

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Узунов Федор Владимирович

Должность: Ректор

Дата подписания: 50.09.2026 09:20:02

Уникальный программный ключ:

fd935d10451b860e912264c0378f8b48452bfcb603f94388008e39877a6bcbf5

**АВТОНОМНАЯ НЕКОММЕРЧЕСКАЯ ОРГАНИЗАЦИЯ
«ОБРАЗОВАТЕЛЬНАЯ ОРГАНИЗАЦИЯ ВЫСШЕГО ОБРАЗОВАНИЯ»**

«УНИВЕРСИТЕТ ЭКОНОМИКИ И УПРАВЛЕНИЯ»

Факультет экономики, управления и юриспруденции

Кафедра «Управление и бизнес-информатика»



УТВЕРЖДАЮ

Проректор по учебно-методической работе

[Signature] / Г.П. Узунова

«02» февраля 2026 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Защита программ и данных

Направление подготовки

10.03.01 «Информационная безопасность»

Профиль

Безопасность компьютерных систем

Квалификация выпускника

Бакалавр

Для всех

форм обучения

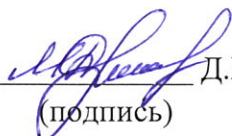
Симферополь, 2026

Рабочая программа составлена в соответствии с ФГОС ВО – бакалавриат по направлению подготовки 10.03.01 «Информационная безопасность» профиль Безопасность компьютерных систем, утвержденным приказом Министерства образования и науки Российской Федерации от 17.11.2020 г. № 1427 (зарегистрировано в Министерстве юстиции РФ 18.02.2021 №62548) с изменениями и дополнениями.

Программу составил Л. В. Яковенко, старший преподаватель

Рабочая программа дисциплины «Защита программ и данных» утверждена на заседании кафедры «Управление и бизнес-информатика».

Протокол № 6 от 29.01.2026 г.

Заведующий кафедрой  Д.В. Моторина
(подпись)

АННОТАЦИЯ	
Индекс дисциплины по учебному плану	Наименование дисциплины
Б1.О.31	Защита программ и данных
Цель изучения дисциплины	формирование у студентов профессиональных компетенций, необходимых для разработки, анализа безопасности и практического внедрения комплексных мер по защите программного обеспечения (ПО) и обрабатываемой им информации от несанкционированного доступа, исследования, модификации и тиражирования.
Место дисциплины в структуре ОПОП	Дисциплина относится к обязательной части блока 1. «Дисциплины (модули)» программы бакалавриата
Компетенции, формируемые в результате освоения дисциплины	ОПК-3
Содержание дисциплины	Тема 1. Методы защиты программ от обратной разработки (Reverse Engineering) и анализа. Тема 2. Технологии контроля целостности и аутентичности программного обеспечения. Тема 3. Безопасная разработка программного обеспечения (DevSecOps) и анализ уязвимостей. Тема 4. Методы защиты обрабатываемых данных во время исполнения (Runtime Application Self-Protection — RASP). Тема 5. Криптографические методы защиты хранимых данных и управление ключами. Тема 6. Защита от недекларированных возможностей (программных закладок) и анализ вредоносного ПО.
Общая трудоемкость дисциплины	Общая трудоемкость дисциплины составляет 6 зачетные единицы (216часов)
Форма промежуточной аттестации	Экзамен/ курсовая работа

Содержание

1. Цель и перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения программы бакалавриата	5
2. Место дисциплины в структуре ОПОП бакалавриата	5
3. Объем дисциплины в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем (по видам занятий) и на самостоятельную работу обучающихся	5
4. Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий	6
5. Контроль качества освоения дисциплины	11
6. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины	11
7. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины	12
8. Методические указания для обучающихся по освоению дисциплины	13
9. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем (при необходимости)	13
10. Описание материально-технического обеспечения, необходимого для осуществления образовательного процесса по дисциплине	14
	15

1. Цель и перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения программы бакалавриата

Цель изучения дисциплины «Защита программ и данных» – формирование у студентов профессиональных компетенций, необходимых для разработки, анализа безопасности и практического внедрения комплексных мер по защите программного обеспечения (ПО) и обрабатываемой им информации от несанкционированного доступа, исследования, модификации и тиражирования.

В результате освоения ОПОП бакалавриата обучающийся должен овладеть следующими результатами обучения по дисциплине:

Коды компетенции	Результаты освоения ОПОП	Перечень планируемых результатов обучения по дисциплине
ОПК-3	Способен использовать необходимые математические методы для решения задач профессиональной деятельности	ОПК-3.1. Применяет соответствующий математический аппарат для решения задач профессиональной деятельности. ОПК-3.2. Анализирует естественнонаучную сущность проблем, возникающих в ходе профессиональной деятельности

2. Место дисциплины в структуре ОПОП бакалавриата

Дисциплина Б1.О.31 «Защита программ и данных» относится к обязательной части блока 1. «Дисциплины (модули)» учебного плана ОПОП бакалавриата по направлению подготовки 10.03.02 «Информационная безопасность». Дисциплина «Информационные технологии в профессиональной деятельности» изучается обучающимися очной формы обучения в 6 семестре, очно-заочной формы обучения – в 6 семестре.

Знания, умения и навыки, полученные при изучении дисциплины «Защита программ и данных», будут необходимы для углубленного и осмысленного восприятия дисциплин: Операционные системы и др.

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

Общая трудоемкость (объем) дисциплины составляет 6 зачетных единицы (з.е.), 216 академических часа.

3.1. Объём дисциплины по видам учебных занятий (в часах)

Для очной формы обучения

Общая трудоёмкость дисциплины составляет 6 зачётных единицы 216 часа

Объём дисциплины	Всего часов
Общая трудоемкость дисциплины	216
Контактная работа	52
Аудиторная работа (всего):	52

Лекции	18
Семинары, практические занятия	34
Самостоятельная работа обучающихся (всего)	128
Курсовая работа	+
Экзамен	36

Для очно-заочной формы обучения

Общая трудоёмкость дисциплины составляет 6 зачётных единиц 216 часа

Объём дисциплины	Всего часов
Общая трудоёмкость дисциплины	216
Контактная работа	40
Аудиторная работа (всего):	40
Лекции	20
Семинары, практические занятия	20
Самостоятельная работа обучающихся (всего)	140
Курсовая работа	+
Экзамен	36

4. Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

4.1. Разделы дисциплины и трудоёмкость по видам учебных занятий (в академических часах)

№ темы	Наименование темы	Всего		Количество часов					
		ОФО	ОЗФО	Контактная работа				Внеаудит. работа	
				Лекции		Практические		Самост. работа	
				ОФО	ОЗФО	ОФО	ОЗФО	ОФО	ОЗФО
1.	Тема 1. Методы защиты программ от обратной разработки (Reverse Engineering) и анализа.	34	34	2	2	4	2	28	30
2.	Тема 2. Технологии контроля целостности и аутентичности программного обеспечения.	34	34	4	2	4	2	28	30
3.	Тема 3. Безопасная разработка программного обеспечения (DevSecOps) и анализ уязвимостей.	34	34	4	2	4	2	28	30
4.	Тема 4. Методы защиты обрабатываемых данных во время исполнения (Runtime Application Self-Protection — RASP).	34	32	4	1	4	1	28	30

5.	Тема 5. Криптографические методы защиты хранимых данных и управление ключами.	32	32	4	1	4	1	28	30
6.	Тема 6. Защита от недекларированных возможностей (программных закладок) и анализ вредоносного ПО.	36	30	6	2	6	2	24	26
	Всего по дисциплине	180	180	18	20	34	20	128	140
	Контроль	36	36						
	Итого	216	216						

4.2. Содержание дисциплины, структурированное по темам (разделам)

Тема 1. Методы защиты программ от обратной разработки (Reverse Engineering) и анализа Раскрывается комплекс мер по защите интеллектуальной собственности и логики работы программного обеспечения. Изучаются техники статической обфускации кода (запутывание потока управления, переименование идентификаторов, изменение типов данных), затрудняющие чтение дизассемблированного листинга. Рассматриваются методы динамической защиты: противодействие отладчикам (*anti-debugging* через проверку флагов состояния процессора или системных структур), обнаружение виртуальных сред (*anti-VM*) и инструментов дизассемблирования. Практическая часть включает изучение принципов виртуализации кода (преобразование инструкций x86 в байт-код собственной виртуальной машины) и использования протекторов для шифрования секций исполняемых файлов.

Тема 2. Технологии контроля целостности и аутентичности программного обеспечения Рассматриваются механизмы подтверждения того, что программа не была модифицирована злоумышленником после выпуска разработчиком. Изучается применение криптографических хеш-функций (*ГОСТ Р 34.11-2012*, *SHA-256*) для создания контрольных сумм модулей. Детально разбирается технология цифровой подписи кода (*Code Signing*): использование инфраструктуры открытых ключей (*PKI*), сертификатов разработчиков и временных меток (*timestamping*) для гарантии авторства и актуальности подписи. Анализируются встроенные средства защиты операционных систем: механизм проверки подписей драйверов в Windows (*KMCS*), система Gatekeeper и нотации App Sandbox в macOS, а также проверка хешей бинарных файлов в Linux через *IMA/EVM*.

Тема 3. Безопасная разработка программного обеспечения (DevSecOps) и анализ уязвимостей Раскрывается методология интеграции проверок безопасности на всех этапах жизненного цикла ПО (*SDLC*). Изучаются инструменты статического анализа (*SAST*) для поиска дефектов исходного кода без его запуска (например, SQL-инъекции, небезопасное использование функций работы с памятью). Рассматривается динамический анализ (*DAST*) и фаззинг (*Fuzzing*) — подача некорректных входных данных для вызова отказов системы и выявления ошибок переполнения буфера (*Buffer Overflow*, *Heap Spray*). Особое внимание уделяется модели угроз (*STRIDE*), практике безопасной кодировки (*Secure Coding Standards*) и устранению уязвимостей из перечня *OWASP Top 10*.

Тема 4. Методы защиты обрабатываемых данных во время исполнения (Runtime Application Self-Protection — RASP) Анализируются технологии активной самозащиты приложений непосредственно в оперативной памяти. Изучаются принципы внедрения защитных агентов, перехватывающих критические вызовы API (функции выделения памяти, обращения к сети, файловой системе). Рассматриваются методы защиты от эксплуатации уязвимостей «на лету»: эмуляция выполнения (*Just-In-Time*

Simulation) подозрительных фрагментов кода, блокировка нетипичных последовательностей системных вызовов (*Syscall Filtering*) и маскировка адресного пространства (*ASLR* — *Address Space Layout Randomization*). Разбираются механизмы предотвращения инъекций кода и несанкционированного чтения конфиденциальных строк в памяти процесса.

Тема 5. Криптографические методы защиты хранимых данных и управление ключами Детально рассматривается защита информации, находящейся в покое (*Data at Rest*). Изучаются режимы полнодискового шифрования (*Full Disk Encryption* — *FDE*, например, BitLocker, LUKS) и прозрачного шифрования баз данных (*Transparent Data Encryption* — *TDE*). Раскрываются принципы иерархии хранения ключей: использование мастер-ключей, хранение которых вынесено в специализированные аппаратные модули безопасности (*HSM*) или облачные сервисы управления ключами (*KMS*). Анализируются алгоритмы шифрования колонок в БД, токенизация данных (замена чувствительной информации несекретными суррогатами) и проблема безопасного удаления остаточной информации с носителей.

Тема 6. Защита от недекларированных возможностей (программных закладок) и анализ вредоносного ПО Тема посвящена выявлению умышленно внедренных скрытых функций («логических бомб», бэкдоров) и анализу враждебного кода. Изучаются методики аудита исходного кода на наличие избыточной функциональности и скрытых условий срабатывания. Рассматриваются поведенческие признаки вредоносного ПО при анализе в песочницах (*sandboxing*): попытки закрепления в системе (*persistence* через реестр или планировщик задач), сетевая активность к управляющим центрам (*C2*), внедрение в чужие процессы (*process injection*). Практическая часть включает реверс-инжиниринг малвари для извлечения индикаторов компрометации (*IoC*) и разработку правил детектирования (сигнатур и правил *YARA*) для средств антивирусной защиты и систем обнаружения вторжений.

4.3. Содержание практических занятий (очная форма обучения)

Тема 1. Методы защиты программ от обратной разработки (Reverse Engineering) и анализа Раскрывается комплекс мер по защите интеллектуальной собственности и логики работы программного обеспечения. Изучаются техники статической обфускации кода (запутывание потока управления, переименование идентификаторов, изменение типов данных), затрудняющие чтение дизассемблированного листинга.
Тема 2. Технологии контроля целостности и аутентичности программного обеспечения Рассматриваются механизмы подтверждения того, что программа не была модифицирована злоумышленником после выпуска разработчиком. Изучается применение криптографических хеш-функций (<i>ГОСТ Р 34.11-2012</i>, <i>SHA-256</i>) для создания контрольных сумм модулей.
Тема 3. Безопасная разработка программного обеспечения (DevSecOps) и анализ уязвимостей Раскрывается методология интеграции проверок безопасности на всех этапах жизненного цикла ПО (<i>SDLC</i>). Изучаются инструменты статического анализа (<i>SAST</i>) для поиска дефектов исходного кода без его запуска (например, SQL-инъекции, небезопасное использование функций работы с памятью).
Тема 4. Методы защиты обрабатываемых данных во время исполнения (Runtime Application Self-Protection — RASP) Анализируются технологии активной самозащиты приложений непосредственно в оперативной памяти. Изучаются принципы внедрения защитных агентов, перехватывающих критические вызовы API (функции выделения

памяти, обращения к сети, файловой системе).

Тема 5. Криптографические методы защиты хранимых данных и управление ключами Детально рассматривается защита информации, находящейся в покое (*Data at Rest*). Изучаются режимы полнодискового шифрования (*Full Disk Encryption — FDE*, например, BitLocker, LUKS) и прозрачного шифрования баз данных (*Transparent Data Encryption — TDE*).

Тема 6. Защита от недеklarированных возможностей (программных закладок) и анализ вредоносного ПО Тема посвящена выявлению умышленно внедренных скрытых функций («логических бомб», бэкдоров) и анализу враждебного кода. Изучаются методики аудита исходного кода на наличие избыточной функциональности и скрытых условий срабатывания. Рассматриваются поведенческие признаки вредоносного ПО при анализе в песочницах (*sandboxing*): попытки закрепления в системе (*persistence* через реестр или планировщик задач), сетевая активность к управляющим центрам (*C2*), внедрение в чужие процессы (*process injection*).

4.4. Содержание самостоятельной работы

Тема 1. Методы защиты программ от обратной разработки (Reverse Engineering) и анализа Методы динамической защиты: противодействие отладчикам (*anti-debugging* через проверку флагов состояния процессора или системных структур), обнаружение виртуальных сред (*anti-VM*) и инструментов дизассемблирования. Практическая часть включает изучение принципов виртуализации кода (преобразование инструкций x86 в байт-код собственной виртуальной машины) и использования протекторов для шифрования секций исполняемых файлов.

Тема 2. Технологии контроля целостности и аутентичности программного обеспечения технология цифровой подписи кода (*Code Signing*): использование инфраструктуры открытых ключей (*PKI*), сертификатов разработчиков и временных меток (*timestamping*) для гарантии авторства и актуальности подписи. Анализируются встроенные средства защиты операционных систем: механизм проверки подписей драйверов в Windows (*KMCS*), система Gatekeeper и нотации App Sandbox в macOS, а также проверка хешей бинарных файлов в Linux через *IMA/EVM*.

Тема 3. Безопасная разработка программного обеспечения (DevSecOps) и анализ уязвимостей Динамический анализ (*DAST*) и фаззинг (*Fuzzing*) — подача некорректных входных данных для вызова отказов системы и выявления ошибок переполнения буфера (*Buffer Overflow*, *Heap Spray*). Особое внимание уделяется модели угроз (*STRIDE*), практике безопасной кодировки (*Secure Coding Standards*) и устранению уязвимостей из перечня *OWASP Top 10*.

Тема 4. Методы защиты обрабатываемых данных во время исполнения (Runtime Application Self-Protection — RASP) Методы защиты от эксплуатации уязвимостей «на лету»: эмуляция выполнения (*Just-In-Time Simulation*) подозрительных фрагментов кода, блокировка нетипичных последовательностей системных вызовов (*Syscall Filtering*) и маскировка адресного пространства (*ASLR — Address Space Layout Randomization*). Разбираются механизмы предотвращения инъекций кода и несанкционированного чтения конфиденциальных строк в памяти процесса.

Тема 5. Криптографические методы защиты хранимых данных и управление ключами Алгоритмы шифрования колонок в БД, токенизация данных (замена чувствительной информации несекретными суррогатами) и проблема безопасного удаления остаточной информации с носителей.

Тема 6. Защита от недеklarированных возможностей (программных закладок) и

анализ вредоносного ПО Реверс-инжиниринг малвари для извлечения индикаторов компрометации (IoC) и разработку правил детектирования (сигнатур и правил YARA) для средств антивирусной защиты и систем обнаружения вторжений.

5. Контроль качества освоения дисциплины

Текущий контроль и промежуточная аттестация осуществляются в соответствии с «Положением о текущем контроле успеваемости и промежуточной аттестации обучающихся в Автономной некоммерческой организации «Образовательная организация высшего образования» «Университет экономики и управления».

Вид промежуточной аттестации – экзамен. Форма проведения промежуточной аттестации – письменный экзамен.

Фонд оценочных средств по дисциплине приведен в приложении к РПД.

6. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

а) основная литература:

1. Батуро, А. Н. Конфигурирование ИТ-инфраструктуры. Прикладное программное обеспечение под ОС Astra Linux Special Edition : учебное пособие для СПО / А. Н. Батуро, Г. М. Бойко. — Москва : Ай Пи Ар Медиа, 2026. — 344 с. — ISBN 978-5-4497-5384-7. — Текст : электронный // Образовательная платформа IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/books/161012/details> (дата обращения: 27.01.2026). — Режим доступа: для авторизир. пользователей

2. Мельников, А. В. Основы информационной безопасности : учебное пособие / А. В. Мельников, С. В. Зарубин. — Москва : Российский государственный университет правосудия имени В.М. Лебедева, 2025. — 220 с. — ISBN 978-5-00209-188-1. — Текст : электронный // Образовательная платформа IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/books/152309/details> (дата обращения: 27.01.2026). — Режим доступа: для авторизир. пользователей

б) дополнительная литература:

3. Баженова, И. Ю. Введение в программирование : учебное пособие / И. Ю. Баженова, В. А. Сухомлин. — 5-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2026. — 326 с. — ISBN 978-5-4497-3298-9. — Текст : электронный // Образовательная платформа IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/books/162502/details> (дата обращения: 27.01.2026). — Режим доступа: для авторизир. пользователей

4. Основы информационных технологий : учебное пособие / С. В. Назаров, С. Н. Белоусова, И. А. Бессонова [и др.]. — 3-е изд. — Москва, Саратов : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020. — 530 с. — ISBN 978-5-4497-0339-2. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/89454.html> (дата обращения: 16.01.2026). — Режим доступа: для авторизир. пользователей.

7. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

1. Информационно-правовой портал «Гарант»: официальный сайт. — URL:

<http://www.garant.ru> – Текст: электронный.

2. Цифровой образовательный ресурс «IPRsmart»: официальный сайт. – URL: <http://www.iprbookshop.ru/> – Текст: электронный.

3. Научная электронная библиотека «КиберЛенинка»: официальный сайт. – URL: <https://cyberleninka.ru/> – Текст: электронный.

4. Российский интернет-портал и аналитическое агентство TAdviser: официальный сайт. – URL: <https://www.tadviser.ru/> – Текст: электронный.

8. Методические указания для обучающихся по освоению дисциплины

При проведении лекций, семинарских (практических) занятий, самостоятельной работе обучающихся применяются интерактивные формы проведения занятий с целью погружения обучающихся в реальную атмосферу профессионального сотрудничества по разрешению проблем, оптимальной выработки навыков и качеств будущего специалиста. Интерактивные формы проведения занятий предполагают обучение в сотрудничестве. Все участники образовательного процесса (преподаватель и обучающиеся) взаимодействуют друг с другом, обмениваются информацией, совместно решают проблемы, моделируют ситуацию.

В учебном процессе используются интерактивные формы занятий:

- творческое задание. Выполнение творческих заданий требует от обучающегося воспроизведение полученной ранее информации в форме, определяемой преподавателем, и требующей творческого подхода;

- групповое обсуждение. Групповое обсуждение кого-либо вопроса направлено на достижение лучшего взаимопонимания и способствует лучшему усвоению изучаемого материала.

В ходе освоения дисциплины при проведении контактных занятий используются следующие формы обучения, способствующие формированию компетенций: лекции-дискуссии; кейс-метод; решение задач; ситуационный анализ; обсуждение рефератов и докладов; разработка групповых проектов; встречи с представителями государственных и общественных организаций.

9. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем (при необходимости)

В процессе лекционных и практических занятий используется следующее программное обеспечение:

- *программы, обеспечивающие доступ в сеть «Интернет» (например, «Microsoft Edge», «Google Chrome»);

- *программы, демонстрации видео материалов (например, проигрыватель «Windows Media Player»);

- *текстовые редакторы и процессоры (например, «Microsoft Office Word»);

- *табличные процессоры (например, «Microsoft Office Excel»);

- *системы управления базами данных (например, «Microsoft Office Access»);

- *программы для демонстрации и создания презентаций (например, «Microsoft PowerPoint»);

*проблемно-ориентированные пакеты прикладных программ по отраслям и сферам деятельности (например, «1С: Управление нашей фирмой», «Loginom Community Edition»).

10. Описание материально-технического обеспечения, необходимого для осуществления образовательного процесса по дисциплине

Учебная аудитория

Оборудование учебной аудитории:

рабочее место преподавателя; посадочные места по количеству обучающихся;

доска классная;

стенды информационные;

Учебно-наглядные пособия: компьютеры с лицензионным программным обеспечением и возможностью подключения к информационно-телекоммуникационной сети Интернет; мультимедийная установка.

Помещение для самостоятельной работы обучающихся

оснащено компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду Университета.