциальной информации
Основные разделы / темы дисциплины	1. Организация и проведение работ по обработке и защите конфиденциальной информации 2. Обращение со служебной информацией ограниченного доступа на предприятиях

2 Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами образовательной программы

Процесс изучения дисциплины «Форензика» направлен на формирование следующих компетенций в соответствии с ФГОС ВО и основной образовательной программой (таблица 1):

Таблица 1 – Компетенции и планируемые результаты обучения по дисциплине

Код и наименование компетенции	Планируемые результаты обучения по дисциплине		
	Перечень знаний	Перечень умений	Перечень навыков
Профессиональные			
Способность проводить анализ рисков информационной безопасности автоматизированной системы (ПК-5)	З1 (ПК-5-2) знать основные методы анализа сетевого трафика;	У1 (ПК-5-2) уметь анализировать сетевую инормацию;	Н1 (ПК-5-2) владеть навыком работы с Wireshark по анализу различных протоколов в сети;
Способность обеспечивать эффективное применение средств защиты информационно-технологических ресурсов автоматизированной системы и восстановление их работоспособности при возникновении нештатных ситуаций (ПК-25)	З2 (ПК-25-4) знать об основных методах исследования компьютерной информации;	У2 (ПК-25-4) уметь анализировать лог файлы операционных систем;	Н2 (ПК-25-4) владеть навыком работы с программой volatility;

Таблица 2 – Компетенции и индикаторы их достижения

Код и наименование компетенции	Индикаторы достижения	Планируемые результаты обучения по дисциплине

Общепрофессиональные		
ОПК-7.3. Способен проводить анализ защищенности и верификацию программного обеспечения информационных систем	**ОПК-7.3..1** Знает виды и порядок проведения анализа защищенности автоматизированных систем	Знает виды и порядок проведения анализа защищенности автоматизированных систем
	ОПК-7.3..2 Умеет выбирать программное и аппаратное обеспечение для проведения анализа защищённости автоматизированных систем	Умеет выбирать программное и аппаратное обеспечение для проведения анализа защищённости автоматизированных систем
	ОПК-7.3..3 Владеет навыками проведения анализа защищенности и верификации программного обеспечения автоматизированных систем	Владеет навыками проведения анализа защищенности и верификации программного обеспечения автоматизированных систем

3 Место дисциплины (модуля) в структуре образовательной программы

Дисциплина «Форензика» изучается на 5 курсе(ах) в 10 семестре(ах).

Дисциплина входит в состав блока 1 «Дисциплины (модули)» и относится к базовой части.

Для освоения дисциплины необходимы знания, умения, навыки и (или) опыт практической деятельности, сформированные в процессе изучения дисциплин / практик: Анализ защищенности распределенных информационных систем; Низкоуровневый анализ машинного кода

Знания, умения и навыки, сформированные при изучении дисциплины «Форензика», будут востребованы при изучении последующих дисциплин: Подготовка к процедуре защиты и защита выпускной квалификационной работы

Дисциплина «Форензика» частично реализуется в форме практической подготовки. Практическая подготовка организуется путем выполнения лабораторных работ.

Дисциплина «Форензика» в рамках воспитательной работы направлена на формирование у обучающихся умения аргументировать, самостоятельно мыслить, развивает профессиональные умения, ответственности за выполнение учебно-производственных заданий

4 Объем дисциплины (модуля) в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся

Общая трудоемкость (объем) дисциплины составляет 3 з.е., 108 акад. час.

Распределение объема дисциплины (модуля) по видам учебных занятий представлено в таблице 2.

Таблица 2 – Объем дисциплины (модуля) по видам учебных занятий

Объем дисциплины	**Всего академических часов**
Общая трудоемкость дисциплины	108
Контактная аудиторная работа обучающихся с преподавателем (по видам учебных занятий), всего	64
В том числе:	
занятия лекционного типа (лекции и иные учебные занятия, предусматривающие преимущественную передачу учебной информации педагогическими работниками)	32
занятия семинарского типа (семинары, практические занятия, практикумы, лабораторные работы, коллоквиумы и иные аналогичные занятия)	32
Самостоятельная работа обучающихся и контактная работа, включающая групповые консультации, индивидуальную работу обучающихся с преподавателями (в том числе индивидуальные консультации); взаимодействие в электронной информационно-образовательной среде вуза	44
Промежуточная аттестация обучающихся – Зачет с оценкой	

5 Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебной работы

Таблица 3 – Структура и содержание дисциплины (модуля)

Наименование разделов, тем и содержание материала	Виды учебной работы, включая самостоятельную работу обучающихся и трудоемкость (в часах)			
	Контактная работа преподавателя с обучающимися			СРС
	Лекции	Семинарские (практические занятия)	Лабораторные занятия	
Раздел 1 Исследование сетевого трафика. Теоретические основы исследования сетевого трафика, Исследование мультимедийного трафика Исследование данных передаваемых через протокол TCP	16		16	22

Наименование разделов, тем и содержание материала	Виды учебной работы, включая самостоятельную работу обучающихся и трудоемкость (в часах)			
	Контактная работа преподавателя с обучающимися			СРС
	Лекции	Семинарские (практические занятия)	Лабораторные занятия	
Раздел 2 Исследование компьютерной информации Теоретические основы исследования компьютерной информации Исследование файловых систем операционных систем Исследование ОЗУ	16		16	22
ИТОГО по дисциплине	**32**		**32**	**44**

6 Внеаудиторная самостоятельная работа обучающихся по дисциплине (модулю)

При планировании самостоятельной работы студенту рекомендуется руководствоваться следующим распределением часов на самостоятельную работу (таблица 4):

Таблица 4 – Рекомендуемое распределение часов на самостоятельную работу

Компоненты самостоятельной работы	Количество часов
Изучение теоретических разделов дисциплины	6
Подготовка к занятиям семинарского типа	16
Подготовка и оформление РГР	22
	44

7 Оценочные средства для проведения текущего контроля и промежуточной аттестации обучающихся по дисциплине (модулю)

Фонд оценочных средств для проведения текущего контроля успеваемости и промежуточной аттестации представлен в Приложении 1.

Полный комплект контрольных заданий или иных материалов, необходимых для оценивания результатов обучения по дисциплине (модулю), практике хранится на кафедре-разработчике в бумажном и электронном виде.

8 Учебно-методическое и информационное обеспечение дисциплины (модуля)

8.1 Основная литература

1. Багмет А.М. Извлечение данных из электронных устройств как самостоятельное следственное действие / А.М. Багмет, С.Ю. Скобелин // Право и кибербезопасность. 2013. N 2. С. 22 - 27.

2. Скобелин С.Ю. Использование цифровых технологий при доказывании преступной деятельности / С.Ю. Скобелин // Российский следователь. 2019. N 3. С. 26 - 28.

3. Шеметов А.К. О понятии виртуальных следов в криминалистике / А.К. Шеметов // Российский следователь. 2014. N 20. С. 52 - 54.

4. Шаньгин, В. Ф. Информационная безопасность компьютерных систем и сетей [Электронный ресурс] : учебное пособие / В. Ф. Шаньгин. – М. : ФОРУМ : ИНФРА-М, 2014. – 416 с. // ZNANIUM.COM : электронно-библиотечная система. – Режим доступа: **http://www.znanium.com/catalog.php**, ограниченный. – Загл. с экрана.

5. Шаньгин, В. Ф. Комплексная защита информации в корпоративных системах [Электронный ресурс] : учебное пособие / В. Ф. Шаньгин. – М. : ФОРУМ : ИНФРА-М, 2013. – 592 с. // ZNANIUM.COM : электронно-библиотечная система. – Режим доступа: **http://www.znanium.com/catalog.php**, ограниченный. – Загл. с экрана.

8.2 Дополнительная литература

1. Колисниченко, Д. Серверное применение Linux / Д. Колисниченко, М. Матвеев, Р. Прокди – Санкт-Петербург : БХВ-Петербург, 2016. – 510 с.

2. Уильям, Р. Windows 7 для продвинутых. Настройка, работа и администрирование/ Р. Уильям – Санкт-Петербург : Питер, 2015. – 576 с.

3. Трент, Р. Unix и Linux. Руководство системного администртора/ Р. Трент – Москва : Горячая Линия - Телеком, 2014. – 352 с.

4. Хилл, Б. Полный справочник по Cisco./ Б. Хилл – Санкт-Петербург : Питер, 2012. – 780 с.

8.3 Методические указания для студентов по освоению дисциплины

Обучение дисциплине «Форензика» предполагает изучение курса на аудиторных занятиях и в ходе самостоятельной работы. Аудиторные занятия проводятся в форме лекций и практических занятий.

Таблица 7 Методические указания к отдельным видам деятельности

Вид учебного занятия	Организация деятельности студента
Лекция	Написание конспекта лекций: кратко, схематично, последовательно фиксировать основные положения. Выделять ключевые слова, формулы, отмечать на полях уточняющие вопросы по теме занятия
Лабораторные занятие	Работа с автоматизированными рабочими местами.
Самостоятельная работа	Для более глубокого изучения разделов дисциплины предусмотрены отдельные виды самостоятельной работы: подготовка к практическим занятиям, изучение теоретических разделов дисциплины, подготовка РГР.

Самостоятельная работа является наиболее продуктивной формой образовательной и познавательной деятельности студента в период обучения. СРС направлена на углубление и закрепление знаний студента, развитие практических умений. СРС по дисциплине «Организация и технология защиты конфиденциальной информации в инфор-

мационных системах» включает следующие виды работ:

– работу с лекционным материалом, поиск и обзор литературы и электронных источников информации по индивидуальному заданию;

– опережающую самостоятельную работу;

– изучение тем, вынесенных на самостоятельную проработку;

– подготовку к практическим занятиям;

– выполнение и оформление РГР.

Контроль самостоятельной работы студентов и качество освоения дисциплины осуществляется посредством:

– представления в указанные контрольные сроки результатов выполнения заданий для текущего контроля;

– выполнения и защиты РГР;

8.4 Современные профессиональные базы данных и информационные справочные системы, используемые при осуществлении образовательного процесса по дисциплине

1. Электронно-библиотечная система ZNANIUM.COM – http://www. znanium.com.
2. Консультант+
3. Научная электронная библиотека Elibrary http://elibrary.ru.

8.5 Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины (модуля)

1. Wireshark (ссылка для свободного скачивания «https://www.wireshark.org/#download»)
2. Volatility (ссылка для свободного скачивания «https://github.com/volatilityfoundation/volatility»)
3. Сайт университета www.knastu.ru[Электронный ресурс]:. Раздел сотрудникам, документы СМК, режим доступа – свободный. Загл. с экрана
4. Научная электронная библиотека Elibrary http://elibrary.ru.

С целью повышения качества ведения образовательной деятельности в университете создана электронная информационно-образовательная среда. Она подразумевает организацию взаимодействия между обучающимися и преподавателями через систему личных кабинетов студентов, расположенных на официальном сайте университета в информационно-телекоммуникационной сети «Интернет» по адресу https://student.knastu.ru. Созданная информационно-образовательная среда позволяет осуществлять взаимодействие между участниками образовательного процесса посредством организации дистанционного консультирования по вопросам выполнения практических заданий. Материалы данного курса (10 семестр) выложены на портал ДО КнАГУ и организация взаимодействия в рамках данной дисциплины проводится с привлечением дистанционных технологий.

8.6 Лицензионное программное обеспечение, используемое при осуществлении образовательного процесса по дисциплине

Таблица 5 – Перечень используемого программного обеспечения

Наименование ПО	Реквизиты
Microsoft® Windows	Лицензионный сертификат № 46243844 от 09.12.2009

Professional 7 Russian	
Open Office или аналог	Свободно-распространяемое
Операционная система Kali Linux или аналог	Свободно-распространяемое
Операционная система Ubuntu или аналог	Свободно-распространяемое
Обозреватель Google Chrome или аналог	Свободно-распространяемое

9 Организационно-педагогические условия

Организация образовательного процесса регламентируется учебным планом и расписанием учебных занятий. Язык обучения (преподавания) — русский. Для всех видов аудиторных занятий академический час устанавливается продолжительностью 45 минут.

При формировании своей индивидуальной образовательной траектории обучающийся имеет право на перезачет соответствующих дисциплин и профессиональных модулей, освоенных в процессе предшествующего обучения, который освобождает обучающегося от необходимости их повторного освоения.

9.1 Образовательные технологии

Учебный процесс при преподавании курса основывается на использовании традиционных, инновационных и информационных образовательных технологий. Традиционные образовательные технологии представлены лекциями и семинарскими (практическими) занятиями. Инновационные образовательные технологии используются в виде широкого применения активных и интерактивных форм проведения занятий. Информационные образовательные технологии реализуются путем активизации самостоятельной работы студентов в информационной образовательной среде.

9.2 Занятия лекционного типа

Лекционный курс предполагает систематизированное изложение основных вопросов учебного плана.

На первой лекции лектор обязан предупредить студентов, применительно к какому базовому учебнику (учебникам, учебным пособиям) будет прочитан курс.

Лекционный курс должен давать наибольший объем информации и обеспечивать более глубокое понимание учебных вопросов при значительно меньшей затрате времени, чем это требуется большинству студентов на самостоятельное изучение материала.

9.3 Занятия семинарского типа

Семинарские занятия представляют собой детализацию лекционного теоретического материала, проводятся в целях закрепления курса и охватывают все основные разделы.

Основной формой проведения семинаров является обсуждение наиболее проблемных и сложных вопросов по отдельным темам, а также разбор примеров и ситуаций в аудиторных условиях. В обязанности преподавателя входят: оказание методической помощи и консультирование студентов по соответствующим темам курса.

Активность на семинарских занятиях оценивается по следующим критериям:

· ответы на вопросы, предлагаемые преподавателем;

· участие в дискуссиях;

· выполнение проектных и иных заданий;

· ассистирование преподавателю в проведении занятий.

Ответ должен быть аргументированным, развернутым, не односложным, содержать ссылки на источники.

Доклады и оппонирование докладов проверяют степень владения теоретическим материалом, а также корректность и строгость рассуждений.

Оценивание заданий, выполненных на семинарском занятии, входит в накопленную оценку.

9.4 Самостоятельная работа обучающихся по дисциплине (модулю)

Самостоятельная работа студентов – это процесс активного, целенаправленного приобретения студентом новых знаний, умений без непосредственного участия преподавателя, характеризующийся предметной направленностью, эффективным контролем и оценкой результатов деятельности обучающегося.

Цели самостоятельной работы:

· систематизация и закрепление полученных теоретических знаний и практических умений студентов;

· углубление и расширение теоретических знаний;

· формирование умений использовать нормативную и справочную документацию, специальную литературу;

· развитие познавательных способностей, активности студентов, ответственности и организованности;

· формирование самостоятельности мышления, творческой инициативы, способностей к саморазвитию, самосовершенствованию и самореализации;

· развитие исследовательских умений и академических навыков.

Самостоятельная работа может осуществляться индивидуально или группами студентов в зависимости от цели, объема, уровня сложности, конкретной тематики.

Технология организации самостоятельной работы студентов включает использование информационных и материально-технических ресурсов университета.

Контроль результатов внеаудиторной самостоятельной работы студентов может проходить в письменной, устной или смешанной форме.

Студенты должны подходить к самостоятельной работе как к наиважнейшему средству закрепления и развития теоретических знаний, выработке единства взглядов на отдельные вопросы курса, приобретения определенных навыков и использования профессиональной литературы.

9.5 Методические указания для обучающихся по освоению дисциплины

При изучении дисциплины обучающимся целесообразно выполнять следующие рекомендации:

1. Изучение учебной дисциплины должно вестись систематически.

2. После изучения какого-либо раздела по учебнику или конспектным материалам рекомендуется по памяти воспроизвести основные термины, определения, понятия раздела.

3. Особое внимание следует уделить выполнению отчетов по практическим занятиям и индивидуальным комплексным заданиям на самостоятельную работу.

4. Вся тематика вопросов, изучаемых самостоятельно, задается на лекциях преподавателем. Им же даются источники (в первую очередь вновь изданные в периодической научной литературе) для более детального понимания вопросов, озвученных на лекции.

При самостоятельной проработке курса обучающиеся должны:

· просматривать основные определения и факты;

· повторить законспектированный на лекционном занятии материал и дополнить его с учетом рекомендованной по данной теме литературы;

· изучить рекомендованную литературу, составлять тезисы, аннотации и конспекты наиболее важных моментов;

· самостоятельно выполнять задания, аналогичные предлагаемым на занятиях;

· использовать для самопроверки материалы фонда оценочных средств.

1. Методические указания при работе над конспектом лекции

В ходе лекционных занятий необходимо вести конспектирование учебного материала. Обращать внимание на категории, формулировки, раскрывающие содержание тех или иных явлений и процессов, научные выводы и практические рекомендации, положительный опыт в ораторском искусстве. Желательно оставить в рабочих конспектах поля, на которых делать пометки из рекомендованной литературы, дополняющие материал прослушанной лекции, а также подчеркивающие особую важность тех или иных теоретических положений. Задавать преподавателю уточняющие вопросы с целью уяснения теоретических положений, разрешения спорных ситуаций.

2. Методические указания по самостоятельной работе над изучаемым материалом и при подготовке к лабораторным занятиям

Начинать надо с изучения рекомендованной литературы. Необходимо помнить, что на лекции обычно рассматривается не весь материал, а только его часть. Остальная его часть восполняется в процессе самостоятельной работы. В связи с этим работа с рекомендованной литературой обязательна. Особое внимание при этом необходимо обратить на содержание основных положений и выводов, объяснение явлений и фактов, уяснение практического приложения рассматриваемых теоретических вопросов. В процессе этой работы необходимо стремиться понять и запомнить основные положения рассматриваемого материала, примеры, поясняющие его, а также разобраться в иллюстративном материале. Оформлять отчеты следует руководствуясь внутренними нормативными документами КнАГУ.

3. Методические указания для выполнения лабораторных работ

Wireshark – программное обеспечение для анализа сетевого трафика. Программа захватывает сетевые пакеты и отображает в наиболее полном виде их содержимое (данные).

Wireshark широко используется сетевыми администраторами для устранения неполадок; Инженеры по сетевой безопасности используют его для проверки проблем безопасности; Разработчики применяют данное ПО для проверки приложений, которые взаимодействуют с сетью; Также данный анализатор подходит для изучения работы внутренних сетевых протоколов.

Лабораторная работа №1

Исследование мультимедийного трафика

Цель работы: научится исследовать мультимедийный трафик в Wireshark.

Для достижения поставленной цели необходимо выполнить ряд задач:

- Воспроизвести аудиопоток, переданный через протокол SIP;
- Выполнить индивидуальный отчет о проделанной работе в соответствии с РД ФГБОУ ВО «КнАГУ».

Протокол установления сеанса (SIP, от англ. Session Initiation Protocol) – протокол передачи данных, описывающий способ установки и завершения пользовательского интернет-сеанса, включающего обмен мультимедийным содержимым (IP-телефония, видео- и аудиоконференции, мгновенные сообщения, онлайн-игры).

Протокол описывает, каким образом клиентское приложение (например, софтфон) может запросить начало соединения у другого, возможно, физически удалённого клиента, находящегося в той же сети, используя его уникальное имя. Протокол определяет способ согласования между клиентами об открытии каналов обмена на основе других протоколов, которые могут использоваться для непосредственной передачи информации (напри-

мер, RTP). Допускается добавление или удаление таких каналов в течение установленного сеанса, а также подключение и отключение дополнительных клиентов (то есть допускается участие в обмене более двух сторон – конференц-связь). Протокол также определяет порядок завершения сеанса.

Разработкой занималась организация IETF MMUSIC Working Group. Протокол начал разрабатываться в 1996 году Хенингом Шулзри (Henning Schulzrinne, Колумбийский университет) и Марком Хэндли (Университетский колледж Лондона). В ноябре 2000 года SIP был утверждён как сигнальный протокол проекта 3GPP и основной протокол архитектуры IMS (модификация 3GPP TS.24.229). Наряду с другим распространённым протоколом H.323, SIP – один из протоколов, лежащих в основе Voice over IP.

В основу протокола рабочая группа MMUSIC заложила следующие принципы:

- Простота: включает в себя только шесть методов (функций)
- Независимость от транспортного уровня, может использовать UDP, TCP, ATM и т. д.
- Персональная мобильность пользователей. Пользователи могут перемещаться в пределах сети без ограничений. Это достигается путём присвоения пользователю уникального идентификатора. При этом набор предоставляемых услуг остается неизменным. О своих перемещениях пользователь сообщает с помощью сообщения REGISTER своему серверу.
- Масштабируемость сети. Структура сети на базе протокола SIP позволяет легко её расширять и увеличивать число элементов.
- Расширяемость протокола. Протокол характеризуется возможностью дополнять его новыми функциями при появлении новых услуг.
- Интеграция в стек существующих протоколов Интернет. Протокол SIP является частью глобальной архитектуры мультимедиа, разработанной комитетом IETF. Кроме SIP, эта архитектура включает в себя протоколы RSVP, RTP, RTSP, SDP.
- Взаимодействие с другими протоколами сигнализации. Протокол SIP может быть использован совместно с другими протоколами IP-телефонии, протоколами ТфОП, и для связи с интеллектуальными сетями.

Клиенты SIP традиционно используют порт 5060 TCP или UDP для соединения элементов SIP-сети. В основном, SIP используется для установления и разъединения голосовых и видеозвонков. При этом он может использоваться и в любых других приложениях, где требуется установка соединения, таких, как системы оповещения, мобильные терминалы и так далее. Существует большое количество рекомендаций RFC, относящихся к SIP и определяющих поведение таких приложений. Для передачи самих голосовых и видеоданных используют другие транспортные протоколы, чаще всего RTP.

Главной задачей разработки SIP было создание сигнального протокола на базе IP, который мог бы поддерживать расширенный набор функций обработки вызова и услуг, представленных в существующей ТфОП. Сам протокол SIP не определяет этих функций, а сосредоточен только на процедурах регистрации пользователя, установления и завершения вызова и соответствующей сигнализации. При этом он был спроектирован с поддержкой таких функциональных элементов сети, как прокси-серверы (Proxy Servers) и Пользовательские Агенты (User Agents). Эти элементы обеспечивают базовый набор услуг: набор номера, вызов телефонного аппарата, звуковое информирование абонента о статусе вызова.

Телефонные сети на основе SIP могут поддерживать и более современные услуги, обычно предоставляемые ОКС-7, несмотря на значительное различие этих двух протоколов. ОКС-7 характеризуется сложной, централизованной интеллектуальной сетью и простыми, неинтеллектуальными, терминалами (традиционные телефонные аппараты). SIP – наоборот, требует очень простую (и, соответственно, хорошо масштабируемую) сеть с ин-

теллектом, встроенным в оконечные элементы на периферии (терминалы, построенные как физические устройства или программы).

SIP используется вместе с несколькими другими протоколами и участвует только в сигнальной части сессии связи. SIP выполняет роль носителя для SDP, который описывает параметры передачи медиаданных в рамках сессии, например используемые порты IP и кодеки. В типичном применении сессии SIP — это просто потоки пакетов RTP. RTP является непосредственным носителем голосовых и видео данных.

Первая предложенная версия стандарта (SIP 2.0) была определена в RFC 2543. Протокол был дополнительно уточнён в RFC 3261, хотя многие реализации по-прежнему основаны на промежуточных версиях стандарта. Обратите внимание, что номер версии остался 2.0.

По заданию необходимо на «Server» установить SIP-сервер. На «PC-1» и «PC-2» установить softphone.

«PC-1» выполняет звонок на «PC-2» через softphone.

Соберем лабораторный стенд в графическом сетевом эмуляторе GNS3 (рисунок 1).

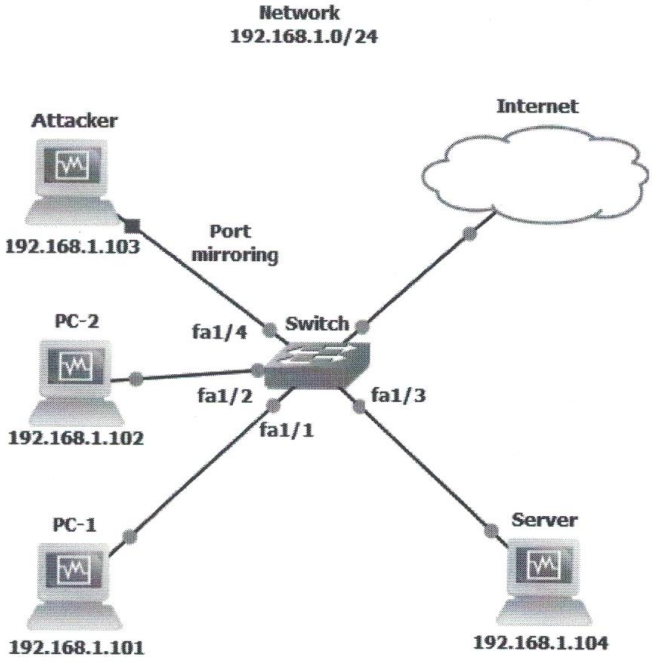

Рисунок 1 – Топология сети для выполнения задания

Лабораторная работа №2

Исследование данных, передаваемых через протокол TCP

Цель работы: восстановления информации, передаваемой по протоколам TCP.
Для достижения поставленной цели необходимо выполнить ряд **задач**:

- Восстановить информацию, переданную по протоколу SMB2;
- Восстановить информацию, переданную по протоколу FTP;

- Восстановить информацию, переданную по протоколу Telnet;
- Восстановить информацию, переданную по протоколу SSH;
- Восстановить информацию, переданную по протоколу HTTP;
- Восстановить информацию, переданную по протоколу HTTPS;
- Восстановить информацию, переданную по протоколу SMTP и POP3;
- Восстановить информацию, переданную по протоколу SMTP и IMAP;
- Восстановить информацию, переданную по протоколу DNS;
- Выполнить индивидуальный отчет о проделанной работе в соответствии с РД ФГБОУ ВО «КнАГУ».

Перехват трафика, передаваемого по протоколу TCP

Соберем лабораторный стенд в графическом сетевом эмуляторе GNS3 (рисунок 2).

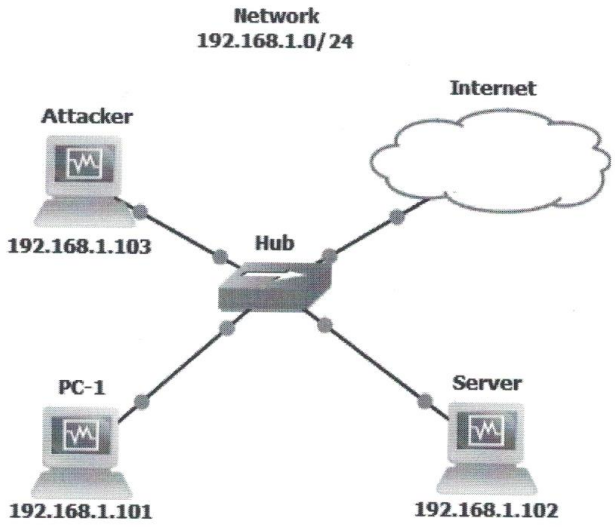

Рисунок 2 – Топология сети для выполнения задания

Протокол SMB (прикладной уровень по модели OSI) (от англ. Server Message Block – блок серверных сообщений) для удалённого доступа к файлам, принтерам и другим сетевым ресурсам, а также для межпроцессного взаимодействия. Вторая версия протокола (SMB2) появилась в Windows Vista, в основном применение данного протокола – реализация Сетей Microsoft Windows и Совместного использования файлов и принтеров.

В Windows Vista появилась новая версия протокола – SMB 2.0. Протокол был значительно упрощён (в SMB было более 100 команд, а в SMB 2 – всего 19); при этом была повышена производительность (благодаря механизму кэширования возможно совмещать несколько команд SMB 2 в одном сетевом запросе и увеличенным буферам чтения и записи), особенно в сетях с высокой латентностью, улучшена масштабируемость и добавлена возможность автоматического продолжения сеанса в случае временного отсоединения от сервера. SMB 2 использует тот же порт (445) как и SMB, но другой заголовок пакетов (0xFF 'S' 'M' 'B' в SMB, 0xFE 'S' 'M' 'B' в SMB 2).

SMB – это протокол, основанный на технологии клиент-сервер, который предоставляет клиентским приложениям простой способ для чтения и записи файлов,

а также запроса служб у серверных программ в различных типах сетевого окружения. Серверы предоставляют файловые системы и другие ресурсы (принтеры, почтовые сегменты, именованные каналы и т. д.) для общего доступа в сети. Клиентские компьютеры могут иметь у себя свои носители информации, но также имеют доступ к ресурсам, предоставленным сервером для общего пользования.

Клиенты соединяются с сервером, используя протоколы TCP/IP (а, точнее, NetBIOS через TCP/IP), NetBEUI или IPX/SPX. После того, как соединение установлено, клиенты могут посылать команды серверу (эти команды называются SMB-команды или SMBs), который даёт им доступ к ресурсам, позволяет открывать, читать файлы, писать в файлы и вообще выполнять весь перечень действий, которые можно выполнять с файловой системой. Однако в случае использования SMB эти действия совершаются через сеть.

Протокол FTP (прикладной уровень модели OSI) (от англ. File Transfer Protocol – Протокол передачи данных).

Клиент по протоколу FTP может подключиться к серверу и работать с его файловой системой (просматривать каталоги, переходить между ними, загружать файлы и т.д.).

TELNET (от англ. **tele**type **net**work) – протокол прикладного уровня для реализации текстового терминального интерфейса по сети.

Протокол является полностью симметричным, хотя в сессии Telnet и выделяется серверная и клиентская части. После установления транспортного соединения (как правило, TCP) оба его конца играют роль «сетевых виртуальных терминалов», обменивающихся двумя типами данных:

- Прикладными данными (то есть данными, которые идут от пользователя к текстовому приложению на стороне сервера и обратно);
- Командами протокола Telnet, частным случаем которых являются опции, служащие для уяснения возможностей и предпочтений сторон.

Прикладные данные проходят через протокол без изменений, то есть на выходе второго виртуального терминала мы видим именно то, что было введено на вход первого. С точки зрения протокола данные представляют просто последовательность байтов (октетов), по умолчанию принадлежащих набору ASCII.

В протоколе не предусмотрено использование ни шифрования, ни проверки подлинности данных. Поэтому он уязвим для любого вида атак, к которым уязвим его транспорт, то есть протокол TCP. Сессия Telnet весьма беззащитна, если только не осуществляется в полностью контролируемой сети или с применением защиты на сетевом уровне (различные реализации виртуальных частных сетей). По причине ненадёжности от Telnet как средства управления операционными системами давно отказались.

SSH (от англ. **S**ecure **Sh**ell — «безопасная оболочка») – сетевой протокол прикладного уровня. Используется для удаленного управления операционными системами и передачи файлов. Ключевая особенность заключается в том, что SSH шифрует трафик, делая подключения безопасными. По умолчанию, использует 22-й порт.

SSH позволяет безопасно передавать в незащищённой среде практически любой другой сетевой протокол. Таким образом, можно не только удалённо работать на компьютере через командную оболочку, но и передавать по шифрованному каналу звуковой поток или видео.

В SSH возможно использование различных видов аутентификации:

- Аутентификация по паролю наиболее распространена. При каждом подключении подобно https вырабатывается общий секретный ключ для шифрования трафика. При выполнении задания используется данный вид аутентификации.

- При аутентификации по ключевой паре предварительно генерируется пара открытого и закрытого ключей для определённого пользователя. На машине, с которой требуется произвести подключение, хранится закрытый ключ, а на удалённой машине – открытый. Эти файлы не передаются при аутентификации, система лишь проверяет, что владелец открытого ключа также владеет и закрытым. При данном подходе, как правило, настраивается автоматический вход от имени конкретного пользователя в ОС.
- Аутентификация по IP-адресу небезопасна, эту возможность чаще всего отключают.

HTTP (англ. HyperText Transfer Protocol – «протокол передачи гипертекста») – протокол прикладного уровня передачи данных изначально – в виде гипертекстовых документов в формате «HTML», в настоящий момент используется для передачи произвольных данных. Основой HTTP является технология «клиент-сервер», то есть предполагается существование:

- Потребителей (клиентов), которые инициируют соединение и посылают запрос;
- Поставщиков (серверов), которые ожидают соединения для получения запроса, производят необходимые действия и возвращают обратно сообщение с результатом.

Основным объектом манипуляции в HTTP является ресурс, на который указывает URI (Uniform Resource Identifier) в запросе клиента. Обычно такими ресурсами являются хранящиеся на сервере файлы, но ими могут быть логические объекты или что-то абстрактное. Особенностью протокола HTTP является возможность указать в запросе и ответе способ представления одного и того же ресурса по различным параметрам: формату, кодировке, языку и т. д. (в частности, для этого используется HTTP-заголовок). Именно благодаря возможности указания способа кодирования сообщения, клиент и сервер могут обмениваться двоичными данными, хотя данный протокол является текстовым.

HTTPS (от англ. HyperText Transfer Protocol Secure) – протокол передачи гипертекста защищенный. Протокол прикладного уровня модели OSI. Является обычным HTTP, который работает через шифрованные транспортные механизмы SSL и TLS. В отличие от HTTP с TCP-портом 80, для HTTPS по умолчанию используется TCP-порт 443.

Данный протокол обеспечивает защиту от атак, основанных на прослушивании сетевого соединения – от снифферских атак и атак типа man-in-the-middle, при условии, что будут использоваться шифрующие средства и сертификат сервера проверен и ему доверяют.

Чтобы подготовить веб-сервер для обработки https-соединений, администратор должен получить и установить в систему сертификат открытого и закрытого ключа для этого веб-сервера. В TLS используется как асимметричная схема шифрования (для выработки общего секретного ключа), так и симметричная (для обмена данными, зашифрованными общим ключом). Сертификат открытого ключа подтверждает принадлежность данного открытого ключа владельцу сайта. Сертификат открытого ключа и сам открытый ключ посылаются клиенту при установлении соединения; закрытый ключ используется для расшифровки сообщений от клиента.

В HTTPS для шифрования используется длина ключа 40, 56, 128 или 256 бит. Некоторые старые версии браузеров используют длину ключа 40 бит (пример тому – IE версий до 4.0), что связано с экспортными ограничениями в США. Длина ключа 40 бит не является сколько-нибудь надёжной. Многие современные сайты требуют использования новых версий браузеров, поддерживающих шифрование с длиной ключа 128 бит, с целью обеспечить достаточный уровень безопасности. Такое шифрование значительно затрудняет злоумышленнику поиск паролей и другой личной информации.

Так как HTTPS не является отдельным протоколом, а является обычным HTTP, работающим через шифрованные транспортные механизмы SSL и TLS, необходимо рассмотреть принцип работы TLS (SSL является устаревшим).

TLS (от англ. transport layer security – Протокол защиты транспортного уровня) – криптографический протокол, обеспечивающий защищенную передачу данных между узлами в сети Интернет. TLS и SSL используют асимметричное шифрование для аутентификации, симметричное шифрование для конфиденциальности и коды аутентичности сообщений для сохранения целостности сообщений.

Данный протокол широко используется в приложениях, работающих с сетью Интернет, таких как веб-браузеры, работа с электронной почтой, обмен мгновенными сообщениями и IP-телефония (VoIP).

TLS даёт возможность клиент-серверным приложениям осуществлять связь в сети таким образом, что нельзя производить прослушивание пакетов и осуществить несанкционированный доступ.

Основные шаги процедуры создания защищённого сеанса связи:

- клиент подключается к серверу, поддерживающему TLS, и запрашивает защищённое соединение;
- клиент предоставляет список поддерживаемых алгоритмов шифрования и хеш-функций;
- сервер выбирает из списка, предоставленного клиентом, наиболее надёжные алгоритмы среди тех, которые поддерживаются сервером, и сообщает о своём выборе клиенту;
- сервер отправляет клиенту цифровой сертификат для собственной аутентификации. Обычно цифровой сертификат содержит имя сервера, имя удостоверяющего центра сертификации и открытый ключ сервера;
- клиент, до начала передачи данных, проверяет валидность (аутентичность) полученного серверного сертификата относительно имеющихся у клиента корневых сертификатов удостоверяющих центров (центров сертификации). Клиент также может проверить, не отозван ли серверный сертификат, связавшись с сервисом доверенного удостоверяющего центра;
- для шифрования сессии используется сеансовый ключ. Получение общего секретного сеансового ключа клиентом и сервером проводится по протоколу Диффи-Хеллмана. Существует исторический метод передачи сгенерированного клиентом секрета на сервер при помощи шифрования асимметричной криптосистемой RSA (используется ключ из сертификата сервера). Данный метод не рекомендован, но иногда продолжает встречаться на практике.

По заданию «PC-1» заходит на сайт «https://vk.com» и выполняет успешную аутентификацию, затем «PC-1» отправляет сообщение через социальную сеть «ВКонтакте».

Установление TLS-соединения - многоступенчатый процесс, достаточно сложный и требующий проведения заметного количества вычислительных операций, особенно жадными в плане процессорного времени оказываются операции проверки подписей и других действий с асимметричными криптосистемами. Каждая итерация, связанная с отправкой запроса и получением ответа, вносит дополнительную задержку, а для массовых сервисов миллисекунды ожидания превращаются в весьма ощутимые затраты на оборудование. Для экономии ресурсов существует сокращённая версия Handshake.

В составе ServerHello сервер передаёт SessionID - идентификатор новой сессии. Этот идентификатор может быть использован клиентом позже, в составе ClientHello. Предполагается, что на стороне сервера (и клиента) сохранены необходимые параметры, которые могут быть быстро восстановлены, возобновив тем самым сессию.

В случае использования сокращённой схемы, сразу после получения ClientHello, содержащего валидный идентификатор сессии SessionID, сервер отвечает сообщениями ServerHello, ChangeCipherSpec и Finished (этот вариант, кстати, напоминает Handshake TLS 1.3). После того как клиент пришлёт свои CCS и Finished, сессия возобновляется и узлы могут начать обмен данными в защищённом режиме.

Сокращённый сценарий содержит существенно меньше сообщений, позволяет не использовать вычислительно затратные операции проверки подписей и генерации общего секрета, кроме того, из-за меньшего количества сообщений заметно снижается задержка при установлении соединения (экономится время, требуемое на отправку пакетов от клиента к серверу и обратно). Современные браузеры широко используют сокращённый Handshake. Сессии могут сохраняться на серверах в течение нескольких минут, нескольких часов, а возможно – дольше.

SMTP – это простой протокол передачи почты. Является протоколом прикладного уровня. С английского языка переводится, как Simple Mail Transfer Protocol. Исходя из названия, можно сделать вывод, что SMTP сервер отвечает за отправку почтовых рассылок. Его задача, как правило, состоит из двух основных функций:

- проверка правильности настроек и выдача разрешения компьютеру, который пытается отправить электронное сообщение;
- отправка исходящего сообщения на указанный адрес и подтверждение успешной отправки сообщения. Если доставка невозможна, сервер возвращает отправителю ответ с ошибкой отправки.

Отправляя email сообщения, SMTP-сервер отправителя устанавливает связь с тем сервером, который будет получать это сообщение. Такое "общение" происходит путем отправки и получения команд, формируя SMTP-сессию с неограниченным количеством SMTP-операций. Обязательными командами для каждой операции являются три:

- определение обратного адреса (MAILFROM);
- определение получателя email сообщения (RCPT TO);
- отправка текста сообщения (DATA).

Определение адреса отправителя, получателя и наличие содержимого письма – это обязательные условия, без которых письмо не будет отправлено.

Электронные почтовые серверы и другие агенты пересылки сообщений используют SMTP для отправки и получения почтовых сообщений, работающие на пользовательском уровне, клиентские почтовые приложения обычно используют SMTP только для отправки сообщений на почтовый сервер для ретрансляции.

В таблице 1 приведены команды SMTP

Таблица 1 Команды SMTP

№	Команда	Описание
1	ATRN	Эта команда выполняет операцию TURN (изменение ролей почтового клиента и сервера) только в том случае, если соединение проходит проверку подлинности.
2	CHUNKING	Это команда ESMTP, функция которой похожа на DATA (сообщение, отправленное клиентом на сервер, чтобы начать процесс отправки электронной почты), за исключением того, что механизм, используемый сервером для выяснения, где заканчивается сообщение, отличается; Количество байтов в сообщении явно отправляется, и сервер подсчитывает количество полученных байтов.
3	DATA	Эта команда дает серверу намек на то, что он готов начать отправку электронного письма.

4	DSN	Эта команда активирует получение уведомлений о состоянии доставки сообщения электронной почты.
5	EHLO	Клиент электронной почты, который поддерживает ESMTP, отправляет эту команду, и сервер возвращает список команд ESMTP, которые он поддерживает. Синтаксис этой команды требует, чтобы отправитель предоставил свое собственное доменное имя в качестве параметра.
6	ETRN	Эта команда отправляется почтовым сервером на другой сервер, требуя, чтобы он начал отправлять электронные сообщения, расположенные на нем.
7	HELP	Эта команда используется для запроса справки с сервера электронной почты и просит сервер вернуть список команд, которые он поддерживает.
8	HELO	Эта команда отправляется чтобы начать сеанс SMTP между отправителем и сервером, а также чтобы сервер мог идентифицировать отправителя. Синтаксис этой команды требует, чтобы отправитель отправил свое доменное имя в качестве параметра.
9	MAIL FROM	Эта команда используется чтобы начать процесс составления электронного письма и позволить серверу знать идентификатор отправителя, и его синтаксис требует обязательного использования адреса электронной почты отправителя в качестве параметра вместе с другими необязательными параметрами.
10	NOOP	Эта команда не выполняет никаких операций и используется исключительно для проверки возможности соединения с сервером.
11	PIPELINING	В терминале или командной строке пакеты ответов для каждого выполненного действия сразу видны отправителю. Эта команда включает конвейерную обработку, которая позволяет отправлять более одной команды за один прием до получения ответов.
12	QUIT	Эта команда используется для завершения сеанса SMTP.
13	RCPT TO	Синтаксис этой команды требует, чтобы адрес электронной почты получателя был предоставлен в качестве обязательного параметра, в дополнение к другим необязательным. Он используется для указания получателя сообщения электронной почты.
14	RSET	Эта команда используется для выполнения операции сброса; Текущий разговор завершается, или сообщение сбрасывается, и можно начинать заново.
15	SAML	Эта команда просит сервер отправить сообщение почтовому клиенту получателя, а также непосредственно на терминал получателя.
16	SEND	Эта команда используется для отправки сообщений электронной почты непосредственно на терминал получателя, а не отправки его клиенту электронной почты.
17	SIZE	Эта команда используется для указания размера отправляемого сообщения, в терминах количества байтов. Он вступает в действие, когда сервер установил ограничение на размер входящих сообщений электронной почты. В отсутствие этого явного объявления размера сообщения сервер пытается определить его с помощью других методов.
18	SOML	Эта команда отправляет электронные письма непосредственно на терминал получателя, если он или она находится в сети. В противном случае сообщение отправляется почтовому клиенту получателя.

19	TURN	Эта команда используется для инверсии функциональных возможностей почтового клиента и почтового сервера. Клиент может получать сообщения с сервера, используя то же соединение.
20	VRFY	В синтаксисе этой команды в качестве параметра принимается адрес электронной почты получателя. Эта команда запрашивает у сервера электронной почты проверку подлинности или существования удаленного пользователя с адреса электронной почты.

Для получения сообщений клиентские приложения обычно используют либо POP (англ. Post Office Protocol – протокол почтового отделения), либо IMAP (англ. Internet Message Access Protocol), либо патентованные системы (такие как Microsoft Exchange и Lotus Notes/Domino) для доступа к учётной записи своего почтового ящика на сервере.

POP3 (англ. Post Office Protocol Version 3 – протокол почтового отделения, версия 3) – стандартный интернет-протокол прикладного уровня, используемый клиентами электронной почты для получения почты с удалённого сервера по TCP-соединению.

В таблице 2 приведены команды протокола.

Таблица 2 Команды протокола POP3

Имя	**Аргументы**	**Ограничения**	**Возможные ответы**
APOP	[имя] [digest]	Её поддержка не является обязательной	* +OK maildrop has n message * -ERR password supplied for [имя] is incorrect
USER	[имя]	–	* +OK name is a valid mailbox * -ERR never heard of mailbox name
PASS	[пароль]	Работает после успешной передачи имени почтового ящика	* +OK maildrop locked and ready * -ERR invalid password * -ERR unable to lock maildrop
DELE	[сообщение]	Доступна после успешной аутентификации	* +OK message deleted * -ERR no such message
LIST	[сообщение]	Доступна после успешной аутентификации	* +OK scan listing follows * -ERR no such message
NOOP	–	Доступна после успешной аутентификации	+OK
RETR	[сообщение]	Доступна после успешной аутентификации	* +OK message follows * -ERR no such message
RSET	–	Доступна после успешной аутентификации	+OK
STAT	–	Доступна после успешной аутентификации	+OK a b

TOP	[сообщение] [количество строк]	Доступна после успешной аутентификации	* +OK n octets * -ERR no such message
QUIT	–	–	+OK

IMAP (от англ. Internet Message Access Protocol) – протокол прикладного уровня для доступа к электронной почте. Базируется на транспортном протоколе TCP и использует порт 143.

IMAP предоставляет пользователю широкие возможности для работы с почтовыми ящиками, находящимися на почтовом сервере. Почтовая программа, использующая этот протокол, получает доступ к хранилищу корреспонденции на сервере так, как будто эта корреспонденция расположена на компьютере получателя. Электронными письмами можно манипулировать с компьютера пользователя (клиента) без постоянной пересылки с сервера и обратно полного содержания писем.

Для отправки писем используется обычно протокол SMTP, так как собственная команда отправки протокола IMAP, называемая APPEND, не содержит в себе механизма передачи служебной информации.

Протокол предусматривает отслеживание состояния сообщения (прочитано, отправлен ответ, удалено и т. д.), благодаря системе флагов, данные о которых хранятся на сервере.

Рассмотрим основные команды IMAP:

- **LOGIN** Позволяет клиенту при регистрации на сервере IMAP использовать идентификатор пользователя и пароль в обычном текстовом виде.
- **AUTHENTICATE** Позволяет клиенту использовать при регистрации на сервере IMAP альтернативные методы проверки подлинности.
- **CLOSE** Закрывает почтовый ящик. Когда почтовый ящик закрыт с помощью этой команды, то сообщения, помеченные флагом \DELETED, удаляются из него.
- **LOGOUT** Завершает сеанс для текущего идентификатора пользователя.
- **CREATE** Создаёт новый почтовый ящик. Имя и местоположение новых почтовых ящиков определяются в соответствии с общими спецификациями сервера.
- **DELETE** Применяется к почтовым ящикам. Сервер IMAP при получении этой команды попытается удалить почтовый ящик с именем, указанным в качестве аргумента команды. Сообщения удаляются вместе с ящиками и восстановлению не подлежат.
- **RENAME** Изменяет имя почтового ящика. Эта команда имеет два параметра — имя почтового ящика, который требуется переименовать, и новое имя почтового ящика.
- **SUBSCRIBE** Добавляет почтовый ящик в список активных ящиков клиента. В этой команде используется только один параметр — имя почтового ящика, который нужно внести в список.
- **UNSUBSCRIBE** Удаляет почтовые ящики из списка активных. В ней так же используется один параметр — имя почтового ящика, который удаляется из списка активных ящиков клиента.
- **LIST** Получить список всех почтовых ящиков клиента.
- **LSUB** В отличие от команды LIST используется для получения списка ящиков, активизированных командой SUBSCRIBE.
- **STATUS** Формирует запрос о текущем состоянии почтового ящика. Первым параметром для этой команды является имя почтового ящика, к которому она применяется. Второй параметр — это список критериев, по которым клиент хочет получить информацию.

Пользователь может получить информацию по критериям:

- MESSAGES — общее число сообщений в почтовом ящике;
- RECENT — число сообщений с флагом \recent;
- UIDNEXT — идентификатор UID, который будет назначен новому сообщению;
- UIDVALIDITY — уникальный идентификатор почтового ящика;
- UNSEEN — число сообщений без флага \seen.

- **APPEND** Добавляет сообщение в конец указанного почтового ящика.
- **CHECK** Устанавливает контрольную точку в почтовом ящике. Любые операции, такие, например, как запись данных из памяти сервера на его жёсткий диск, должны выполняться при соответствующем состоянии почтового ящика. Именно для проверки целостности почтового ящика после дисковых и других подобных им операций и применяется команда CHECK.
- **EXPUNGE** Удаляет из почтового ящика все сообщения, помеченные флагом \DELETED, при этом почтовый ящик не закрывается.
- **SEARCH** Поиск сообщений по критериям в активном почтовом ящике с последующим отображением результатов в виде номера сообщения.
- **FETCH** Получить текст почтового сообщения. Команда применяется только для отображения сообщений. В отличие от POP3, клиент IMAP не сохраняет копию сообщения на клиентском ПК.
- **STORE** Изменяет информацию о сообщении.
- **COPY** Копирует сообщения из одного почтового ящика в другой.
- **UID** Используется в связке с командами **FETCH**, **COPY**, **STORE** или SEARCH. С её помощью в этих командах можно использовать реальные идентификационные номера UID вместо последовательности чисел из диапазона номеров сообщений.
- **CAPABILITY** Запрос у сервера IMAP информации о его возможностях.
- **NOOP** Команда ничего не делает. Она может применяться для поддержки активности во время сеанса для того, чтобы сеанс не прекратился по таймеру интервала ожидания. Ответ сервера на команду NOOP всегда должен быть положительным.

DNS (англ. Domain Name System «система доменных имён») – компьютерная распределённая система для получения информации о доменах. Протокол прикладного уровня, работает на порту 53 TCP/UDP. Чаще всего используется для получения IP-адреса по имени хоста (компьютера или устройства), получения информации о маршрутизации почты и/или обслуживающих узлах для протоколов в домене (SRV-запись).

Основой DNS является представление об иерархической структуре имени и зонах. Каждый сервер, отвечающий за имя, может делегировать ответственность за дальнейшую часть домена другому серверу (с административной точки зрения – другой организации или человеку), что позволяет возложить ответственность за актуальность информации на серверы различных организаций (людей), отвечающих только за «свою» часть доменного имени.

Wireshark – это приложение, которое может работать со структурой самых различных сетевых протоколов, и поэтому позволяет разобрать сетевой пакет, отображая значение каждого поля протокола любого уровня. Данная программа позволяет сетевой плате принимать все пакеты вне зависимости от того, кому они адресованы. Распространение ПО под свободной лицензией GNU GPL и наличие удобного графического интерфейса делают её отличным выбором для изучения работы сетевых протоколов и передаваемых пакетов.

Лабораторная работа №3

Исследование файловых систем, операционных систем

Восстановление данных – процесс извлечения различных файлов и информации с любого компьютерного носителя, при условии, что эта информация не может быть прочитана обыкновенным способом. Потребность в восстановлении информации может возникнуть при программном или аппаратном (физическом) повреждении носителя.

В настоящий момент существует два основных способа восстановления утерянной информации:

1) Программный. Этот способ позволяет вернуть информацию при сохраненной работоспособности самого накопителя. Этот способ позволяет восстановить данные без физического вмешательства в устройство носителя, а также его применение не затрагивает микропрограммы и структуру модулей служебной информации накопителя. Причинами в потребности к восстановлению информации данным способом могут послужить форматирование разделов жесткого диска, частичное/полное удаление данных, неудачное форматирование, полное/частичное разрушение файловой системы.

2) Программно-аппаратный. Данный способ восстановления применяется при физическом повреждении накопителя. Для каждого типа носителя (накопитель на гибком магнитном диске, носители CD/DVD/BR, NAND-Flash, накопитель на жестком магнитном диске и тому подобное) применяется свой способ восстановления.

Жесткий диск — основное устройство хранения данных. Он может иметь различную структуру хранения файлов и каталогов, которая обеспечивает непосредственное расположение данных на диске. Файловая система чаще всего интегрирована в операционную, а некоторые операционные системы поддерживают несколько файловых систем.

Цель работы: познакомится с различными файловыми системами и научится извлекать из них информацию.

Для достижения поставленной цели необходимо выполнить ряд **задач**:

1. Исследовать структуру файловых систем FAT32 и NTFS.
2. В ручном режиме продемонстрировать удаление и восстановление информации в файловых системах FAT32 и NTFS.
3. Протестировать различное ПО для восстановления файлов (5 шт).
4. Протестировать различное ПО для надежного удаления файлов (2 шт).

Выполнить индивидуальный отчет о проделанной работе в соответствии с РД ФГБОУ ВО «КнАГУ».

Хранение информации, структура хранения информации на жестком диске

Самой маленькой единицей информации, которую использует система управления жестким диском называется сектором. В большинстве современных носителей размер сектора равняется 512 б. Нулевой сектор жесткого диска называется MBR – главная загрузочная запись, так же в нулевом секторе находится таблица разделов. В эту таблицу заносится не больше четырех записей с номером начального сектора раздела и его размером.

Раздел, запись о котором находится в таблице разделов нулевого сектора, называется первичным. При необходимости использовании больше четырех разделов, в таблицу разделов вносится запись о расширенном разделе. Этот тип раздела является контейнером, в котором может создаваться неограниченное количество томов, однако в операционной системе Windows количество томов может быть ограничено количеством букв латинского алфавита.

Раздел – это размеченное пространство на жестком диске, созданное для того, чтобы хранить в нем информацию. Для организации структуры хранения файлов должна существовать файловая система. В операционной системе Windows зачастую используются файловые системы NTFS и FAT32, оперирующие более крупными структурами данных на диске, которые называются кластерами.

Кластер – объединение нескольких однородных элементов. В нашем случае – секторов жесткого диска. Это минимальная единица размещения информации на носителе, состоящая из двух или большего количества секторов дорожки. Например, если мы запишем на диск файл размером 20 байт, то на диске он все равно займет 1024 байт. Кластер может включать в себя произвольное число секторов от 1 (512 байт) до 128 (64 кбайт).

Файловая система – система, определяющая способ хранения, организации и именования данных, записанных на носитель информации в различном оборудовании. Мы рассмотрим две наиболее часто встречающиеся и используемые системы организации информации – это FAT32 и NTFS. Ниже приведена таблица сравнения файловых систем (Таблица 3).

Таблица 3 Сравнение файловых систем

Характеристика	FAT 32	NTFS
Максимальное количество файлов	268 435 444	4 294 967 295
Поддержка сжатия	Не поддерживается	На уровне файловой системы для файлов, каталогов и дисков
Аудит	Не поддерживается	С использованием SACL
Средства безопасности	Атрибуты файлов	Атрибуты файлов авторизации с использованием DACL, шифрование с использованием EFS
Максимальный размер файла	4 Гб	~16 384 ГБ или ~16 Тб
Поддержка ссылок различных типов	Не поддерживает	Жесткие и символьные ссылки, соединения для каталогов
Максимальный размер тома	127.53 Гб	$9.4*10^{12}$ Гб или 9.4 ЗБ
Размер диска	8 000 Гб или 8 Тб	18 446 744 073,7 Гб или 16 экзабайт

Каждый диск имеет две области: Каталог и область хранения информации. Каталог содержит в себе указание на начало размещение файла на диске и его имя. В операционной системе Windows имя файла может иметь максимальную длину в 255 символов. Существуют одноуровневые и многоуровневые файловые системы. Для удобства поиска, при хранении на диске тысяч файлов, используется многоуровневая иерархическая система, которая имеет древовидную структуру. Пример такой системы представлен на рисунке 3.

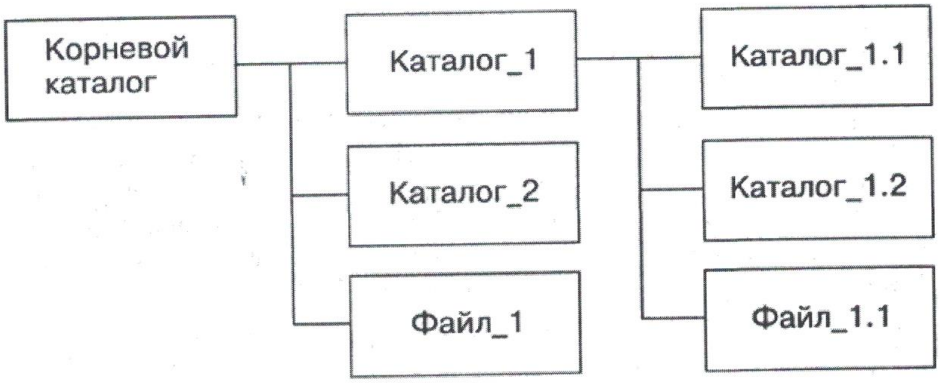

Рисунок 3 – Иерархическая файловая система

В файловой системе NTFS при удалении файла, его имя исключается из индекса родительского каталога. Происходит пересортировка индекса, в результате чего имя файла может быть утеряно. От сюда следует, что имя удаленного файла больше не будет находиться в исходном каталоге. В дальнейшем свободные записи MFT (главная файловая таблица) и кластеры, которые хранили удаленный файл могут быть перезаписаны при записи на диск других файлов, поэтому пока запись не производилась, все удаление файла сводится к изменению его атрибутов, что является возвратимой операцией.

Для того, чтобы восстановить удаленный файл в файловой системе NTFS, необходимо просканировать MFT на предмет свободных записей. При обнаружении такой записи возможно определить имя удаленного файла, которое хранится в одном из атрибутов, но его удается определить не всегда. После, по сохранившемся указателям на кластеры, которые занимает удаленный файл удается восстановить его. Такая операция может быть успешной при условии, что занимаемые удаленным файлом кластеры не были перезаписаны другим файлом.

Загрузочный сектор (boot sector) – первый сектор тома, в котором хранятся параметры файловой системы. Также в загрузочный сектор входит код, который отвечает за чтение файлов в память. Такой код несет смысл только если том является системным (т. е. только для диска C). В процессе монтирования тома, ntfs.sys проводит валидацию загрузочного сектора и признает его своим в случае совпадение необходимых параметров.

При удалении файла в файловой системе FAT32, первый символ имени файла заменяется специальным кодом 0xE5. После чего цепочка кластеров, которые занимает данный файл в таблице размещения обнуляется. Так как информация о размере файла, располагающаяся рядом с именем файла, при данной операции остается нетронутой, то в случае, если кластеры файла располагались на диске последовательно, и они не были перезаписаны иной информацией, возможно восстановление файла.

Recuva – Утилита для восстановления случайно удаленных файлов с жестких дисков компьютеров и съемных носителей от компании Piriform. Recuva имеет довольно понятный и удобный интерфейс.

R-Studio – Полноценная утилита для восстановления утерянной информации, основанная на уникальной технологии анализа данных на носителе. Присутствует возможность использования утилиты на удаленных компьютерах. Так же, в R-Studio реализована возможность восстановления информации по сигнатурам для сильно поврежденных или неизвестных файловых систем.

Disk Drill – Программа для восстановления утерянных и удаленных файлов. Имеет возможность восстановления разделов жесткого диска, а также медиа-файлов. Программа позволяет восстановить файлы в случае сбоя жесткого диска, повторного разбиения жесткого диска на разделы и в других подобных ситуациях. Так же имеет возможность предварительного просмотра файлов перед восстановлением.

R.saver – Программа для восстановления файлов с различных файловых систем, а также для копирования файлов с не поддерживаемых в Windows файловых систем от компании R.LAB. Использует полнофункциональные алгоритмы версий UFS Explorer.

WinHex – HEX-редактор, позволяющий работать с жесткими дисками, дискетами, CD/DVD и другими устройствами. Программа может клонировать диски, надежно удалять данные без возможности удаления, имеет редактор оперативной памяти, поддерживает файлы размером более 4 ГБ.

Privazer – программа, предназначенная для очистки персонального компьютера. Позволяет удалять следы Интернет-активности, неиспользуемые ярлыки, освобождать дисковое пространство, надежно удалять файлы и многое другое.

Eraser – инструмент безопасности для Windows. Позволяет полностью удалить информацию путем многократной перезаписи.

Лабораторная работа №4

Исследование ОЗУ

Форензика памяти – криминалистический анализ дампа памяти компьютера.

Его основное применение – расследование сложных компьютерных атак, которые достаточно скрытны, чтобы не оставлять данные на жестком диске компьютера.

Следовательно, память (RAM) должна анализироваться на предмет криминалистической информации.

Цель работы: познакомится с различными инструментами для анализа ОЗУ.

Для достижения поставленной цели необходимо выполнить ряд **задач**:

5. Исследовать инструменты для анализа ОЗУ.
6. Восстановить информацию о процессах из дампа ОЗУ.

Выполнить индивидуальный отчет о проделанной работе в соответствии с РД ФГБОУ ВО «КнАГУ».

Получение данных о компьютере

Воспользуемся инструментом Volatility Framework для просмотра имени компьютера. Команда приведена в листинге 1

Листинг 1 – Просмотр имени компьютера из дампа ОЗУ

```
volatility -f <memory_image> –profile=<profile> envars | findstr COMPUTERNAME
```

С помощью volatility также можно просмотреть профиль. Команда для просмотра профиля приведена в листинге 2.

Листинг 2 – Просмотр профиля volatiltiy

```
volatility --info
```

В Windows имеется множество переменных среды для запуска процессов, которые могут извлекать справочные данные, такие как ОС, TEMP, windir, Path и т.д. и используемое в настоящее время имя хоста будет храниться в переменной с именем COMPUTERNAME.

Переменные среды можно посмотреть так же с помощь PowerShell.

Листинг 3 – Просмотр переменных среды

```
Get-ChildItem Env
```

Работа с процессами

Воспользуемся инструментом Volatility Framework для просмотра процессов из дампа памяти. Команда приведена в листинге 4.

Листинг 4 – Просмотр процесса из дампа ОЗУ

```
volatility -f <file_image> --profile==<profile_name> pslist
```

При использовании аргумента pslist volatility framework находит и просматривает двусвязный список процессов и выводит сводку данных. Этот метод обычно не может показать вам завершенные или скрытые процессы.

При использовании аргумента pstree volatility framework берет вывод из pslist и форматирует его в виде дерева, чтобы вы могли легко увидеть родительские и дочерние отношения.

При использовании аргумента psscan volatility framework сканирует объекты _EPROCESS, а не полагается на связанный список. Этот плагин также может найти завершенные и несвязанные (скрытые) процессы.

При использовании аргумента psxview volatility framework находит процессы, используя списки альтернативных процессов, поэтому вы можете ссылаться на разные источники информации и выявлять вредоносные несоответствия.

Практическое использование

Воспользуемся volatility framework на примере получения пароля от учетной записи пользователя. Наш дамп называется ch2.dmp. Первым делом идентифицируем профиль дампа.

Листинг 5 – Получение профиля дампа

```
volatility -f ch2.dmp imageinfo
```

Воспользуемся инструментом Volatility Framework для перечисления хайва реестра.

Листинг 6 – Просмотр переменных среды

```
volatility -f ch2.dmp --profile=Win7SP1x86 hivelist
```

Теперь с помощью виртуального смещения SYSTEM и SAM мы можем извлечь хэши.

Листинг 7 – Извлечение хешей

```
volatility -f ch2.dmp --profile=Win7SP1x86 hashdump -y 0x8b21c008 -s 0x9aad6148 > hash-
```

```
es.txt
```

Теперь найдем хеш пароля пользователя John Doe.

Листинг 8 – Считывание файла

```
cat hashes.txt
```

4. Методические указания по выполнению расчетно-графической работы

Теоретическая часть расчетно-графической работы выполняется по установленным темам с использованием практических материалов. К каждой теме расчетно-графической работы рекомендуется примерный перечень узловых вопросов, список необходимой литературы. Излагая вопросы темы, следует строго придерживаться плана. Работа не должна представлять пересказ отдельных глав учебника или учебного пособия. Необходимо изложить собственные соображения по существу излагаемых вопросов, внести свои предложения. Общие положения должны быть подкреплены и пояснены конкретными примерами. Излагаемый материал при необходимости следует проиллюстрировать таблицами, схемами, диаграммами.

РГР

Сканирование сети

Цель сканирования сети состоит в том, чтобы выяснить, какие компьютеры подключены к сети и какие сетевые сервисы на них запущены.

Первая задача решается путем посылки Echo-сообщений протокола ICMP с помощью утилиты ping с последовательным перебором всех адресов сети или отправкой Echo-сообщения по широковещательному адресу.

Анализируя трафик и отслеживая Echo-сообщения, посылаемые за короткий промежуток времени всем узлом, можно выявит попытки сканирования. Чтобы не дать себя раскрыть злоумышленник может растянуть отправку сообщений во времени. Вместо Echo-сообщений могут применяться TCP-сегменты с code bit RST, ответы на несуществующие DNS-запросы. Если злоумышленник получит в ответ ICMP Destination Unreachable пакет с кодом 1 (host unreachable), то значит, тестируемый узел выключен или не подключен к сети.

Чтобы определить, какие сервисы запущены, нужно узнать, какие порты открыты, так как существует набор сервисов, за которыми закреплены определенные порты (TCP-порты). Далее эту информацию можно использовать для осуществления атаки на более высоком уровне. Для сканирования TCP-портов существует несколько способов. Самый простой – установление TCP-соединений с тестируемым портом. В этом случае появляется большое количество открытых и сразу прерванных соединений, поэтому атаку в такой реализации просто обнаружить. Другой способ, так называемый half-open scanning (сканирование в режиме половинного открытия). В этом режиме злоумышленник отправляет сегмент с code bit SYN на тестируемый порт и ждёт ответа. Если в качестве ответа пришёл сегмент с code bit RST, то это значит, что порт закрыт, а если сегмент с code bit SYN, ACK– порт открыт. Тогда злоумышленник отправит на этот порт сегмент с флагом RST. Так как соединение не было открыто, то обнаружить это сканирование гораздо сложнее.

И еще один способ заключается в том, чтобы отправлять сегменты с флагами FIN(no more data from sender), PSH(push function), URG(urgent pointer field significant) либо вообще с пустым полем code bit. Если порт закрыт, то в ответ придёт сегмент с флагом RST, если ответа не будет, то порт открыт (так как такой сегмент просто игнорируется).

Цель работы: познакомится и реализовать на практике некоторые виды сетевого сканирования.

Для достижения поставленной цели необходимо выполнить ряд **задач**:

1) Собрать лабораторный стенд, включающий в себя:
   - Windows Server, с каким-либо сервисом, например, веб-сайт.
   - Linux Server, с каким-либо сервисом, например, FTP сервер.
   - Компьютер злоумышленника.
   - Компьютер клиента.
2) Установить сканер сети «nmap» или «Zenmap» на компьютер злоумышленника.
3) Используя сканер сети выполнить все виды сканирования согласно таблице 1. Проанализировать полученные данные. При сканировании осуществляйте перехват трафика через сниффер «Wireshark». Проанализировать полученные данные в снифферe.
4) Выполнить индивидуальный отчет о проделанной работе в соответствии с РД ФГБОУ ВО «КнАГУ».

Таблица 4 Сканирование сети

п/п	Тип сканирования	Задание	
	ICMP Scanning		Выполнить ICMP Scanning Windows Server.
			Выполнить ICMP Scanning Linux Server.
	Ping Sweep	Выполнить Ping Sweep, на всю сеть.	
	TCP Connect / Full Open Scan		Выполнить TCP Connect / Full Open Scan Windows Server.
			Выполнить TCP Connect / Full Open Scan Linux Server.
	Stealth Scan / Half Open Scan		Выполнить Stealth Scan / Half Open Scan Windows Server.
			Выполнить Stealth Scan / Half Open Scan Linux Server.
	Xmas Scan		Выполнить Xmas Scan Windows Server.
			Выполнить Xmas Scan Linux Server.
	Fin Scan		Выполнить Fin Scan Windows Server.
			Выполнить Fin Scan Linux Server.
	NULL Scan		Выполнить NULL Scan Windows Server.
			Выполнить NULL Scan Linux Server.
	ACK Flag Probe Scanning		Выполнить ACK Flag Probe Scanning Windows Server.
			Выполнить ACK Flag Probe Scanning Linux Server.
	IDLE / IPID Header Scan		Выполнить IDLE / IPID Header Scan Windows Server.
			Выполнить IDLE / IPID Header Scan Linux Server.
	UDP Scanning		Выполнить UDP Scanning Windows Serv-

			er.
			Выполнить UDP Scanning Linux Server.
	Banner Grabbing		Выполнить Banner Grabbing Windows Server.
			Выполнить Banner Grabbing Linux Server.
	Drawing Network Diagrams	Выполнить Drawing Network Diagrams.	

Общие сведения об ICMP Scanning

Задачи администраторов по обнаружению работающих хостов в сети могут быть удовлетворены обычным ICMP пингом. Задачу обнаружения хостов иногда называют пинг сканированием (ping scan).

Выполнение ICMP Scanning

Для выполнения лабораторной работы соберем стенд в графическом сетевом симуляторе GNS3 (рисунок 4).

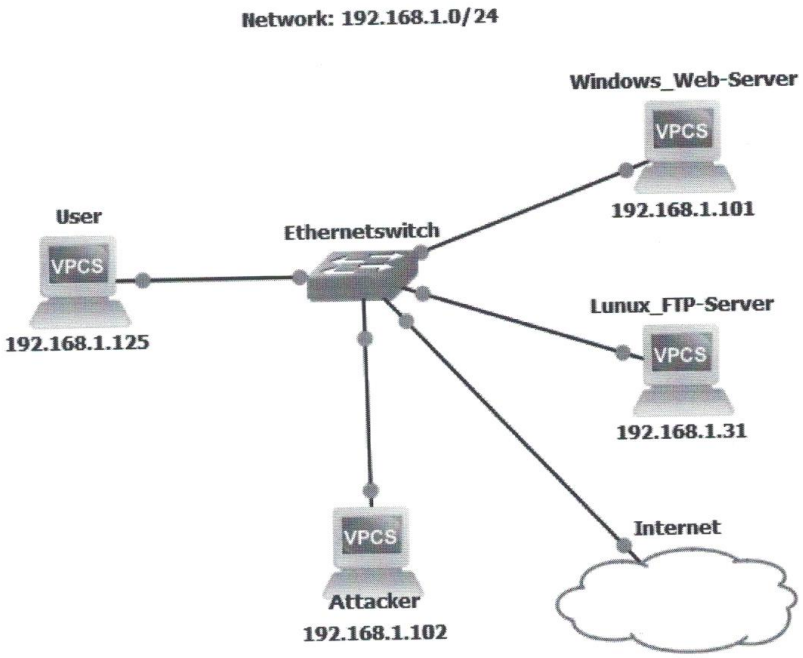

Рисунок 4 – Лабораторный стенд для выполнения работы

Web-server развернут на Windows Server 2012, FTP-server работает на Debian.

1. Выполнить ICMP Scanning сервера с Windows;
2. Выполнить ICMP Scanning сервера с Debian

Для выполнения задания используем Kali Linux и утилиту Nmap.

Ping sweep используется для определения доступных хостов в диапазоне IP-адресов, путем отправки ICMP ECHO запросов множеству хостов. Если хост доступен, то он вернет ICMP ECHO ответ.

Сканирование TCP Connect определяет открыт ли порт, совершая процедуру трехэтапного рукопожатия. Сканирование TCP Connect создает полное соединение и сразу же сбрасывает его, посылая RST-пакет.

Stealth Scan сбрасывает TCP соединение между клиентом и сервером до того момента, как завершится трехэтапное рукопожатие. Таким образом соединение остается полуоткрытым.

Атакующие используют технику Stealth Scan, чтобы обойти правила файерволов, механизмы логгирования и выдать себя за обычный сетевой трафик.

Общие сведения о Xmax Scan

При Xmas сканировании атакующий посылает TCP-кадр удаленному устройству, в котором установлены флаги FIN, URG и PUSH.

Отдельно FIN сканирование работает только с операционными системами, в который TCP/IP-стек реализован по RFC 793.

Данный вид сканирования не будет работать против любой текущей версии Microsoft Windows.

Общие сведения о Fin Scan

Некоторые серверы способны отследить попытку SYN-сканирования их портов. Например, попытка SYN-сканирования может быть распознана по поступлению «поддельных» SYN-пакетов на закрытые порты защищаемого сервера, и в случае опроса нескольких портов сервер разрывает соединение для защиты от сканирования.

Общие сведения о NULL Scan

Единственное отличие данного вида сканирования от Fin Scan в том, что при NULL сканировании не устанавливается вообще никаких флагов. Однако поведение отвечающих узлов остается прежним.

Общие сведения об ACK Flag Probe Scanning

Атакующие посылают зондирующие TCP-пакеты с установленным ACK-флагом удаленному хосту и затем анализируют информацию заголовков (TTL и WINDOW поля) пришедших RST-пакетов, чтобы определить «открыт» или «закрыт» порт.

- Если значение TTL пришедшего RST-пакета с определенного порта меньше границы в 64, тогда порт «открыт»;
- Если значение WINDOW пришедшего RST-пакета с определенного порта имеет не нулевое значение, тогда порт «открыт».

Общие сведения о UDP Scanning
В протоколе UDP нет процедуры трехэтапного рукопожатия. Система не отвечает, когда порт открыт.

Если UDP-пакет отправлен на закрытый порт, то система отвечает сообщением ICMP port unreachable.

Общие сведения о Banner Grabbing
Banner grabbing – это метод определения операционной системы, запущенной на целевой системе. Существует два типа banner grabbing: активный и пассивный.

Активный banner grabbing:

- Специально сформированные пакеты посылаются на удаленный хост и ответы на них запоминаются.
- Затем эти ответы сравниваются с ответами из базы данных, в которой находятся ответы на подобные запросы от определенной ОС.
- Ответы от разных ОС различаются из-за реализации TCP/IP стека протоколов в системе.

Пассивный banner grabbing:

- Banner grabbing на основе сообщений об ошибках. Сообщения об ошибках представляют информацию о типе сервиса, типе ОС и SSL утилит, используемых целевым удаленным хостом.
- Прослушка сетевого траффика. Сбор и анализ пакетов от цели позволяет атакующему определить ОС на удаленном хосте.
- Banner grabbing по расширениям страниц. Просмотр расширений URL может помочь в определении версии приложения. Например: .aspx => IIS сервер и платформа Windows.

Определение ОС, работающей на целевом хосте, позволяет атакующему определить уязвимости системного уровня и эксплоиты, которые могут работать на системе для проведения дальнейших атак.

Выполнение Banner Grabbing

Общие сведения о Drawing Network Diagrams
Рисование целевых сетевых диаграмм дает значительное количество информации о сети и ее архитектуре злоумышленнику.

Сетевая диаграмма показывает логический и физический путь к потенциальной цели.

Сканирование сети имеет своей целью выявление подключенных к сети компьютеров и определение работающих на них сетевых сервисов (открытых портов TCP или UDP). Первая задача выполняется посылкой ICMP-сообщений Echo с помощью программы ping с последовательным перебором адресов узлов в сети. Стоит попробовать отправить Echo-сообщение по широковещательному адресу – на него ответят все компьютеры, поддерживающие обработку таких сообщений.
Чтобы определить какие UDP или TCP-приложения запущены на обнаруженных компьютерах, используются программы-сканеры, например, программа nmap (или zenmap – вариант с GUI). Поскольку номера портов всех основных сервисов Интернета стандартизованы, то, определив, например, что порт 25/TCP открыт, можно сделать вывод о том, что данный хост является сервером электронной почты, и т. д. Полученную информацию злоумышленник может использовать для развертывания атаки на уровне приложения.

При выполнении работы производилось сканирование Windows- и Debian-сервера, чтобы выявить принципиальные различия в поведении данных систем во время проведения сканирования.

10 Описание материально-технического обеспечения, необходимого для осуществления образовательного процесса по дисциплине (модулю)

10.1 Учебно-лабораторное оборудование

Таблица 6 – Перечень оборудования лаборатории

Аудитория	Наименование аудитории (лаборатории)	Используемое оборудование
202/5	Лаборатория программно-аппаратных средств защиты информации	СЗИ НСД Secret Net, СЗИ НСД Dallas Lock, СЗИ НСД Страж NT, СЗИ НСД Щит РЖД, СЗИ НСД Аура ,СЗИ НСД Криптон ,СЗИ НСД Аккорд, ФИКС, Ревизор 1,2 как для операционных систем семейства Windows так и для Linux, Ревизор Сети 2.0, Анализатор сетевого траффика Астра,Агент инвентаризации сети,Сканер сетевой безопасности XSpider, Терьер, Secret Net Touch Memory Card, Криптон АМДЗ, Аккорд АМДЗ, КриптоПРО АРМ, ,CryptoPro CSP 3.6, VipNet firewall, Etoken PKI Client, Etoken, Ноутбук с Windows 7+проектор. 16 ПЭВМ на базе процессоров не ниже Intel Pentium IV
201/5	Лаборатория технических средств и методов защиты информации	специализированное оборудование по защите информации от утечки по акустическому каналу и каналу побочных электромагнитных излучений и наводок: Соната АВ с оконечными устройствами (виброизлучатели, акустические излучатели), генератор шума электромагнитного поля ВетоМ, генератор ЛГШ 503, генератор Соната РС-1 Технические средства контроля эффективности защиты информации от утечки по указан-

		ным каналам: Комплект измерительных антенн Альбатрос 3, селективный микровольтмерт SMV 8,5, селективный микровольтметр SMV 11, комплекс Спрут-мини-А в комплекте с программным обеспечением, Unipan 233, ПЭВМ семества Secret, Поисковый прибор ST033P Пиранья в комплекте с програмнным обеспечением. иное дополнительное оборудование: нелинейный локатор NR-m, генератор сигналов АКИП 3410, комплект измерительных антенн Альбатрос, пробник напряжения СРФ-1, антенны DP-1 и DP-3, генераторы сигналов серии Г3 и Г4. Комплект тестовых программ Зебра для Windows, для МСВС лицензия номер 592

10.2 Технические и электронные средства обучения

Лекционные занятия

Аудитории для лекционных занятий укомплектованы мебелью и техническими средствами обучения, служащими для представления учебной информации большой аудитории (наборы демонстрационного оборудования (проектор, экран, компьютер/ноутбук), учебно-наглядные пособия, тематические иллюстрации).

Лабораторные занятия

Для лабораторных занятий используется аудитория № 201 , оснащенная оборудованием, указанным в табл. 8:

Самостоятельная работа.

Помещения для самостоятельной работы оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и доступом к электронной информационно-образовательной среде КнАГУ:

- читальный зал НТБ КнАГУ;
- компьютерные классы (ауд. 311 корпус № 5, ауд. 205 корпус № 5, ауд. 313 корпус № 5).

11 Иные сведения

Методические рекомендации по обучению лиц с ограниченными возможностями здоровья и инвалидов

Освоение дисциплины обучающимися с ограниченными возможностями здоровья может быть организовано как совместно с другими обучающимися, так и в отдельных группах. Предполагаются специальные условия для получения образования обучающимися с ограниченными возможностями здоровья.

Профессорско-педагогический состав знакомится с психолого-физиологическими особенностями обучающихся инвалидов и лиц с ограниченными возможностями здоровья, индивидуальными программами реабилитации инвалидов (при наличии). При необходимости осуществляется дополнительная поддержка преподавания тьюторами, психологами, социальными работниками, прошедшими подготовку ассистентами.

В соответствии с методическими рекомендациями Минобрнауки РФ (утв. 8 апреля 2014 г. N АК-44/05вн) в курсе предполагается использовать социально-активные и рефлексивные методы обучения, технологии социокультурной реабилитации с целью оказания помощи в установлении полноценных межличностных отношений с другими студен-

тами, создании комфортного психологического климата в студенческой группе. Подбор и разработка учебных материалов производятся с учетом предоставления материала в различных формах: аудиальной, визуальной, с использованием специальных технических средств и информационных систем.

Освоение дисциплины лицами с ОВЗ осуществляется с использованием средств обучения общего и специального назначения (персонального и коллективного использования). Материально-техническое обеспечение предусматривает приспособление аудиторий к нуждам лиц с ОВЗ.

Форма проведения аттестации для студентов-инвалидов устанавливается с учетом индивидуальных психофизических особенностей. Для студентов с ОВЗ предусматривается доступная форма предоставления заданий оценочных средств, а именно:

· в печатной или электронной форме (для лиц с нарушениями опорно-двигательного аппарата);

· в печатной форме или электронной форме с увеличенным шрифтом и контрастностью (для лиц с нарушениями слуха, речи, зрения);

· методом чтения ассистентом задания вслух (для лиц с нарушениями зрения).

Студентам с инвалидностью увеличивается время на подготовку ответов на контрольные вопросы. Для таких студентов предусматривается доступная форма предоставления ответов на задания, а именно:

· письменно на бумаге или набором ответов на компьютере (для лиц с нарушениями слуха, речи);

· выбором ответа из возможных вариантов с использованием услуг ассистента (для лиц с нарушениями опорно-двигательного аппарата);

· устно (для лиц с нарушениями зрения, опорно-двигательного аппарата).

При необходимости для обучающихся с инвалидностью процедура оценивания результатов обучения может проводиться в несколько этапов.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ[1]
по дисциплине

Форензика

Направление подготовки	*10.05.03 "Информационная безопасность автоматизированных систем"*
Направленность (профиль) образовательной программы	*Анализ безопасности информационных систем*
Квалификация выпускника	*специалист по защите информации*
Год начала подготовки *(по учебному плану)*	*2021*
Форма обучения	*очная*
Технология обучения	*традиционная*

Курс	Семестр	Трудоемкость, з.е.
5	*A*	*3*

Вид промежуточной аттестации	Обеспечивающее подразделение
Зачет с оценкой	*Кафедра ИБАС - Информационная безопасность автоматизированных систем*

[1] В данном приложении представлены типовые оценочные средства. Полный комплект оценочных средств, включающий все варианты заданий (тестов, контрольных работ и др.), предлагаемых обучающемуся, хранится на кафедре в бумажном и электронном виде.

1 Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами образовательной программы

Таблица 1 – Компетенции и планируемые результаты обучения по дисциплине

Код и наименование компетенции	Планируемые результаты обучения по дисциплине		
	Перечень знаний	Перечень умений	Перечень навыков
Профессиональные			
Способность проводить анализ рисков информационной безопасности автоматизированной системы (ПК-5)	З1 (ПК-5-2) знать основные методы анализа сетевого трафика;	У1 (ПК-5-2) уметь анализировать сетевую инормацию;	Н1 (ПК-5-2) владеть навыком работы с Wireshark по анализу различных протоколов в сети;
Способность обеспечивать эффективное применение средств защиты информационно-технологических ресурсов автоматизированной системы и восстановление их работоспособности при возникновении нештатных ситуаций (ПК-25)	З2 (ПК-25-4) знать об основных методах исследования компьютерной информации;	У2 (ПК-25-4) уметь анализировать лог файлы операционных систем;	Н2 (ПК-25-4) владеть навыком работы с программой volatility;

Таблица 2 – Паспорт фонда оценочных средств

Контролируемые разделы (темы) дисциплины	**Код контролируемой компетенции**	**Наименование оценочного средства**	**Показатели оценки**
1. Исследование мультимедийного трафика	ПК-5	Лабораторная работа 1	Умение восстановить аудиопоток, переданный через протокол SIP, восстановить видеопоток, выполнить отчет о проделанной работе.
2. Исследование данных передаваемых через протокол TCP	ПК-5	Лабораторная работа 2	Умение в программе Wireshark восстановить информацию, переданную по протоколу FTP, SMB2, HTTP, SMPT, POP3, DNS

3. Исследование файловых систем операционных систем	ПК-25	Лабораторная работа 3	Умение проводить исследование файловых систем операционных систем
4. Исследование ОЗУ	ПК-23	Лабораторная работа 4	Умение сформировать перечень требований к персоналу при работе с конфиденциальными документами, описать технологии защиты сведений
Исследовать лог файлы веб-сервера	ПК-5 ПК-25	Расчетно-графическая работа	Показывает умения и навыки по восстановлению информации о процессах из дампа ОЗУ, извлечения информации из процесса по дампу ОЗУ

2 Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующие процесс формирования компетенций

Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, представлены в виде технологической карты дисциплины (таблица 3).

Таблица 3 – Технологическая карта

	Наименование оценочного средства	Сроки выполнения	Шкала оценивания	Критерии оценивания
3 семестр Промежуточная аттестация в форме зачета с оценкой				
1	Лабораторная работа 1	В течение семестра	10 баллов	10 баллов - студент правильно выполнил задание. Показал отличные знания, навыки и умения рамках освоенного учебного материала. 5 балла - студент выполнил задание с небольшими неточностями. Показал хорошие знания, навыки и умения рамках освоенного учебного материала. 3 балла - студент выполнил задание с существенными неточностями. Показал удовлетворительные знания, навыки и умения рамках освоенного учебного материала. 2 балла - при выполнении задания студент продемонстрировал недостаточный уровень знаний. 0 баллов – задание не выполнено.
2	Лаборатор-	В течение семестра	10 баллов	10 баллов - студент правильно выполнил зада-

	Наименование оценочного средства	Сроки выполнения	Шкала оценивания	Критерии оценивания
	ная работа 2			ние. Показал отличные знания, навыки и умения рамках освоенного учебного материала. 5 балла - студент выполнил задание с небольшими неточностями. Показал хорошие знания, навыки и умения рамках освоенного учебного материала. 3 балла - студент выполнил задание с существенными неточностями. Показал удовлетворительные знания, навыки и умения рамках освоенного учебного материала. 2 балла - при выполнении задания студент продемонстрировал недостаточный уровень знаний. 0 баллов – задание не выполнено.
2	Лабораторная работа 3	В течение семестра	10 баллов	10 баллов - студент правильно выполнил задание. Показал отличные знания, навыки и умения рамках освоенного учебного материала. 5 балла - студент выполнил задание с небольшими неточностями. Показал хорошие знания, навыки и умения рамках освоенного учебного материала. 3 балла - студент выполнил задание с существенными неточностями. Показал удовлетворительные знания, навыки и умения рамках освоенного учебного материала. 2 балла - при выполнении задания студент продемонстрировал недостаточный уровень знаний. 0 баллов – задание не выполнено.
2	Лабораторная работа 4	В течение семестра	10 баллов	10 баллов - студент правильно выполнил задание. Показал отличные знания, навыки и умения рамках освоенного учебного материала. 5 балла - студент выполнил задание с небольшими неточностями. Показал хорошие знания, навыки и умения рамках освоенного учебного материала. 3 балла - студент выполнил задание с существенными неточностями. Показал удовлетворительные знания, навыки и умения рамках освоенного учебного материала. 2 балла - при выполнении задания студент продемонстрировал недостаточный уровень знаний. 0 баллов – задание не выполнено.
5	Расчетно-графическая работа 1	В течение семестра	15 баллов	15 баллов - студент правильно выполнил задания. Показал отличные владения навыками применения полученных знаний и умений при решении профессиональных задач в рамках

	Наименование оценочного средства	Сроки выполнения	Шкала оценивания	Критерии оценивания
				усвоенного учебного материала. Ответил на все дополнительные вопросы на защите. 10 баллов - студент выполнил задание с небольшими неточностями. Показал хорошие владения навыками применения полученных знаний и умений при решении профессиональных задач в рамках усвоенного учебного материала. Ответил на большинство дополнительных вопросов на защите. 5 баллов - студент выполнил задания с существенными неточностями. Показал удовлетворительное владение навыками применения полученных знаний и умений при решении профессиональных задач в рамках усвоенного учебного материала. При ответах на дополнительные вопросы на защите было допущено много неточностей. 0 баллов - при выполнении задания студент продемонстрировал недостаточный уровень владения навыками применения полученных знаний и умений при решении профессиональных задач в рамках усвоенного учебного материала. При ответах на дополнительные вопросы на защите было допущено множество неточностей.
Текущий контроль:			55 баллов	
ИТОГО:			55 баллов	

Критерии оценки результатов обучения по дисциплине:
0 – 64 % от максимально возможной суммы баллов – «неудовлетворительно» (недостаточный уровень для промежуточной аттестации по дисциплине);
65 – 74 % от максимально возможной суммы баллов – «удовлетворительно» (пороговый (минимальный) уровень);
75 – 84 % от максимально возможной суммы баллов – «хорошо» (средний уровень);
85 – 100 % от максимально возможной суммы баллов – «отлично» (высокий (максимальный) уровень).

3 Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующие процесс формирования компетенций в ходе освоения образовательной программы

3.1 Задания для текущего контроля успеваемости

Задания для лабораторных работ

Лабораторная работа № 1 Исследование мультимедийного трафика.
В программе Wireshark выполнить:

- Восстановить аудиопоток, переданный через протокол SIP.
- Восстановить видеопоток.

Выполнить отчет о проделанной работе.

Лабораторная работа № 2 Исследование данных передаваемых через протокол TCP.

В программе Wireshark выполнить:

- Восстановить информацию, переданную по протоколу FTP.
- Восстановить информацию, переданную по протоколу SMB2.
- Восстановить информацию, переданную по протоколу HTTP.
- Восстановить информацию, переданную по протоколу SMPT.
- Восстановить информацию, переданную по протоколу POP3.
- Восстановить информацию, переданную по протоколу DNS.
- Восстановить информацию, переданную по протоколу NNTP.
- Восстановить информацию, переданную по протоколу telnet.
- Восстановить информацию, переданную по протоколу LDAP.
- Восстановить информацию, переданную по протоколу HTTPS.

Выполнить отчет о проделанной работе.

Лабораторная работа № 3 Исследование файловых систем операционных систем

- Исследовать структуру файловых систем FAT32, NTFS, Ext4.
- В ручном режиме продемонстрировать удаление и восстановление информации в файловых системах FAT32, NTFS, Ext4.
- Протестировать различное ПО для восстановления файлов (5 шт).
- Протестировать различное ПО для надежного удаления файлов (2 шт).

Выполнить отчет о проделанной работе.

Лабораторная работа № 4 Исследование ОЗУ:

- Восстановить информацию о процессах из дампа ОЗУ.
- Извлечь информацию из процесса по дампу ОЗУ.

Выполнить отчет о проделанной работе.

Задание для расчетно-графической работы

Исследовать лог файлы веб-сервера:

- браузер пользователя;
- персональный межсетевой экран на компьютере пользователя;
- антивирусная программа на компьютере пользователя;

- операционная система пользователя;
- DNS-сервер (резолвер), к которому обращался браузер пользователя перед запросом веб-страницы, а также DNS-сервера (держатели зон), к которым рекурсивно обращался этот резолвер;
- все маршрутизаторы по пути от компьютера пользователя до веб-сервера и до DNS-серверов, а также билинговые системы, на которые эти маршрутизаторы пересылают свою статистику;
- средства защиты (межсетевой экран, система обнаружения атак, антивирус), стоящие перед веб-сервером и вовлеченными DNS-серверами;
- веб-сервер;
- CGI-скрипты, запускаемые веб-сервером;
- Веб-сервера всех счетчиков и рекламных баннеров, расположенных на просматриваемой пользователем веб-странице (как правило, они поддерживаются независимыми провайдерами);
- веб-сервер, на который пользователь уходит по гиперссылке с просматриваемой страницы;
- прокси-сервер (если используется);
- оборудование СОРМ со стороны пользователя и со стороны веб-сервера.

Выполнить отчет о проделанной работе.

Лист регистрации изменений к РПД

	Номер протокола заседания кафедры, дата утверждения изменения	Количество страниц изменения	Подпись разработчика РПД