МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное бюджетное образовательное учреждение

высшего образования

«ИРКУТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

ФГБОУ ВО «ИГУ»

Кафедра алгебраических и информационных систем

«Утверждаю»
Директор ИМЭИ ИГУ
_____Фалалеев М.В.
«30» марта 2016 г.

Рабочая программа дисциплины (модуля)

Б1.В.ОД.19 КРИПТОГРАФИЯ

Направление подготовки 09.03.03 Прикладная информатика

Тип образовательной программы *академический бакалавриат*

Направленность (профиль) Информационная сфера

Квалификация (степень) выпускника – БАКАЛАВР

Форма обучения *заочная*

Иркутск 2016

Согласовано с УМК ИМЭИ

Протокол № 1

от 11 февраля 2016 г.

Председатель _____ Рожина Л.В.

Рекомендовано кафедрой

Протокол № 7

от 2 февраля 2016 г.

Ио зав. кафедрой

_____ Манцивода А.В.

Содержание

1. Цели и задачи дисциплины 4
2. Место дисциплины в структуре ОПОП 4
3. Требования к результатам освоения дисциплины 4
4. Объем дисциплины и виды учебной работы 5
5. Содержание дисциплины 5
5.1. Содержание разделов и тем дисциплины 5
5.2 Разделы дисциплины и междисциплинарные связи с обеспечиваемыми (последующими) дисциплинами 7
5.3. Разделы и темы дисциплин и виды занятий 7
6. Перечень семинарских, практических занятий и лабораторных работ, план самостоятельной работы студентов, методические указания по организации самостоятельной работы студентов 8
7. Примерная тематика курсовых работ (проектов) 10
8. Учебно-методическое и информационное обеспечение дисциплины 10
9. Материально-техническое обеспечение дисциплины 11
10. Образовательные технологии 11
11. Оценочные средства (ОС) 11

1. Цели и задачи дисциплины

Целями освоения дисциплины «Криптография» являются ознакомление студентов с основами криптографии и основами обеспечения защиты информации, формирование практических умений и навыков, необходимых для приобретения квалификации бакалавра прикладной информатики, формирование ключевых профильных компетенций. Задачи освоения дисциплины: дать специальные знания по дисциплине, достичь достаточного уровня знаний по криптографическим методам обеспечения информационной безопасности и сформировать у студентов практические навыки работы со средствами обеспечения информационной безопасности.

2. Место дисциплины в структуре ОПОП

Дисциплина «Криптография» относится к дисциплинам вариативной части профессионального цикла по направлению подготовки «Прикладная информатика» (бакалавриат). Знания, умения и навыки, формируемые в процессе освоения дисциплины «Криптография» используются при прохождении производственной практики и выполнении выпускной квалификационной работы. Для усвоения дисциплины «Криптография» необходимы знания и навыки, сформированные дисциплинами «Программирование» и «Языки программирования».

3. Требования к результатам освоения дисциплины

Процесс изучения дисциплины направлен на формирование следующих компетенций:

ОПК-3 – способностью использовать нормативно-правовые документы, международные и отечественные стандарты в области информационных систем и технологий;

ОПК-4 – способностью решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности;

ПК-23 – способность применять системный подход и математические методы в формализации решения прикладных задач.

В результате изучения дисциплины студент должен:

Знать: международные и национальные стандарты шифрования; основные виды угроз информационной безопасности и способы противодействия этим угрозам; алгоритмы обеспечения целостности информации; инфраструктуру открытых ключей.

Уметь: использовать криптографические преобразования для решения задач профессиональной деятельности; анализировать криптографические алгоритмы взаимодействия на наличие уязвимостей.

Владеть: навыками работы с алгоритмами и методами шифрования, работы с криптопровайдерами, использования криптографических примитивов в языках программирования.

4. Объем дисциплины и виды учебной работы

Вид учебной работы	Всего часов / зачетных единиц	Семестры
		Сессия С
Аудиторные занятия (всего)	8	8
В том числе:	–	–
Лекции	4	4
Практические занятия (ПЗ)	0	0
Семинары (С)	0	0
Лабораторные работы (ЛР)	4	4
Контроль самостоятельной работы (КСР)	9	9
Самостоятельная работа (всего)	91	91
В том числе:	–	–
Домашние задания	91	91
Вид промежуточной аттестации (зачет, экзамен)	экзамен	экзамен
Общая трудоемкость часы	108	108
зачетные единицы	3	3

5. Содержание дисциплины

5.1. Содержание разделов и тем дисциплины

РАЗДЕЛ 1. ОСНОВЫ КРИПТОГРАФИИ

ТЕМА 1.1. Базовые понятия криптографии

Общие принципы и модели. Защита от несанкционированного доступа. Понятие ключа. Шифрование и кодирование. Криптосистемы. История криптографии. Шифр простой замены. Шифр Цезаря. Шифр вертикальной перестановки. Гаммирование. Основные способы криптоанализа простых шифров. Идеальный шифр.

ТЕМА 1.2. Симметричные криптосистемы

Шифр Вернама. Лемма о теоретически стойком шифре. Потоковые шифры. Атаки на потоковые шифры. Двухразовый блокнот. Пример уязвимости в протоколе. 802.11b WEP. Возможные улучшения протокола.

ТЕМА 1.3. Поточные шифры

Примеры потоковых шифров. RC4 и его уязвимости. CSS и его уязвимости. eStream. Безопасность потоковых генераторов. Статистический тест для генератора. Криптографическое определение псевдослучайного генератора. Семантически безопасный потоковый шифр. Примеры таких шифров.

ТЕМА 1.4. Блочное шифрование

Принцип итерирования. Понятие псевдослучайной функции и псевдослучайной перестановки. Шифр DES. Описание структуры. Теорема о трехраундовой сети Фейстеля. Шифр AES. Описание структуры и принципы работы. Переборные атаки на блочные шифры. Проблема DES. Тройной DES. Атака методом встречи посередине. Переборные атаки на блочные шифры. Проблема DES. Атаки на блочные шифры через утечки по побочным каналам. Атаки на основе отказов. Линейный криптонанализ DES (основные идеи). Шифр ГОСТ 28147-89 и его инфраструктура.

ТЕМА 1.5. Режимы шифрования

Построение блочных шифров на основе псевдослучайных генераторов. Шифрование нескольких сообщений на одном ключе. Атака на основе выборочного открытого текста. Пример шифра нестойкого к атаке на основе выборочного открытого текста. Случайное шифрование. Одноразовое шифрование. Режим CBC. Шифрование с предсказанным значением вектора инициализации. CBC для одноразового шифрования. Набивка блоков. Режим CTR. Сравнение режимов CBC и CTR. Режим OFB.

РАЗДЕЛ 2. ОБЕСПЕЧЕНИЕ ЦЕЛОСТНОСТИ СООБЩЕНИЙ

ТЕМА 2.1. Коды аутентификации сообщений

Коды аутентификации сообщений. Построение кодов аутентификации больших сообщений. Конструкции CBC-MAC и NMAC. Их эффективность. Схема атаки с использованием коллизий. Конструкция HMAC. Построение кодов аутентификации больших сообщений. Механизмы набивки сообщений. Конструкции CMAC, PMAC, одноразовый MAC.

ТЕМА 2.2. Хеш-функции

Построение кодов аутентификации сообщений из стойких к коллизиям функций. Атака на основе парадокса дней рождений. Конструкция Меркла-Дамгарда для построения хеш-функций. Построение безопасных сжимающих функций. Хеш-функции на основе блочных шифров.

ТЕМА 2.3. Стандарты аутентичного шифрования

Необходимость применения аутентичного шифрования. Понятие целостности шифр-текста. Атака на основе выборочного шифр-текста. Аутентичное шифрования на примере протокола TLS. Построение схем аутентичного шифрования из блочных шифров и кодов аутентификации сообщений. Стандарты аутентичного шифрования. Атаки на схемы аутентичного шифрования. Атака на основе анализа набивки сообщения.

РАЗДЕЛ 3. ШИФРОВАНИЕ СООБЩЕНИЙ НА ОТКРЫТОМ КЛЮЧЕ

ТЕМА 3.1. Асимметричное шифрование

Распределение ключей шифрования. Процедуры обмена ключами. Общее введение в теорию асимметричных криптосистем. Эффективные алгоритмы возведения в степень. Алгоритмы факторизации больших чисел. Понятие функции с лазейкой.

ТЕМА 3.2. Преобразование RSA

Устройство RSA. Эффективность реализации. Криптостойкость RSA. Варианты стандартов PKCS1. Атаки на криптосистему RSA. Атака на основе общего RSA модуля. Атака на основе малого значения закрытого ключа.

ТЕМА 3.3. Метод ключевого обмена Диффи-Хелмана

Протокол ключевого обмена для нескольких участников. Некоторые модификации метода. Односторонняя генерация ключа. Задача дискретного логарифмирования. Алгоритмы вычисления дискретного логарифма. Алгоритм шагов.

ТЕМА 3.4. Преобразование Эль-Гамаля

Вычисление и проверка подписи. Шифрование и дешифрование. Эффективность реализации. Особенности использования сеансового ключа. Практическое применение преобразования Эль-Гамаля для задачи доказательства знания.

5.2 Разделы дисциплины и междисциплинарные связи с обеспечиваемыми (последующими) дисциплинами

№ п/п	Наименование обеспечиваемых (последующих) дисциплин	№№ разделов и тем данной дисциплины, необходимых для изучения обеспечиваемых (последующих) дисциплин								

5.3. Разделы и темы дисциплин и виды занятий

№ п/п	Наименование раздела	Наименование темы	Виды занятий в часах					
			Лекц.	Практ. зан.	Семин	Лаб. зан.	СРС	Всего
1	2	3	4	5	6	7	8	9

1	Основы криптографии	Базовые понятия криптографии					4	4
2		Симметричные криптосистемы	1			1	8	10
3		Поточные шифры					6	6
4		Блочное шифрование	1			1	8	10
5		Режимы шифрования					6	6
6	Обеспечение целостности сообщений	Коды аутентификации сообщений					3	3
7		Хеш-функции					6	6
8		Стандарты аутентичного шифрования	1			1	10	12
9	Шифрование сообщений на открытом ключе	Асимметричное шифрование	1			1	8	10
10		Преобразование RSA					16	16
11		Метод ключевого обмена Диффи-Хелмана					8	8
12		Преобразование Эль-Гамаля					8	8

6. Перечень семинарских, практических занятий и лабораторных работ, план самостоятельной работы студентов, методические указания по организации самостоятельной работы студентов

№ п/п	№ раздела и темы дисциплины	Наименование семинаров, практических и лабораторных работ	Трудоемкость (часы)	Оценочные средства	Формируемые компетенции
1	**2**	**3**	**4**	**5**	**6**
1	1.1	Базовые понятия криптографии		Проверка домашней работы	ОПК-3, ОПК-4, ПК-23
2	1.2	Симметричные криптосистемы	1	Проверка домашней работы	ОПК-4
3	1.3	Поточные шифры		Проверка домашней работы	ОПК-4

4	1.4	Блочное шифрование	1	Проверка домашней работы, контрольная работа	ОПК-3, ОПК-4
5	1.5	Режимы шифрования		Проверка домашней работы	ОПК-3, ОПК-4
6	2.1	Коды аутентификации сообщений		Проверка домашней работы	ОПК-4
7	2.2	Хеш-функции		Проверка домашней работы	ОПК-3, ОПК-4, ПК-23
8	2.3	Стандарты аутентичного шифрования	1	Проверка домашней работы, контрольная работа	ОПК-3, ОПК-4
9	3.1	Асимметричное шифрование	1	Проверка домашней работы	ОПК-1, ОПК-4
10	3.2	Преобразование RSA		Проверка домашней работы	ОПК-3, ОПК-4, ПК23
11	3.3	Метод ключевого обмена Диффи-Хелмана		Проверка домашней работы	ОПК-4
12	3.4	Преобразование Эль-Гамаля		Проверка домашней работы	ОПК-4

6.1. План самостоятельной работы студентов

№ нед	№ раздела	Вид самостоятельной работы	Задание	Рекомендуемая литература	Количество часов
1	2	3	4	5	6
1-7	1	Работа со справочной литературой, первоисточниками, конспектом, выполнение домашнего задания	Повторение и углубленное изучение учебного материала лекции, домашнее задание с использованием конспекта лекций, литературы, Интернет-ресурсов	Источники 1-4 из основной и 5-9 из дополнительной литературы; Самостоятельный поиск литературы на образовательных ресурсах, доступные по логину и паролю, предоставляемым Научной библиотекой	32
8-11	2				19
12-15	3				40

				ИГУ	

6.2. Методические указания по организации самостоятельной работы студентов

Самостоятельная работа студентов – индивидуальная учебная деятельность, осуществляемая без непосредственного руководства преподавателя (научного руководителя (консультанта), в ходе которой студент активно воспринимает, осмысливает полученную информацию, решает теоретические и практические задачи. В процессе проведения самостоятельной работы формируются компетенции ОПК-3, ОПК-4, ПК-23.

Внеаудиторная самостоятельная работа студентов заключается в подготовке к выполнению лабораторных работ. Оценка внеаудиторной самостоятельной работы студентов проводится посредством проверки домашних работа. Самостоятельная работа проводится в соответствии с предложенным в разделе 6.1 графиком.

Для подготовки к выполнению лабораторным заданиям рекомендуется пользоваться основной и дополнительной учебно-методической литературой, представленной в разделе 8, а также самостоятельно проводить поиск литературы на образовательных ресурсах, доступные по логину и паролю, предоставляемым Научной библиотекой ИГУ.

7. Примерная тематика курсовых работ (проектов)

Учебным планом не предусмотрено.

8. Учебно-методическое и информационное обеспечение дисциплины

а) основная литература

1. Введение в криптографию / ред. В. В. Ященко. – М.: Изд-во МЦНМО, 2012. – 347 с. – ISBN: 978-5-4439-0026-1 (26 экз.)
2. Рябец Л.В. Задачник-практикум по криптографии: учеб. пособие / Л.В. Рябец. – Иркутск : Изд-во Вост-Сиб. гос. акад. образ., 2013. – 76 с. – ISBN: 978-5-85827-864-1 (30 экз.)
3. Шаньгин В.Ф. Защита компьютерной информации. Эффективные методы и средства / В.Ф. Шаньгин. – М.: ДМК-Пресс. – 2010. – 542 с. – ISBN: 978-5-94074-518-1 (25 экз.)
4. Виноградов И.М. Основы теории чисел : учеб. пособие / И.М. Виноградов. – М.: Лань, 2009. – 176 с. ISBN 978-5-8114-0535-0. Режим доступа: ЭБС «Лань». – Неогранич. доступ.

б) дополнительная литература

5. Герман О.Н. Теоретико-числовые методы в криптографии: учебник для студ. учреждений высш. проф. образования / О.Н. Герман. – М.: Академия. – 2012. – 257 с. – ISBN: 978-5-7695-6786-5. Режим доступа: ЭЧЗ «Библиотех». – Неогранич. доступ.

6. Смарт Н. Криптография: учебное пособие / Н. Смарт – М.: Техносфера, 2005. – 525 с. – ISBN 5-94836-043-1 (5 экз.)
7. Нечаев, В.И. Элементы криптографии: основы теории защиты информации: Учеб.пособие для ун-тов и пед.вузов / В.И. Нечаев. – М.: Высш. шк., 1999. – 109 с. – ISBN 5060036448 (17 экз.).
8. Чмора А. Л. Современная прикладная криптография: учеб.пособие / А.Л. Чмора – М.: Гелиос АРВ, 2001. – 244 с. – ISBN 5854380374 (51экз.)
9. Глухов М.М. Введение в теоретико-числовые методы криптографии / М.М. Глухов, И.А. Круглов, А.Б. Пичкур, А.В. Черемушкин. – СПб.: Лань. – 2011. – 400 с. – ISBN: 978-5-8114-1116-0. Режим доступа: ЭБС «Лань». – Неогранич. доступ.

в) программное обеспечение

10. Комплект разработчика приложений Java Platform (JDK) 8, Standard Edition (распространяется бесплатно);
11. Интегрированная среда разработки NetBeans IDE 8 (распространяется бесплатно, LGPLv2.1, GPLv2 with Classpatch exception);
12. LibreOffice Impress (распространяется бесплатно, GNU LGPL v3+, MPL 2.0);

г) базы данных, информационно-справочные и поисковые системы

13. Java Cryptography Architecture (JCA) Reference Guide. URL: https://docs.oracle.com/javase/8/docs/technotes/guides/security/crypto/CryptoSpec.html

9. Материально-техническое обеспечение дисциплины

- Учебная аудитория для проведения занятий лекционного типа, оборудованная специализированной (учебной) мебелью (доска), оборудованием для презентации учебного материала: проектор, ноутбук с установленным программным обеспечением из пункта 8.в;
- Компьютерный класс с доской, проектором и выходом в сеть Интернет;
- Электронная поддержка дисциплины в ИОС «DOMIC» (http://domic.isu.ru/): слайды к лекциям, лабораторные работы, задания для самостоятельного выполнения, тестовые задания.
- Материалы для проведения контрольных работ.

10. Образовательные технологии

Теоретические занятия дисциплины представлены в виде лекций. Чтение лекций по данной дисциплине проводится как в классической форме, так и с использованием мультимедийных презентаций. Слайд-конспект курса лекций предназначен для более глубокого усвоения материала при изучении разделов, связанных с теоретической частью курса. Чтение лекций по темам также предполагает разбор конкретных ситуаций в качестве примеров, подкрепляющих теоретический материал. Студентам предоставляется возможность копирования презентаций для самоподготовки и подготовки к выполнению лабораторных и домашних заданий.

11. Оценочные средства (ОС)

Оценочные средства используются для оценки достижений студентов в процессе изучения дисциплины; управления процессом приобретения студентами необходимых знаний, умений, навыков и формирования компетенций; оценки способностей студента к

творческой деятельности, обеспечивающей решения новых задач; обеспечения соответствия результатов обучения задачам будущей профессиональной деятельности. Для этого осуществляется поэтапный контроль степени освоения компетенций. В таблице приведены этапы освоения компетенций и виды оценочных средств, предназначенных для оценивания компетенций на разных стадиях обучения студентов.

№ п/п	№ раздела	Контролируемые компетенции	Вид оценочного средства
1	Раздел 1	ОПК-3, ОПК-4, ПК-23	Контроль в виде тестов Разноуровневые задания к лабораторному практикуму
2	Раздел 2	ОПК-3, ОПК-4, ПК-23	Контроль в виде тестов Разноуровневые задания к лабораторному практикуму
3	Раздел 3	ОПК-3, ОПК-4, ПК-23	Контрольные работы Разноуровневые задания к лабораторному практикуму
4	Промежуточная аттестация	ОПК-3, ОПК-4, ПК-23	Экзамен

Контроль качества освоения студентами дисциплины осуществляется непрерывно в течение всего периода обучения с использованием балльно-рейтинговой системы (БРС). Индикатором сфомированности компетенции является начисление студенту баллов за контроль в виде тестов, лабораторные задания, домашние задания.

11.1. Оценочные средства для входного контроля

Не требуется.

11.2. Оценочные средства текущего контроля формируются в соответствии с Положением о балльно-рейтинговой системе.

Выполнение тестов, лабораторных и домашних заданий в информационно-образовательной среде дает 50 баллов.

Для успешной сдачи экзамена требуется осветить два теоретических вопроса, каждый из которых дает 20 баллов. После процедуры выбора, экзаменуемому предоставляется 60 минут для подготовки ответа на выбранные вопросы. После ответа на основные вопросы экзаменуемому может быть задан дополнительный вопрос, оценивающийся в 10 баллов.

Критерии оценивания теоретических вопросов:

20 баллов: экзаменуемый самостоятельно и без наводящих вопросов осветил все основные разделы вопроса, ввел необходимые определения, описал необходимые алгоритмы; привел примеры.

10-15 баллов: экзаменуемый, после задания уточняющих и наводящих вопросов, в достаточно полной степени осветил содержание вопроса, ввел определения, возможно не осветил некоторые шаги алгоритмов, привел примеры.

5 баллов: экзаменуемый не осветил вопрос, но ответил на наводящие, уточняющие и дополнительные вопросы, привел основные определения, но не привел необходимые примеры.

0 баллов: не ответил на выбранные им вопросы экзамена и уточняющие вопросы экзаменатора.

Тест к разделу Аутентичное шифрование

Вопрос 1 Предположим, что система MAC (S,V) используется для защиты файлов в файловой системе компьютера. К каждому файлу добавляется специальный tag. Алгоритм S в своей работе использует только содержимое файла. Для каких атак подобная система MAC является уязвимой?

Варианты ответов

Выполнить конкатенацию двух файлов в системе и создать третий.

Удалить последний байт в файле.

Изменить время модификации файла.

Поменять местами два файла в файловой системе.

Добавить новые данные в файл.

Вопрос 2 Пусть система MAC (S,V) является безопасной. Система определена над пространством (K,M,T), где M={0,1}n и T={0,1}128. Какие из предложенных ниже конструкций являются безопасными MAC?

Варианты ответов

$S'(k,m)=S(k,m)$ и $V'(k,m,t)=\{V(k,m,t),1,m\neq 0n;otherwise$

$S'(k,m)=S(k,m \,//\, m)$ и $V'(k,m,t)=V(k,m \,//\, m,t)$

$S'(k,m)=S(k,m)[0,...,126]$ и $V'(k,m,t)=[V(k,m,t \,//\, 0)orV(k,m,t \,//\, 1)]$

$S'(k,m)=\{S(k,1n),S(k,m),m=0n;otherwise$ и $V'(k,m,t)=\{V(k,1n,t),V(k,m,t),m=0n;otherwise$

$S'(k,m)=[S(k,m),S(k,0n)]$ и $V'(k,m,(t1,t2))=[V(k,m,t1)andV(k,m,t2)]$

$S'(k,m)=[t=S(k,m),output(t,t)]$ и $V'(k,m,(t1,t2))=\{V(k,m,t1),0,t1=t2;otherwise$

Вопрос 3 Схема вычисления кодов аутентификации сообщений ECBC-MAC использует фиксированный вектор инициализации (в лекции использовался нулевой вектор). Предположим, что вместо фиксированного, для каждого сообщения используется случайный вектор IV и этот вектор добавляется к tag. Другими словами, S(k,m)=(r,ECBCr(k,m)), где r это случайный IV. Алгоритм проверки по ключу k, сообщению m и тегу (r,t) выдает 1 ("yes"), если t=ECBCr(k,m).

Такая схема построения MAC является небезопасной. Атакующий может получить тег (r,t) для сообщения m, состоящего из одного блока. Теперь он способен сконструировать новый тег для нового сообщения. Укажите пару тег, сообщение, которую может сконструировать атакующий.

Варианты ответов

Тег ($m \oplus t$,t) является правильным для сообщения $m \oplus 1n$.

Тег ($r \oplus t$,r) является правильным для сообщения m.

Тег ($r \oplus t$,m) является правильным для сообщения 0n.

Тег ($r \oplus 1n$,t) является правильным для сообщения $m \oplus 1n$.

Вопрос 4 Алиса намеревается организовать широковещательную отправку сообщений для 6 получателей B1,…,B6. Секретность сообщений не важна, важна целостность и аутентичность. Каждый из получателей должен быть уверен, что сообщение получено от Алисы.

Для своей задачи Алиса решила использовать MAC. Предположим, что Алиса и 6 пользователей работают с одним общим ключом k. Тогда каждый из B1,…,B6 при получении сообщения может проверить правильность тега. Но эта схема небезопасна, т.к., например, пользователь B1 может создать сообщение, прикрепить к нему тег, полученный на ключе k и отправить в сеть. При этом, пользователи B2,…,B6 будут уверены, что сообщение получено от Алисы.

Вместо этого, Алиса сгенерировала 4 секретных ключа S={k1,…,k4}. Каждому пользователю Bi она передала подмножество ключей $Si \subseteq S$. Теперь Алиса при передаче сообщения прикладывает к нему четыре тега, каждый из которых получен на одном из четырех ключей. Когда пользователь Bi получает сообщение он проверяет только те теги сообщения, для которых у него есть ключи.

Выберите правильный способ, которым Алиса может распределить случайные ключи так, чтобы никто из пользователей не смог бы создавать сообщения для других пользователей правильные сообщения от имени Алисы

Варианты ответов

S1={k1,k2}, S2={k1}, S3={k1,k4}, S4={k2,k3}, S5={k2,k4}, S6={k3,k4}

S1={k2}, S2={k2,k3}, S3={k3,k4}, S4={k1,k3}, S5={k1,k2}, S6={k1,k4}

S1={k1,k2}, S2={k2,k3}, S3={k3,k4}, S4={k1,k3}, S5={k1,k2}, S6={k1,k4}

S1={k1k2}, S2={k1,k3}, S3={k1,k4}, S4={k2,k3}, S5={k2,k4}, S6={k3,k4}

Вопрос 5 Рассмотрим схему ECBC-MAC на основе шифра AES. Предположим, что мы вычислили тег для длинного сообщения m, состоящего из n блоков. Длина сообщения кратна длине блока AES и набивка сообщения не использовалась. Пусть сообщение m′ получено из сообщения m заменой последнего бита на противоположный.

Сколько операций AES необходимо выполнить для того, чтобы из тега сообщения m и ключа MAC получить тег нового сообщения m′.

Варианты ответов
2
n−1
3
4

Вопрос 6 Пусть H:M→T стойкая к коллизиям хеш функция. Какая из следующий функций также будет стойкой к коллизиям.

Варианты ответов
H′(m)=H(m[0,…,|m|−2])
H′(m)=H(m) // H(m)
H′(m)=H(H(m))
H′(m)=H(m)⊕H(m⊕1|m|)
H′(m)=H(m)[0,…,31]
H′(m)=H(|m|), где |m| есть длина сообщения m
H′(m)=H(m // m)
H′(m)=H(m // 0)
H′(m)=H(m)⊕H(m)

Лабораторная работа. Шифрование AES.

Шифрование сообщений в Java

Для работы с алгоритмами и протоколами шифрования будем использовать стандартный API JCA, содержащийся в JDK 1.6. Приведенные ниже классы и примеры будут работать как со стандартным криптопровайдером JCE, так и с любым другим криптопровайдером, поддерживающим реализованный в Java API (например, Bouncy Castle).

При шифровании данных управление ключами может происходить двумя способами: шифрование на случайном ключе, шифрование на известном ключе.

Рассмотрим процедуру симметричного шифрования на случайном ключе. Ключ создается объектом класса KeyGenerator из пакета javax.crypto. Для создания генератора нужно воспользоваться специальным методом

```
KeyGenerator kg = KeyGenerator.getInstance("DES");
```

где аргументом метода выступает имя алгоритма, для которого происходит генерация ключа (в данном случае DES). Список доступных алгоритмов здесь. После создания генератора необходимо использовать метод init() для его инициализации и метод generateKey() — для создания ключа.

Например, следующий код создает ключ шифрования на основании безопасного генератора случайных чисел для алгоритма AES:

```
KeyGenerator kg = KeyGenerator.getInstance("AES");
kg.init(new SecureRandom());
SecretKey key = kg.generateKey();
```

При использовании в качестве ключа готовой последовательности байт нужно воспользоваться классом SecretKeySpec из пакета javax.crypto.spec. Ключ создается через параметры конструктора, в которых указывается массив байт, содержащих ключ, и алгоритм шифрования. Например, следующий код генерирует ключ для алгоритма AES на основе строки из шестнадцати символов "1234567890123456":

```
byte[] newKey = "1234567890123456".getBytes();
SecretKey key = new SecretKeySpec(newKey, "AES");
```

Класс Cipher является основным для работы с алгоритмами шифрования и расположен в пакете javax.crypto. Этот класс предоставляет возможность работы как с алгоритмами шифрования, так и с режимами работы шифров, схемами дополнения сообщений. В зависимости от параметров инициализации, объект класса Cipher может работать как в режиме шифрования, так и в режиме расшифрования, с симметричными и асимметричными алгоритмами.

Для создания нового объекта класса следует воспользоваться специальным методом:

```
Cipher c = Cipher.getInstance(algorithm)
```

где algorithm представляет собой строку вида: **алгоритм шифрования\режим работы\алгоритм дополнения**

Имена доступных алгоритмов шифрования и их настройки прописаны в стандарте Java, и, обычно, они совпадают с общепринятыми обозначениями. Для дополнения сообщений, в основном, используются два стандарта: NoPadding — для шифрования без дополнения сообщения (например, для режима шифрования CTR) и PKCS5Padding — для дополнения сообщения до размера, кратного размеру блока используемого алгоритма (например, для режима CBC).

Для реализации процедуры шифрования тексты должны быть представлены в виде массива байт. Опишем шифрование сообщения по шагам:

Создание объекта Cipher с использованием метода getInstance.

Инициализация объекта Cipher с помощью метода init() для шифрования или расшифрования. Этот метод принимает на вход два обязательных параметра и один необязательный параметр. Первый параметр указывает режим работы алгоритма: Cipher.ENCRYPT_MODE или Cipher.DECRYPT_MODE. Второй параметр — ключ шифрования. Тип ключа зависит от используемого алгоритма шифрования. Третьим параметром указывается вектор инициализации алгоритма: объект класса IvParameterSpec.

Шифрование или расшифрование производится методами update() для работы с сообщением в режиме потока (обработка сообщения по частям) и doFinal() для обработки сообщения полностью.

Ниже приведен пример шифрования сообщения OpenText шифром AES в режиме CBC на случайном ключе и случайном векторе инициализации:

```
KeyGenerator kg = KeyGenerator.getInstance("AES");
kg.init(new SecureRandom());
SecretKey key = kg.generateKey();

SecureRandom sr = new SecureRandom();
byte[] IV = new byte[16];
sr.nextBytes(IV);
```

```
Cipher c = Cipher.getInstance("AES/CBC/PKCS5Padding");

c.init(Cipher.ENCRYPT_MODE, key, new
IvParameterSpec(IV));

byte[] encryptedData = c.doFinal(OpenText.getBytes());
```

Так как массив байт невозможно вывести в текстовый файл, то результат шифрования нужно закодировать либо в шестнадцатеричном формате, либо в формате Base64.

Работа с форматом Base64 в Java обеспечивается двумя классами BASE64Encoder и BASE64Decoder из пакета sun.misc. Для выполнения преобразований используются методы encode и decodeBuffer соответствующих классов.

По полученным параметрам выполнить расшифрование заданного шифр-текста. Исходный открытый текст зашифрован алгоритмом AES в режиме CBC. В варианте содержится шифр-текст, вектор инициализации, ключ шифрования. Все данные представлены в формате BASE64. Требуется расшифровать шифр-текст.

1. Расшифрование нужно проводить по следующей схеме:
2. Раскодировать ключ, вектор инициализации и шифр-текст из формата BASE64 в последовательности байт.
3. С использованием объекта класса SecretKeySpec по последовательности байт ключа создать объект класса SecretKey.
4. Создать объект класса Cipher с параметром "AES/CBC/ISO10126Padding".
5. При инициализации шифра требуется подключить вектор инициализации с помощью объекта класса IvParameterSpec.
6. Выполнить расшифрование шифр-текста.

В результате расшифрования должен получиться массив байт, который в кодировке ASCII можно интерпретировать как осмысленный текст. Для преобразования массива байт в строку можно воспользоваться конструктором класса String.

Лабораторная работа. Метод шагов

Пусть дано некоторое большое простое число p и g — некоторый элемент из Z^*_p. Предположим, что известен элемент h, такой что h=g^x \mod p. Требуется найти x = ind_g h \mod p.

Предположим, что 1 \leqslant x \leqslant 2^{40}. Тривиальный алгоритм поиска x потребует перебора всех 2^{40} вариантов.

Пусть B=2^{20}. Так как x < B^2, то можно записать следующее разложение: x=x_0B+x_1, где x_0,x_1 принадлежат отрезку [0,B-1]. Тогда h=g^x=g^{x_0B+x_1}=\left(g^B\right)^{x_0}\cdot g^{x_1} \mod p.

Иначе \frac{h}{g^{x_1}}=\left(g^B\right)^{x_0}\mod p.

Так как переменные x_0 и x_1 теперь расположены по разным сторонам равенства, то можно попытаться подобрать их значения, используя метод встречи посередине. Для этого потребуется:

- Построить таблицу всех возможных значений левой части равенства для x_1 = 0,1, \ldots ,2^{20}.

- Для каждого значения x_0 = 0,1, \ldots ,2^{20} проверить, найдется ли число \left(g^B\right)^{x_0} \!\!\!\! \mod p в созданной на предыдущем шаге таблице. Если равенство найдено, то пара (x_0,x_1) дает решение x=x_0B+x_1.

При выполнении алгоритма потребуется 2^{20} операций для построения таблицы и еще 2^{20} просмотров этой таблицы. Для составления таблицы потребуется использование класса BigInteger. Временное хранилище для таблицы поиска лучше организовать с помощью стандартной коллекции HashMap.

Лабораторная работа. Атака на RSA при использовании закрытого ключа

Даны числа N и e, формирующие открытый ключ преобразования RSA. Известно, что значение закрытого ключа d мало (это можно предположить исходя из значения ключа e). С использованием подходящих дробей требуется получить значение ключа d, и затем на основе его значения решить задачу факторизации для числа N.

Вычисление закрытого ключа

Известно, что при $d<\frac{\sqrt[4]{N}}{3}$ выполняется соотношение $$ \left|\frac{e}{N}-\frac{k}{d}\right|<\frac{1}{2d^2}. $$

Таким образом, для нахождения закрытого ключа можно воспользоваться подходящими дробями согласно алгоритму:

Представить дробь $\frac{e}{N}$ в виде цепной дроби.

Для найденной цепной дроби построить последовательность подходящих дробей.

Начиная с некоторого номера знаменатель подходящей дроби можно рассматривать в качестве кандидата на закрытый ключ d.

Для каждого кандидата проверить условие корректности закрытого ключа. Согдасно определению открытого и закрытого ключей для любого сообщения m должно выполняться соотношение $$m^{ed} \equiv m (N)$$.

Знаменатель подходящей дроби, удовлетворяющий условию корректности можно рассматривать в качестве закрытого ключа преобразования RSA.

Факторизация модуля при известном закрытом ключе

Пусть известны модуль N, открытый ключ e и закрытый ключ d, полученный на предыдущем шаге.

Возьмем произвольное число x. Обозначим за $y_1 = x^{\frac{ed-1}{2}}$ (это возможно, поскольку ed это нечетное число). Тогда $$y_1^2\equiv 1 (N).$$ Иначе $$(y_1-1)(y_1+1)\equiv0(N).$$ Возможны следующие варианты:

Если $y_1=1$. Если $\frac{ed-1}{2}$ — число четное, то введем $y_2 = x^{\frac{ed-1}{4}}$ и проанализируем значение y_2 (все рассуждения справедливы для y_2). Иначе потребуется выбрать другое основание x.

Если $y_1=-1$. То потребуется выбрать другое основание x.

Если y_1 есть число отличное от 1 и -1, то тогда делитель числа N вычисляется по формуле:$$p=НОД(y_1-1,N).$$

Лабораторная работа. Шифрование с использованием библиотеки OpenSSL

Механизм шифрования сообщений

Цель данного задания заключается в ознакомлении с основными возможностями криптографической библиотеки OpenSSL. Библиотека, в основном, предназначена для организации взаимодействия на основе инфраструктуры открытых ключей. Но в последних версиях библиотеки реализована функциональность по симметричному шифрованию сообщений. Библиотека работает из командной строки в пакетном режиме.

Версии библиотек: openssl-1.0.2-i386-win32.zip, openssl-1.0.2-x64_86-win64.zip

Требуется сгенерировать случайный ключ шифрования, зашифровать алгоритмом AES некоторый открытый текст и передать ключ шифрования через преобразование RSA с помощью алгоритма шифрования сеансового ключа и зашифрованное сообщение на проверку преподавателю.

Часть 1. Шифрование сообщения

С помощью OpenSSL зашифруйте сообщение. Шифрование нужно проводить по следующей схеме:

Выберите некоторое сообщение (содержащее в том числе номер зачетной книжки) и поместите его в файл. Сообщение может быть любого разумного размера и должно читаться без использования специальных программ.

Сгенерируйте с помощью команды openssl rand (doc) случайный ключ необходимой для алгоритма AES-128-CBC длины и сохраните его в файл в шестнадцатеричном формате .

Сгенерируйте с помощью команды openssl rand вектор инициализации необходимой для алгоритма AES-128-CBC длины и сохраните его в файл в шестнадцатеричном формате.

С помощью команды openssl enc (doc) зашифруйте алгоритмом AES-128-CBC файл с открытым текстом на сгенерированных ранее ключе и векторе инициализации. **NB!** При шифровании следует использовать именно ключ шифрования, а не пароль.

Сохраните ключ шифрования и вектор инициализации в одном файле в двух разных строках без дополнительных комментариев.

Часть 2. Самостоятельное шифрование ключа симметричного шифрования

В этой части лабораторной потребуется создать собственную пару RSA-ключей и пройти обе фазы алгоритма шифрования симметричного ключа. Работать можно по следующей схеме:

С помощью команды openssl genrsa (doc) сгенерируйте закрытый RSA-ключ. Длина ключа должна составлять 2048 бит. Для большей надежности ключ можно зашифровать паролем.

С помощью команды openssl rsa (doc) сгенерируйте открытый RSA-ключ.

На открытом RSA-ключе с помощью openssl rsautl (doc) зашифруйте файл с ключом и вектором инициализации.

Проведите проверку своих действий и расшифруйте на закрытом RSA-ключе полученный на предыдущем шаге файл.

Проверьте, что расшифрование прошло успешно и в итоге получился тот же самый ключ симметричного шифрования.

Расшифруйте сообщение.

Часть 3. Отправка результатов на проверку

Для отправки результатов шифрования на проверку необходимо выполнить следующие действия:

Загрузите открытый RSA-ключ преподавателя.

На загруженном открытом ключе с помощью openssl rsautl зашифруйте файл с ключом симметричного шифрования и вектором инициализации.

Отправьте на проверку файл с зашифрованным текстом и файл с зашифрованным симметричным ключом. Свои RSA-ключи в ИОС Домик загружать не требуется.

Проверка результата шифрования будет проводиться следующими командами:

```
openssl rsautl -decrypt -inkey private.pem -in key.hex.enc -out key.hex
openssl enc -aes-128-cbc -d -in ct.txt -out ot.txt -K 000000 -iv 000000 -p
```

11.3. Оценочные средства для промежуточной аттестации.

Вопросы к экзамену (пятый семестр)

1. Общие принципы криптографии. Защита от несанкционированного доступа. Основные способы криптоанализа простых шифров. Идеальный шифр.
2. Шифр Вернама. Лемма о теоретически стойком шифре. Потоковые шифры. Атаки на потоковые шифры. Двухразовый блокнот.
3. Примеры потоковых шифров. RC4 и его уязвимости.
4. Безопасность потоковых генераторов. Статистический тест для генератора. Криптографическое определение псевдослучайного генератора.
5. Принцип итерирования. Понятие псевдослучайной функции и псевдослучайной перестановки. Шифр DES. Описание структуры. Теорема о трехраундовой сети Фейстеля.
6. Шифр AES. Описание структуры и принципы работы. Переборные атаки на блочные шифры.
7. Построение блочных шифров на основе псевдослучайных генераторов. Шифрование нескольких сообщений на одном ключе. Атака на основе выборочного открытого текста. Пример шифра нестойкого к атаке на основе выборочного открытого текста.
8. Случайное шифрование. Одноразовое шифрование. Режим CBC. Шифрование с предсказанным значением вектора инициализации. CBC для одноразового шифрования. Набивка блоков. Режим CTR. Сравнение режимов CBC и CTR. Режим OFB.
9. Коды аутентификации сообщений. Построение кодов аутентификации больших сообщений. Конструкции CBC-MAC и NMAC. Их эффективность. Схема атаки с использованием коллизий.
10. Конструкция HMAC. Построение кодов аутентификации больших сообщений. Механизмы набивки сообщений.
11. Построение кодов аутентификации сообщений из стойких к коллизиям функций. Атака на основе парадокса дней рождений.
12. Конструкция Меркла-Дамгарда для построения хеш-функций. Построение безопасных сжимающих функций. Хеш-функции на основе блочных шифров.
13. Необходимость применения аутентичного шифрования. Понятие целостности шифр-текста. Атака на основе выборочного шифр-текста. Аутентичное шифрования на примере протокола TLS.
14. Стандарт PKCS1. Атаки на RSA. Атака при малом значении закрытого ключа.
15. Задача дискретного логарифмирования. Метод ключевого обмена Диффи-

Хеллмана.

16. Алгоритмы дискретного логарифмирования. Алгоритм шагов.
17. Преобразование Эль-Гамаля. Шифрование и создание подписи сообщений.

Разработчики: доцент Рябец Л.В.