**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра информационной безопасности**

отчет

**по лабораторной работе №6**

**по дисциплине «Криптография и защита информации»**

Тема: **Изучение хэш-функций**

|  |  |  |
| --- | --- | --- |
| Студентка гр. 8383 |  | Гречко В.Д. |
| Преподаватель |  | Племянников А.К. |

Санкт-Петербург

2021

**Цель работы.**

Исследование хэш-функций MD5, SHA-256, SHA-512, SHA-3, кода контроля целостности HMAC и анализ атак дополнительной коллизии на хэш-функцию. Получить практические навыки работы с хэш-функциями и атакой на них, в том числе и в программном продукте Cryptool 1 и 2.

**Исследование лавинного эффекта MD5, SHA-1, SHA-256, SHA-512.**

Основные параметры и обобщенная схема хэш-функций MD5, SHA-1

MD5:

* MD буфер, входное сообщение, дополненное недостающими битами (от 1 до 448 бит) и 64-битным представлением длины исходного сообщения, функция сжатия (MD буфер, массив вычисляемых величин (32 бита) – T, 4 элементарных функции (), – циклический сдвиг).
* Схема:

|  |  |  |
| --- | --- | --- |
|  |  |  |

SHA-1:

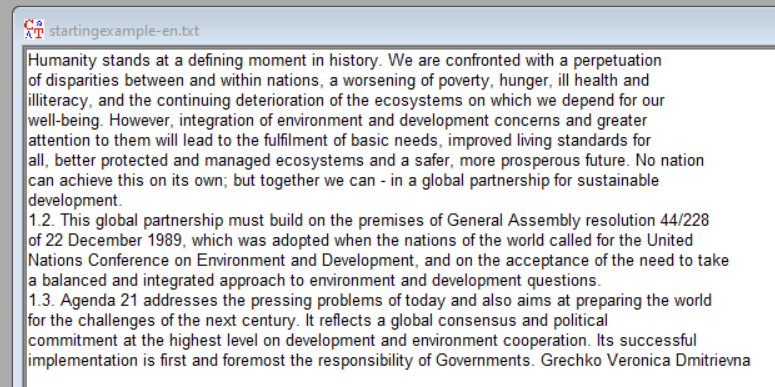
* MD буфер, входное сообщение, дополненное недостающими битами (от 1 до 448 бит) и 64-битным представлением длины исходного сообщения, функция сжатия (MD буфер, константа , 4 элементарных функции (), – циклические сдвиги, – 32-битовые слова (80 слов)).
* Схема:

|  |  |  |
| --- | --- | --- |
|  |  |  |

Задание.

1. Открыть текст не менее 1000 знаков. Добавить свое ФИО последней строкой. Перейти к утилите Indiv.Procedures->Hash->Hash Demonstration..

Возьмем текст из reference в CrypTool и добавим своё ФИО. Результат представлен на рис. 1.



*Рисунок 1 – Текст для хэш-функций*

1. Задать хэш-функцию, подлежащую исследованию: MD5, SHA-1, SHA-256, SHA-512.,
2. Для каждой хэш-функции повторить следующие действия:
   1. Измените (добавлением, заменой, удалением символа) исходный файле
   2. Зафиксировать количество измененных битов в дайджесте модифицированного сообщения.
   3. Вернуть сообщение в исходное состояние.
3. Выполните процедуру 3 раза (добавлением, заменой, удалением символа) и подсчитайте среднее количество измененных бит дайджеста. Зафиксировать результаты в таблице.

Сведем выполнение пунктов 2-4 в единую таблицу 1 для простоты анализа.

*Таблица 1 – Фактические и усредненные параметры лавинного эффекта*

|  |  |  |  |
| --- | --- | --- | --- |
| Хэш-функция | Тип изменения | Количество измененных битов в дайджесте | Среднее количество измененных бит дайджеста |
| MD5 | добавление | 60 бит – 46% | 65 бит – 51% |
| замена | 69 бит – 54% |
| удаление символа | 68 бит – 53% |
| SHA-1 | добавление | 83 бит – 52% | 80 бит – 50% |
| замена | 80 бит – 50% |
| удаление символа | 78 бит – 49% |
| SHA-256 | добавление | 116 бит – 45% | 127 бит – 50% |
| замена | 131 бит – 60% |
| удаление символа | 135 бит – 53% |
| SHA-512 | добавление | 247 бит – 48% | 253 бит – 49% |
| замена | 257 бит – 51% |
| удаление символа | 255 бит – 50% |

**Хэш-функция SHA-3.**

Основные параметры и обобщенная схема хэш-функции Keccak Hash, (SHA-3) на основе изученной презентации

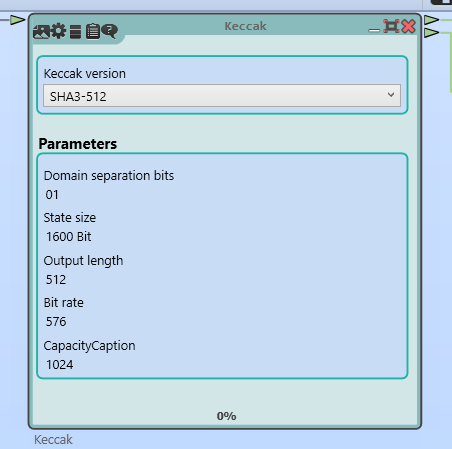
* Основные параметры: исходное сообщение , внутреннее состояние , функция - многораундовая бесключевая псевдослучайная перестановка, длина слова . Размер части состояния, который записывается и считывается, называется «скоростью» и обозначается , а размер части, которая нетронута вводом / выводом, называется «ёмкостью» и обозначается .
* Схема:

|  |  |
| --- | --- |
|  |  |

Задание.

1. Открыть шаблон Keccak Hash (SHA-3) в Cryptool 2
2. В модуле Keccak сделать следующие настройки:
   1. Adjust manually=ON
   2. Keccak version= SHA3-512

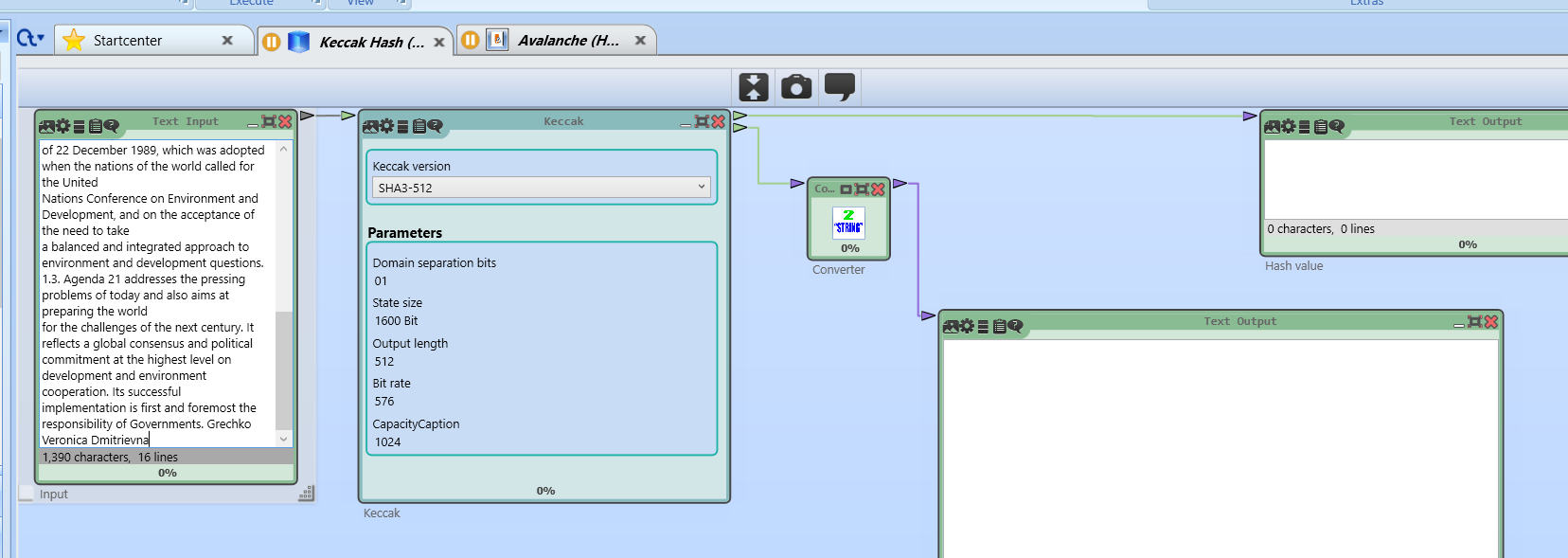
Шаблон с установленными настройками представлен на рис. 2



*Рисунок 2 – Изменённые настройки*

1. Загрузить файл из предыдущего задания

Шаблон с исходным текстом представлен на рис. 3.

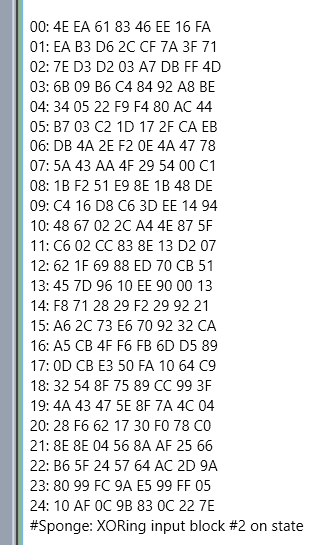


*Рисунок 3 – Загруженный текст*

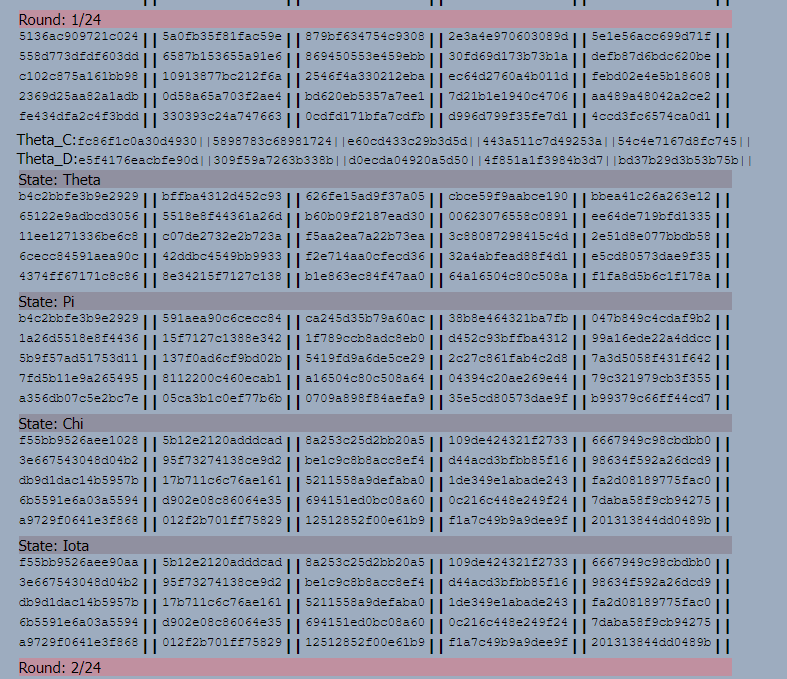
1. Запустить проигрывание шаблона в режиме ручного управления:
   1. Сохранить скриншоты преобразований первого раунда
   2. Сохранить скриншот заключительной фазы
   3. Сохранить значение дайджеста

Запустим проигрывание шаблона в режиме ручного управления.

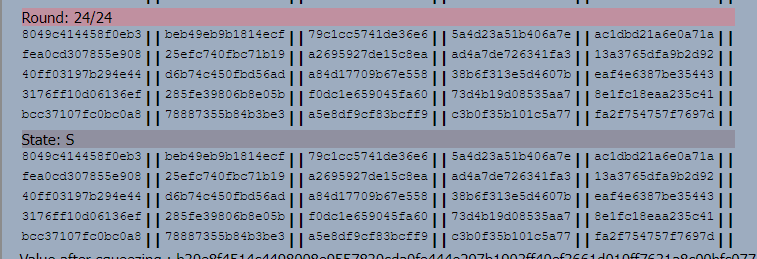
Так как CrypTool 2 показывает только 20 раундов, для демонстрации последнего был использован сторонний сайт. Скриншоты преобразований первого раунда представлены на рис. 4 -5. Скриншот заключительной фазы представлен на рис 6. Значение дайджеста представлено на рис 7.



*Рисунок 4 – Преобразование первого раунда в CrypTool 2*



*Рисунок 5 – Подробные преобразования в первом раунде*



*Рисунок 6 – Заключительная фаза преобразований*



*Рисунок 7 – Значение дайджеста*

Значение дайджеста:

B3 0E 8F 45 14 C4 49 80 08 E9 55 78 30 CD A0 FE 44 4E 29 7B 19 03 FF 40 EF 36 61 D0 10 FF 76 31 A8 C0 0B FC 07 71 C3 BC CF 4E 81 B1 B9 9E B4 BE 19 1B C7 FB 40 C7 EF 25 AD 56 BD 0F 45 4C B7 D6

1. Вычислить значения дайджеста для модифицированных текстов из предыдущего задания

Сведем результат вычислений в таблицу 2.

*Таблица 2 – Значения дайджеста при модификации исходного текста*

|  |  |
| --- | --- |
| Тип изменения | Значение дайджеста |
| добавление | 52 7B 3A A0 B6 9D FB B0 AB 70 0A 72 AF E2 9F D1 53 E5 01 08 12 68 F4 50 E6 E0 EF D0 53 FA FB 87 B7 CB 6C 6C BB 75 26 54 03 77 F7 F3 D6 C2 23 73 57 39 5E EE 89 19 0F 9E 49 6B D1 8C 4A B2 C3 52 |
| замена | 5B 17 B6 F0 AD 90 17 93 2E 0C 68 AF 6E FA 8C 45 32 7A 15 52 24 B1 6E 01 7F FD 5E 1D EB FA 44 C1 86 51 FB 81 D8 B4 DB 12 CB 54 38 AA 08 59 57 D9 48 72 D4 1E 6F F2 5A A2 6B 8F 10 DA E5 CA F4 8A |
| удаление символа | 29 8E 31 80 83 D7 A4 79 AE 83 68 C0 4A BE 31 C0 15 83 68 8C 2A 9E E0 40 75 7F 64 D7 82 3F CB 12 F8 A9 7C 70 98 8A 35 ED 6C 86 9D 15 39 EE B0 4E C5 C4 32 C3 0C 37 E2 6F 7C AE 15 E6 C9 84 7E 22 |
| Исходное значение | B3 0E 8F 45 14 C4 49 80 08 E9 55 78 30 CD A0 FE 44 4E 29 7B 19 03 FF 40 EF 36 61 D0 10 FF 76 31 A8 C0 0B FC 07 71 C3 BC CF 4E 81 B1 B9 9E B4 BE 19 1B C7 FB 40 C7 EF 25 AD 56 BD 0F 45 4C B7 D6 |

1. Подсчитать лавинный эффект с помощью самостоятельно разработанной автоматизированной процедуры

Для подсчета лавинного эффекта воспользуемся встроенным шаблоном Avalanche effect в CrypTool2 и изменим хэш-функцию на Keccak.

Представим полученные значения лавинного эффекта в таблице 3 для большей наглядности.

*Таблица 3 – Фактические и усредненные параметры лавинного эффекта*

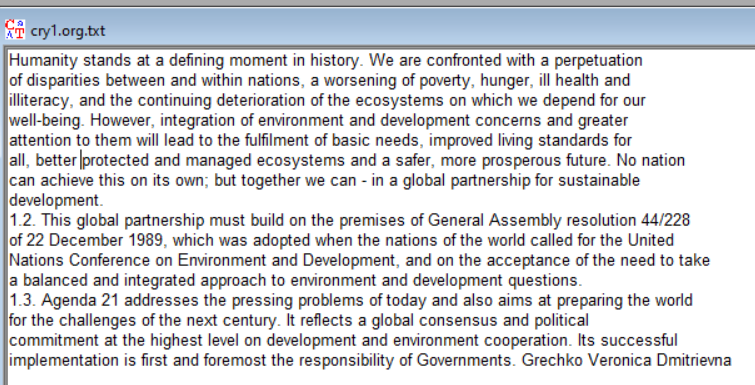
|  |  |  |  |
| --- | --- | --- | --- |
| Хэш-функция | Тип изменения | Количество измененных битов в дайджесте | Среднее количество измененных бит дайджеста |
| Keccak Hash (SHA-3) | добавление | 253 бит – 49.4% | 256 бит – 50 % |
| замена | 264 бит – 51.6 % |
| удаление символа | 252 бит – 49.2% |

**Контроль целостности по коду HMAC.**

Задание.

1. Выбрать текст на английском языке (не менее 1000 знаков), добавить собственное ФИО и сохранить в файле формата .TXT.

Выбранный и изменённый текст приведен на рис. 8.

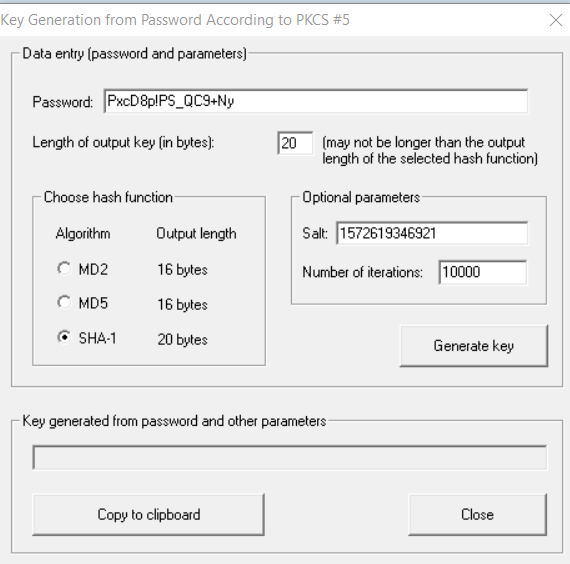


*Рисунок 8 – Исходный текст*

1. Придумать пароль и сгенерировать секретный ключ утилитой Indiv.Procedures->Hash-> Key Generation из Cryptool 1. Сохранить ключ в файле формата .TXT. Прочитать Help к этой утилите.

Пароль: PxcD8p!PS\_QC9+Ny

Параметры генерации:



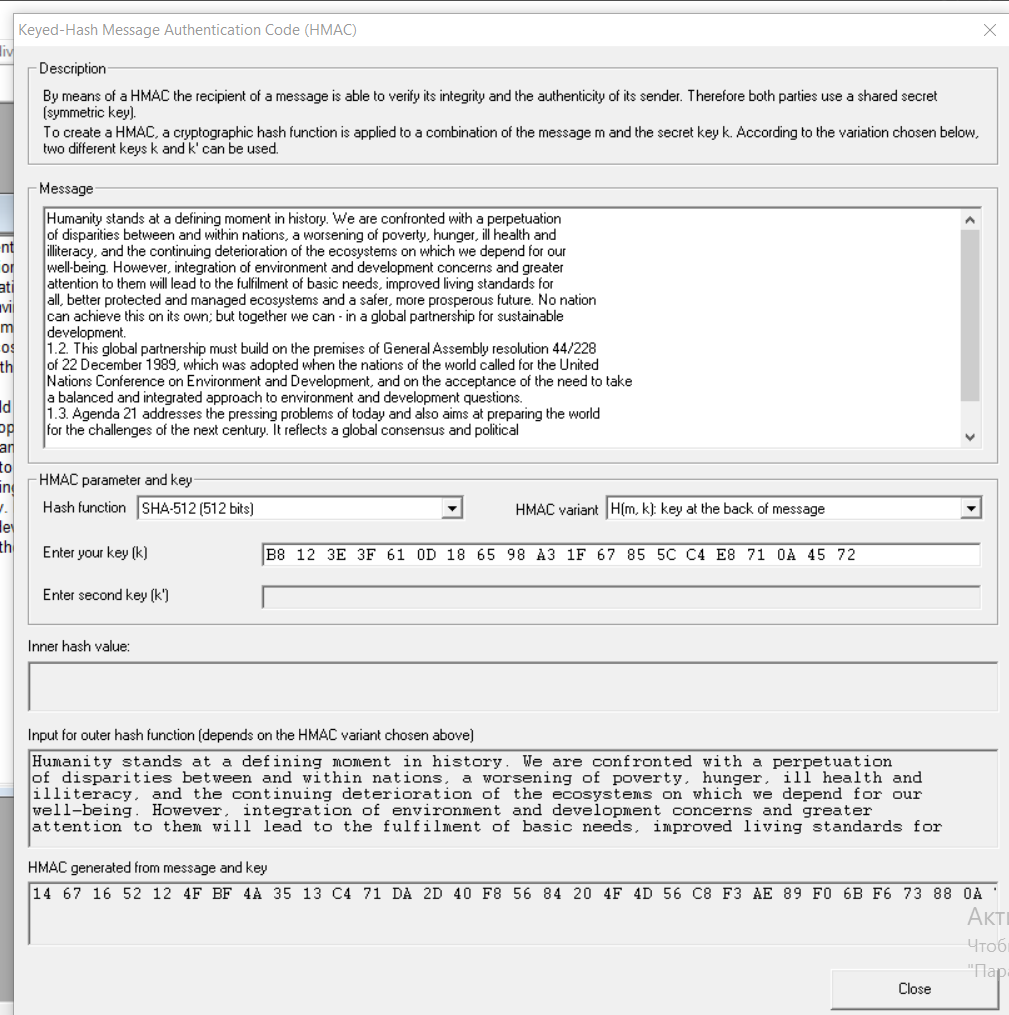
*Рисунок 9 – Параметры генерируемого ключа*

Сгенерированный секретный ключ:

B8 12 3E 3F 61 0D 18 65 98 A3 1F 67 85 5C C4 E8 71 0A 45 72

1. Сгенерировать HMAC для имеющегося текста и ключа с помощью утилиты Indiv.Procedures->Hash-> Generation of HMACs. Сохранить HMAC в файле формата .TXT. Прочитать Help к этой утилите.

Параметры генерации:



*Рисунок 10 – Параметры генерации HMAC*

Полученный HMAC: 14 67 16 52 12 4F BF 4A 35 13 C4 71 DA 2D 40 F8 56 84 20 4F 4D 56 C8 F3 AE 89 F0 6B F6 73 88 0A 70 4D 0B DC 92 8E A1 0A 93 47 7B 53 26 C4 26 A9 20 DE 6E 3B 9D 74 B6 BC 9B 30 7F 94 48 6E 11 77

1. Передать пароль, HMAC (и его характеристики), исходный текст и модифицированный текст коллеге, не раскрывая, какой текст является корректным. Попросите коллегу определить это самостоятельно.

В таблице 4 представлены полученные значения у коллеги.

***Описание действий передающий стороны на примере выполненного задания:***

Сначала отправитель должен придумать достаточно сложный пароль, далее необходимо настроить достаточно надежную генерацию ключа – CrypTool предлагает использовать по умолчанию seed и 1000 итераций. После этого необходимо определить параметры для HMAC – хэш-функцию и конструкцию добавления (H(M, k), H(k, M) и т.д.). После этого вычисляется сам HMAC и сообщение передаётся получателю.

***Описание действий принимающей стороны на примере выполненного задания:***

Предполагается, что получатель знает характеристики для генерации секретного ключа. Тогда, первым шагом будет сгенерировать секретный ключ по заданным параметрам. Далее, получатель должен вычислить хэш полученного сообщения вместе с секретным ключом. Потом он сравнивает полученное значение хэша с тем, которое он получил. Сообщение является подлинным только в том случае, если два значения хэша равны.

*Таблица 4 – Эксперимент*

|  |  |
| --- | --- |
| Исходный текст:  Economic policies of individual countries and international economic relations both have  great relevance to sustainable development. The reactivation and acceleration of development  requires both a dynamic and a supportive international economic environment and determined  policies at the national level. It will be frustrated in the absence of either of these  requirements. A supportive external economic environment is crucial. The development process  will not gather momentum if the global economy lacks dynamism and stability and is beset with  uncertainties. Neither will it gather momentum if the developing countries are weighted down  by external indebtedness, if development finance is inadequate, if barriers restrict access  to markets and if commodity prices and the terms of trade of developing countries remain  depressed. The record of the 1980s was essentially negative on each of these counts and needs  to be reversed. The policies and measures needed to create an international environment that  is strongly supportive of national development efforts are thus vital. International  cooperation in this area should be designed to complement and support - not to diminish or  subsume - sound domestic economic policies, in both developed and developing countries, if global progress towards sustainable development is to be achieved. Kireev Konstantin Aleksandrovich | Модифицированный текст:  Economic policies of individual countries and international economic relations both have  great relevance to sustainable development. The reactivation and acceleration of development  requires both a dynamic and a supportive international economic environment and determined  policies at the national level. It will be frustrated in the absence of either of these  requirements. A supportive external economic environment is crucial. The development process  will not gather momentum if the global economy lacks dynamism and stability and is beset with  uncertainties. Neither will it gather momentum if the developing countries are weighted down  by external indebtedness, if development finance is inadequate, if barriers restrict access  to markets and if commodity prices and the terms of trade of developing countries remain  depressed. The record of the 1980s was essentially negative on each of these counts and needs  to be reversed. The policies and measures needed to create an international environment that  is strongly supportive of national development efforts are thus vital. International  cooperation in this area should be designed to complement and support - not to diminish or  subsume - sound domestic economic policies, in both developed and developing countries, if global progress towards sustainable development is to be achieved. Kireev Konstantin Aleksandrovich |
| Переданный HMAC:  14 67 16 52 12 4F BF 4A 35 13 C4 71 DA 2D 40 F8 56 84 20 4F 4D 56 C8 F3 AE 89 F0 6B F6 73 88 0A 70 4D 0B DC 92 8E A1 0A 93 47 7B 53 26 C4 26 A9 20 DE 6E 3B 9D 74 B6 BC 9B 30 7F 94 48 6E 11 77 | Переданный HMAC:  14 67 16 52 12 4F BF 4A 35 13 C4 71 DA 2D 40 F8 56 84 20 4F 4D 56 C8 F3 AE 89 F0 6B F6 73 88 0A 70 4D 0B DC 92 8E A1 0A 93 47 7B 53 26 C4 26 A9 20 DE 6E 3B 9D 74 B6 BC 9B 30 7F 94 48 6E 11 77 |
| Вычисленное получателем значение:  14 67 16 52 12 4F BF 4A 35 13 C4 71 DA 2D 40 F8 56 84 20 4F 4D 56 C8 F3 AE 89 F0 6B F6 73 88 0A 70 4D 0B DC 92 8E A1 0A 93 47 7B 53 26 C4 26 A9 20 DE 6E 3B 9D 74 B6 BC 9B 30 7F 94 48 6E 11 77 | Вычисленное получателем значение:  80 5F 8D 89 2A 27 34 18 AF 75 42 2C 61 FA 56 46 AA 9B A4 54 42 41 13 F3 06 FE 71 6B 98 84 48 2A CD EE 15 54 DC AF 41 C8 C2 3F 7D 51 C5 39 2E C2 BC F7 21 96 23 0A B6 D4 E9 AA D9 DC D4 29 17 CB |

**Атака дополнительной коллизии на хэш-функцию.**

Описание атаки в терминах парадокса «дня рождения».

Атака строится на парадоксе «дня рождения». Сам парадокс состоит в следующем:

Каково минимальное число k - студентов в аудитории, такое, что с заданной вероятностью по крайней мере два студента имеют тот же самый день рождения?

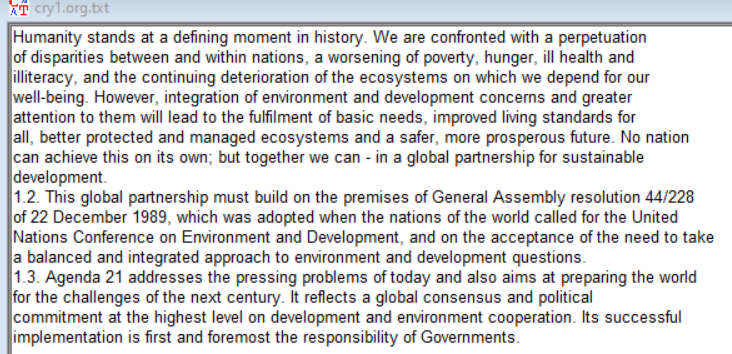
Было вычислено, что если k = 23, то с вероятностью P > 0,5 двое из студентов родились в один день (игнорируя год их рождения).

Суть метода состоит в значительном уменьшении количества передаваемых хеш-функции аргументов, необходимого для обнаружения коллизии, поскольку если хеш-функция генерирует ‑битное значение, то число случайных аргументов хеш-функции, для которого с большой вероятностью будет обнаружена хотя бы одна коллизия хеш-функции (то есть найдётся хотя бы одна пара равных хеш-кодов, полученных на разных аргументах), равно не , а только около .

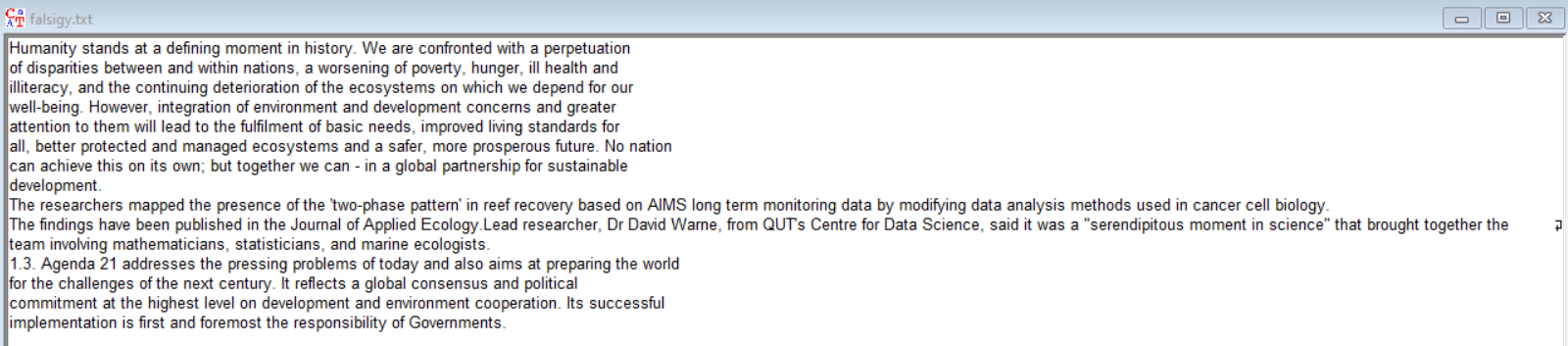
Задание.

1. Сформировать два текста на английском языке - один истинный, а другой фальсифицированный. Сохранить тексты в файлах формата \*.txt.

Сформированные тексты представлены на рис. 11 и 12 соответственно.



*Рисунок 11 – Истинный текст*



*Рисунок 12 – Фальсифицированный текст*

1. Утилитой Analysis-> Attack on the hash value... произвести модификацию сообщений для получения одинакового дайджеста. В качестве метода модификации выбрать Attach characters-> Printable characters.

Функция для исследований SHA-1.

1. Проверить, что дайджесты сообщений действительно совпадают с заданной точностью.

Проверка на совпадение представлена в Приложении А.

1. Сохранить исходные тексты, итоговые тексты и статистику атаки для отчета.

Исходные тексты представлены в пункте 1. Итоговые тексты приведены в приложении А.

1. Зафиксировать временную сложность атаки для 8, 16, 32,40, 48, … бит совпадающих частей дайджестов.

Временная сложность атаки для заданного числа бит приведена в таблице 5.

*Таблица 5 – Временная сложность атаки для n бит совпадающих частей дайджестов*

|  |  |  |
| --- | --- | --- |
| Количество значащих бит | Время атаки | Количество запусков и итоговое число шагов |
| 8 | 0 year(s), 0 day(s), 0 hour(s), 0 minute(s) und 0.00 second(s) | 3 итерации, 261 обращение к хэш-функции, 102 шага на подбор совпадений |
| 16 | 0 year(s), 0 day(s), 0 hour(s), 0 minute(s) und 0.00 second(s) | 2 итерации, 1799 обращений к хэш-функции, 715 шага на подбор совпадений |
| 32 | 0 year(s), 0 day(s), 0 hour(s), 0 minute(s) und 0.28 second(s) | 1 итерации, 136,188 обращений к хэш-функции, 51,147 шага на подбор совпадений |
| 40 | 0 year(s), 0 day(s), 0 hour(s), 0 minute(s) und 26.35 second(s) | 2 итерации, 12,796,324 обращений к хэш-функции, 4,929,928 шага на подбор совпадений |
| 48 | 0 year(s), 0 day(s), 0 hour(s), 4 minute(s) und 48.22 second(s) | 1 итерация, 135,044,174 обращений к хэш-функции, 53,313,549 шага на подбор совпадений |
| 96 |  |  |
| 104 |  |  |

**Выводы.**

1. Было проведено исследование лавинного эффекта для представленных хэш-функций. У каждой из них замена/ добавление/ удаление одного символа в прообразе приводило к изменению от 45% до 60% битов значения дайджеста (при этом среднееколичество измененных бит дайджеста во всех случаях составило 50%)**.** Кроме того, при любом изменении текста хэш-значения модифицированного файла абсолютно не совпадали с исходными. Следовательно, представленные функции обладают одним из пяти свойств идеальной криптографической хэш-функции – лавинным эффектом.
2. Было проведено исследование лавинного эффекта для хэш-функции Keccak (SHA-3). Выявлено, что данная функция им обладает – при модификации одного символа в сообщении происходило изменение в среднем 50% бит в хэше. Также, как и у хэш-функций из пункта 1 дайджест модифицированного файла абсолютно не совпадал с исходным. Сравнивания количество измененных бит дайджеста для каждого из типов изменений у всех исследуемых функций, было обнаружено, что наибольший процент получается при замене символа (исключение составил только SHA-1, у которого замена – 50%, добавление – 52%).
3. Был изучен HMAC – один из механизмов проверки целостности информации. HMAC является настройкой над MAC. Для генерации данного кода используются две константы, ipad и opad, где байт 0x36 или 0x5c повторяется *b* раз, секретный ключ (который известен как отправителю, так и получателю) и заданная хэш-функция. Само значения HMAC вычисляется по формуле:

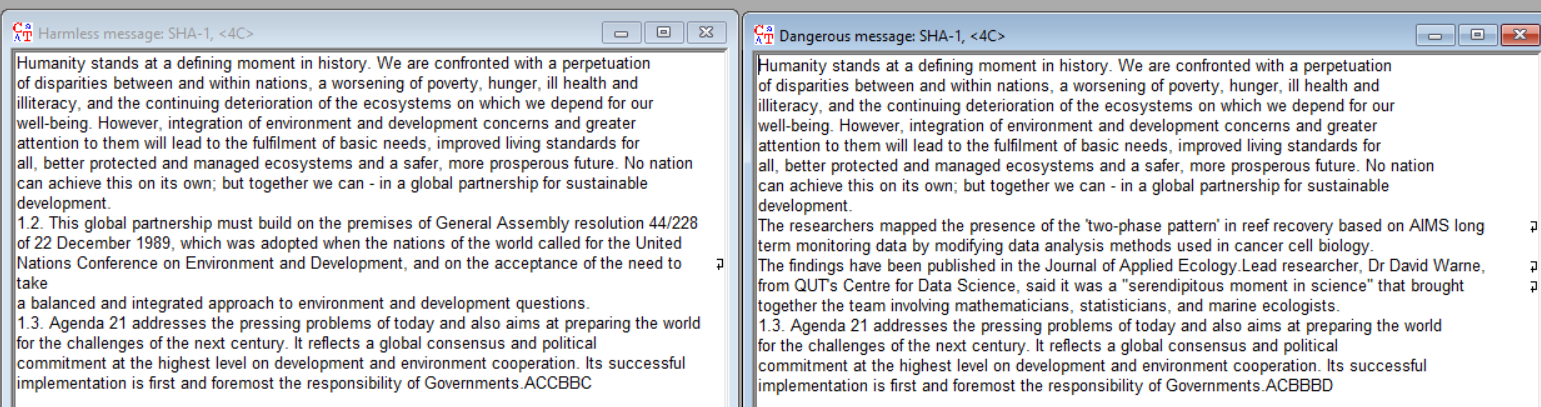
В данный момент для создания секретного ключа используется генерация на основе пароля и двух других необязательных параметров (количество итераций и начальное значение). Секретный ключ – результат повторного использования хэш-функции над заданным паролем. Количество итераций хэша значительно увеличивает количество усилий, необходимых для успешной атаки (CrypTool предлагал оптимальный вариант в 1000 итераций). Начальное значение (seed) предоставляет возможность генерации различных ключей на основе одного пароля, что затрудняет атаку вида «перебор по словарю».

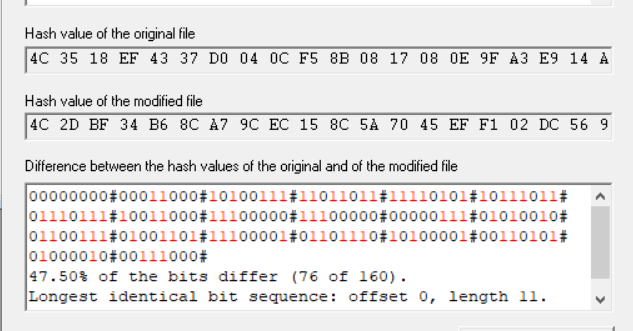
1. Была изучена атака дополнительной коллизии. Выявлено, что для двух сообщений и значимых по содержанию временная сложность атаки вычисления 48 и более бит растет экспоненциально для обеих хэш-функций (SHA-1,MD5). Истинная цель атаки – получение полностью одинакового дайджеста для двух сообщений (иначе пары ) – занимает около лет (для обеих хэш-функций). Следовательно, данные хэш-функции устойчивы к коллизиям.

ПРИЛОЖЕНИЕ А

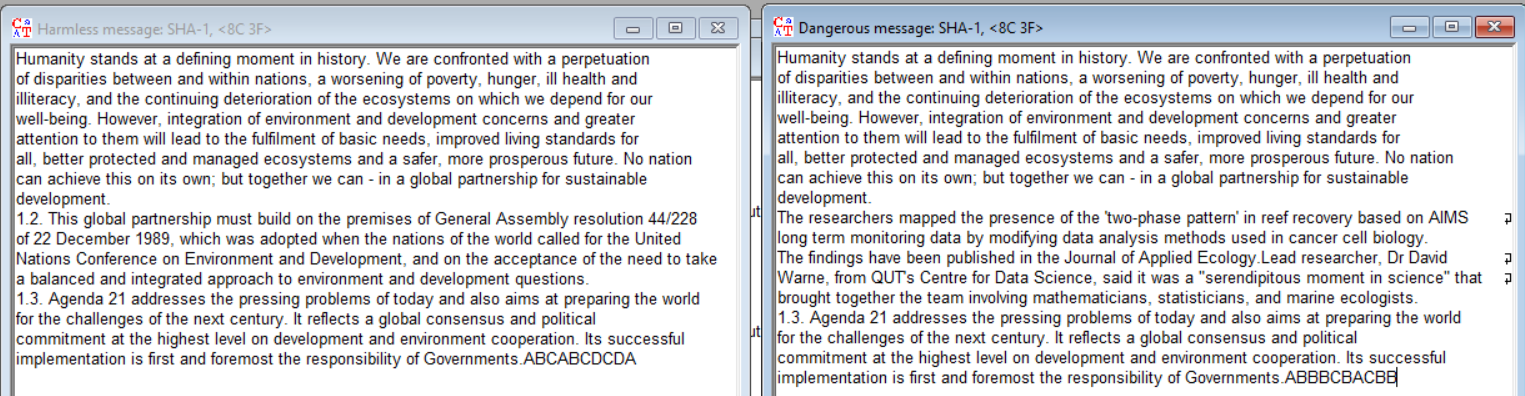
Подробные таблицы для подпункта 6.4

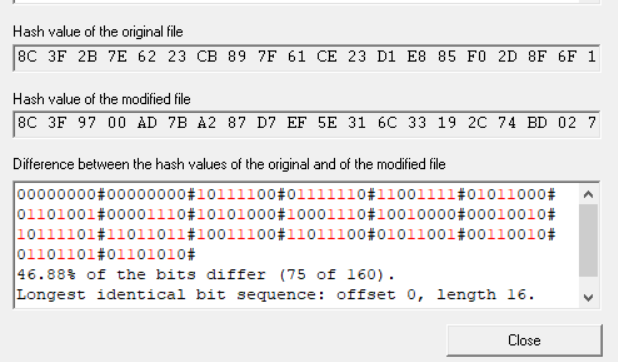
8 значащих бит:



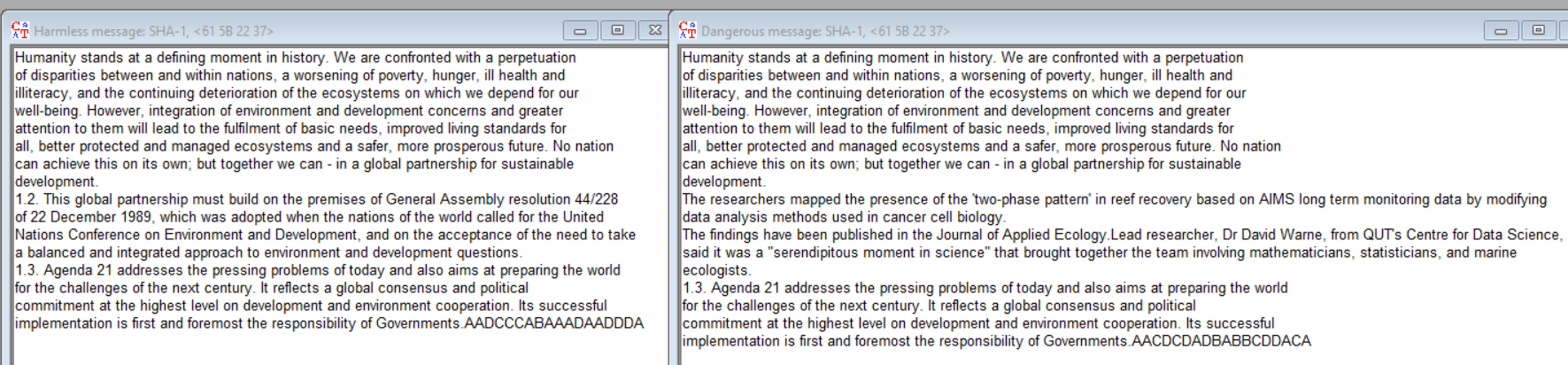


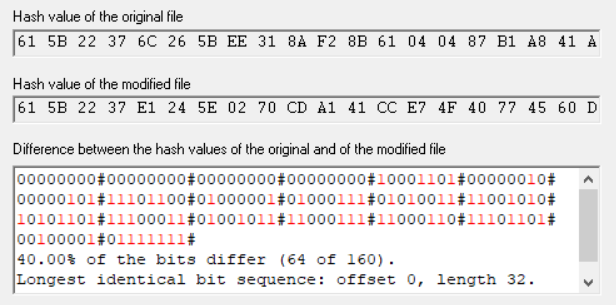
16 значащих бит:



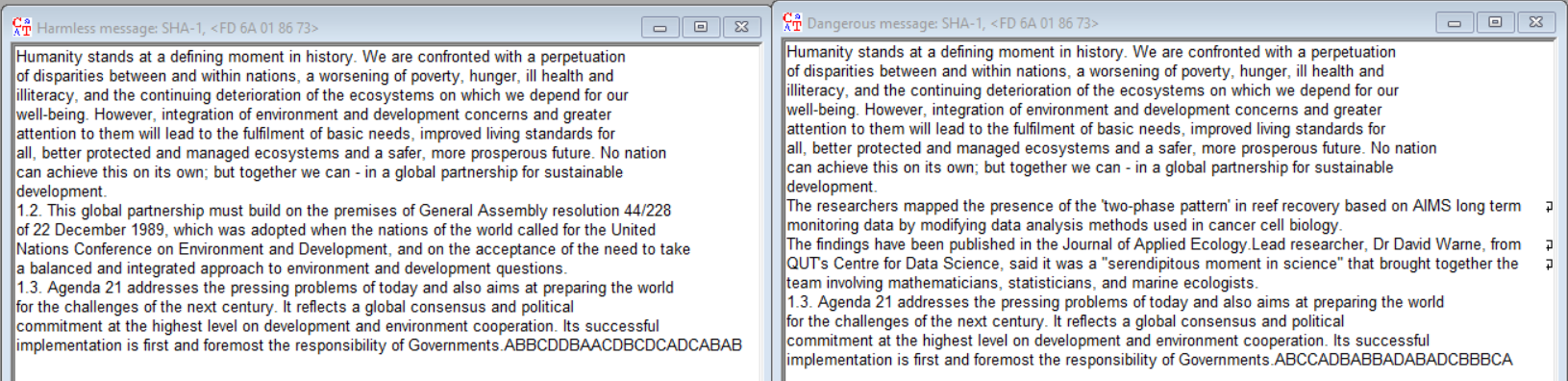


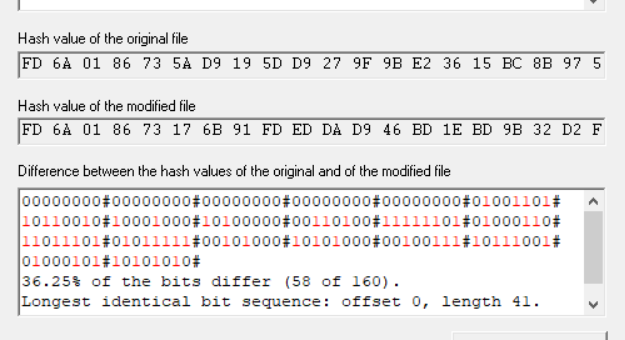
32 значащих бит:



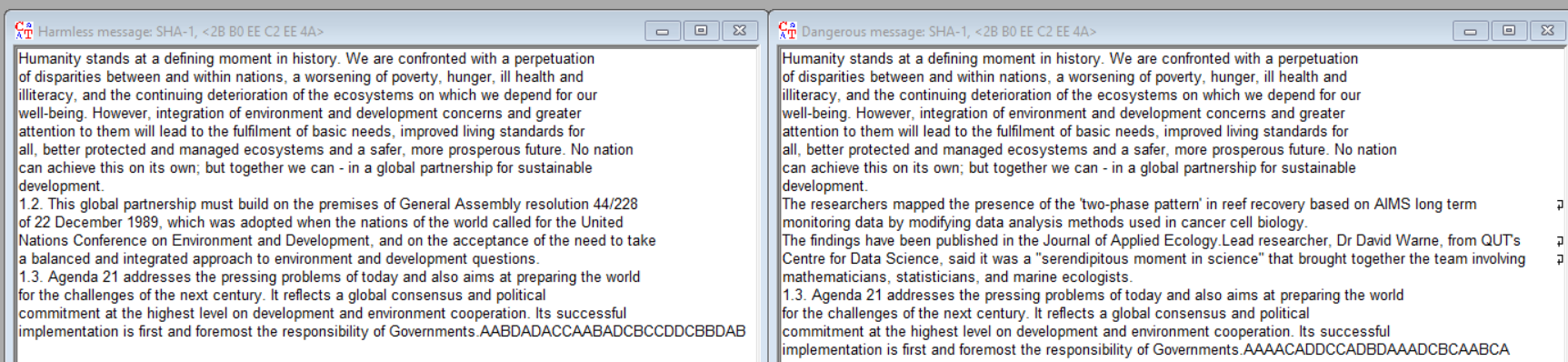


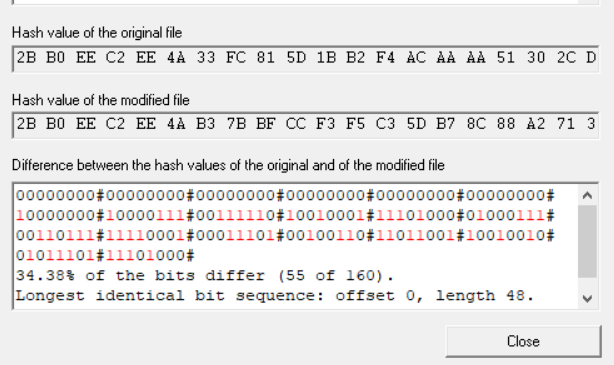
40 значащих бит:



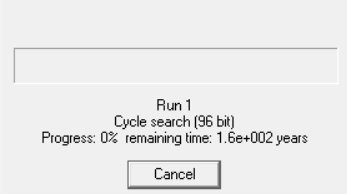


48 значащих бит:





96 значащих бит:



104 значащих бит

