Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Уфимский государственный нефтяной технический университет»

Кафедра вычислительной техники и инженерной кибернетики

Лабораторная работа №7

по дисциплине Информационная безопасность

Алгоритм шифрования RSA

Выполнил: студент гр. БПОи-16-01 К. И. Камалов

Проверил: кандидат технических наук, доцент Т.Х. Агишев

Уфа, 2019

**ЗАДАНИЕ №4**

**Атака на алгоритм шифрования RSA,**

**основанный на Китайской теореме об остатках**

*Цель работы*: изучить атаку на алгоритм шифрования RSA посредством Китайской теоремы об остатках.

*Ход работы*:

– ознакомиться с теорией, («Атака на основе Китайской теоремы об остатках»);

–Экспонента для всех вариантов *е* = 3;

– используя Китайскую теорему об остатках, получить исходный текст;

– результаты и промежуточные вычисления значений для любых трех блоков шифрованного текста оформить в виде отчета

Исходные данные:

Три пользователя имеют модули *N*1, *N*2,   
*N*3. Все пользователи используют экспоненту *e* = 3. Всем пользователям было послано некое сообщение *x*, причем пользователи получили сообщения *С*1 , С2 , *С*3

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Вариант | N1 | N2 | N3 | C1 | C2 | C3 |
| 24 | 588420697063 | 587923144219 | 588187913011 | 549837201524  41062678977  558361700271  340524813262  380989701140  72777555501  148516910596  239893043138  550275162587  504639682332  286887201361  109402692159 | 549767170235  417891652669  18105731747  561561302352  24647793868  459446623668  97023224221  315504133409  175823987146  98357398109  554098076849  2182860074 | 431364287035  324545189457  251931822912  72129334869  312886718193  268813577112  455339743965  91344488466  67230102308  491296963100  514352457528  80958415820 |

**Атака на основе Китайской теоремы об остатках.**

Системы шифрования с открытыми ключами работают сравнительно медленно. Для повышения скорости шифрования RSA на практике используют малую экспоненту зашифрования.

Вместе с тем выбор небольшой экспоненты *е* может привес­ти к негативным последствиям. Дело в том, что у нескольких корреспондентов могут оказаться одинаковые экспоненты *е*.

Пусть, например, три корреспондента имеют попарно взаимно простые модули *N*1, *N*2, *N*3 и общую экспоненту *е* = 3. Если еще один пользователь посылает им некое цирку­лярное сообщение *x*, то криптоаналитик противника может получить в свое распоряжение три шифрованных текста   
*i =* 1,2, 3. Далее он может найти решение системы сравнений, лежащее в интервале 0 < *y* < *N*1∙*N*2∙*N*3



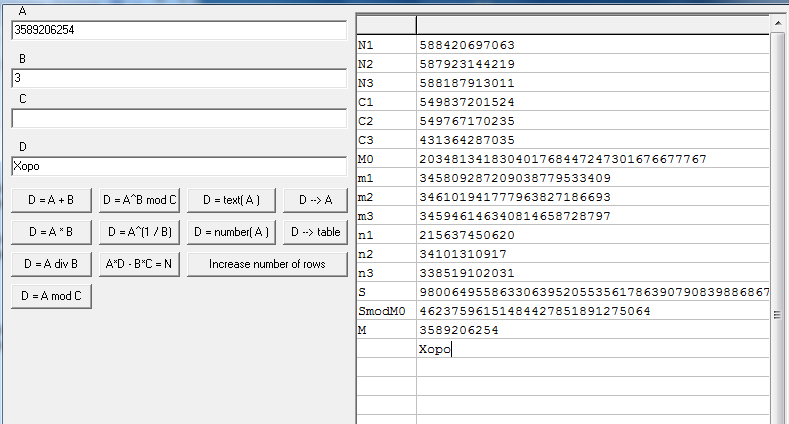
По китайской теореме об остатках такое решение единственно, а так как , то *y* = *x*3. Значение *х* можно найти, вычислив кубический корень .

Выбор малой экспоненты расшифрования *d* также нежелателен в связи с возможностью определения *d* простым перебором.

*Ход работы:*

Найдем *M*0 = *N*1∙*N*2∙*N*3. Далее находим

1. *m*1 = *N*2∙*N*3
2. *m*2 = *N*1∙*N*3
3. *m*3 = *N*1∙*N*2
4. *n*1 = *m*1-1 mod *N*1
5. *n*2 = *m*2-1 mod *N*2
6. *n*3 = *m*3-1 mod *N*3
7. *S* = С1∙*n*1∙*m*1 + *С*2∙*n*2∙*m*2 + *С*3∙*n*3∙*m*3
8. *S* mod *M*0 = 1000000000
9. *x* = (*S* mod *M*0)1/3 = Хоро



## **ЗАДАНИЕ № 5**

**Атака на алгоритм шифрования RSA**

**посредством метода Ферма**

*Цель работы:* изучить атаку на алгоритм шифрования RSA посредством метода Ферма.

*Ход работы:*

– ознакомиться с теорией в параграфе «Взлом RSA при неудачном выборе параметров криптосистемы»;

– по исходным данным, используя разложение модуля на простые числа методом Ферма, определить:

– множители модуля (*p* и *q*);

– значение функции Эйлера для данного модуля ;

– обратное значение экспоненты по модулю ;

– дешифровать зашифрованный текст, исходный текст должен быть фразой на русском языке;

– результаты и промежуточные вычисления оформить в виде отчета.

*Примечание*. Для выполнения практического задания рекомендуется использовать программу ВCalc.exe, которая находится на диске, приложенном к методическим указаниям.

|  |  |  |  |
| --- | --- | --- | --- |
| Вариант | Модуль, *N* | Экспо-нента, *e* | Блок зашифрованного текста, *C* |
| 24 | 40358026832293420582949851555688676761696698814881953082145567852270677790701800197308972113768532163014654669471670072369582748803992082057785233343991223059436508097027052487023746483576032570653936171402800244059753199908736541330736660359917302550498793306772636359956175817139979184454628016678628738264503111827049912948466857255270835537440949330111048883084562110141398334866166994558318997474175064048359054722419010659417230380256175187396800817305301565588713169244847050931595064000573130951630243604581420578033051253397318524750579966828894192332706076604398168041552148573085499792968811050267377918333 | 1355999 | 21846434284632909627705942971226364897833997421008173081736603565343819267254671854932062260327340689959708335069917418214741749558864967531675224505343223109419019661981334796530270845103654432607198476618982527787366799327190698418689377538701068783875767507450202850090256442975296418516267455736892917989495023709017207430445669175069421793626335362962662857361564455054220087431054358002977481132657054239071727528235757001539647594198238664205118330122012546210751137172986061129696174215124560966776011375251892920970288852207629545179650665241987357282611185826465237603715738775196766596629584562213642973066 |

**Ход работы:**

1. Вычисляем *n* = [sqrt(*N*)] + 1.

2. *t*1 = *n* + 1. Возводим число *t*1 в квадрат.

3. Вычисляем *w*1 = *t*1^2 – *N*.

4. Проверяем, является ли *w*1 квадратом целого числа, w1 – квадрат целого числа.

5. Вычисляем *p* = *t*1 + sqrt(*w*1) и *q* = *t*3 – sqrt(*w*3).

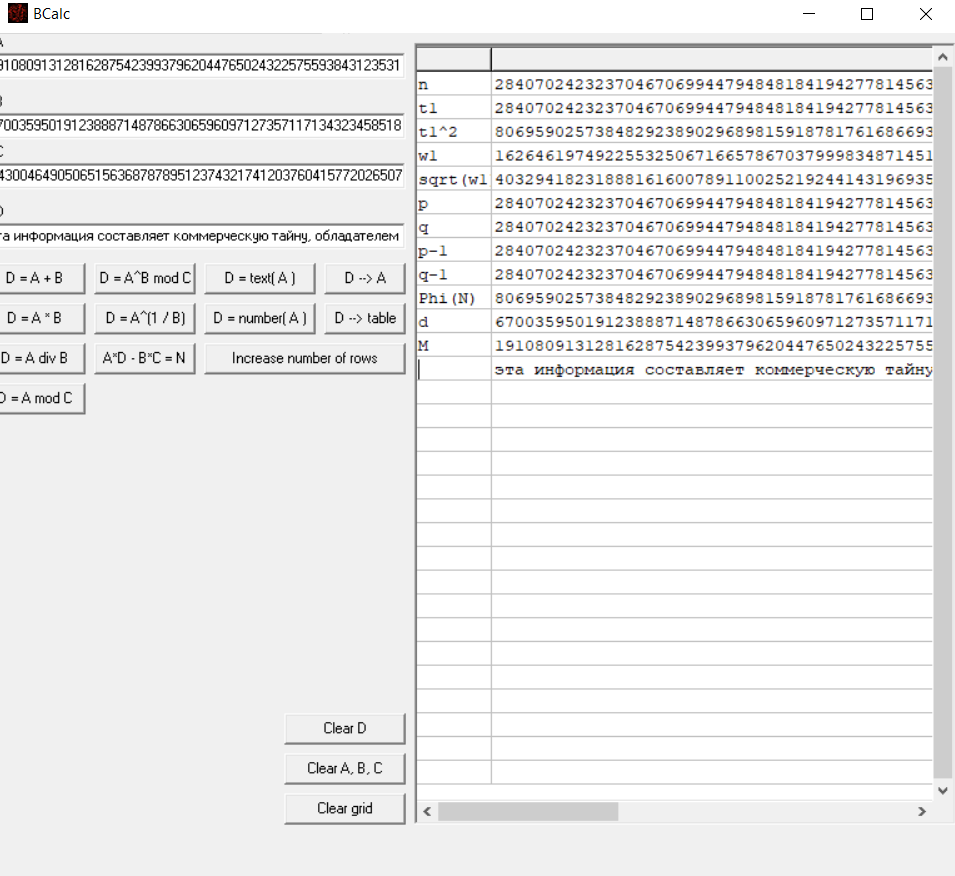
6. Вычисляем Phi(*N*) = (*p* – 1)(*q* – 1).

7. Вычисляем *d*, как обратный экспоненте *e*: *A*:= *e*; *B*:= –1; *C*:= Phi(*N*).

Нажимаем «*D* = *A*^*B* mod *C*».

8. Производим дешифрацию шифрблока *С*: *A*:= *C*; *B*:= *d*; *C*:= *N*. Нажимаем «*D* = *A*^*B* mod *C*». В поле *D* находится исходное сообщение *M*. Переводим *M* в текстовый вид.

Для этого *A*:= *M*, нажимаем «*D* = text(*A*)», в итоге получаем расшифрованный текст:



## **ЗАДАНИЕ № 6**

**Атака на алгоритм шифрования RSA**

**методом бесключевого чтения**

*Цель работы*: изучить атаку на алгоритм шифрования RSA посредством метода бесключевого чтения.

*Ход работы*:

– ознакомиться с теорией, («Бесключевое чтение»);

– по исходным данным определить значения *r* и *s* при условии, что   
*e*1∙r –*e*2∙*s* = 1. Для этого необходимо использовать расширенный алгоритм Евклида;

– используя значения *r* и *s*, получить исходный текст;

– результаты и промежуточные вычисления значений для любых трех блоков шифрованного текста оформить в виде отчета

*Примечание.* Для выполнения практического задания рекомендуется использовать программу ВCalc.exe, которая находится на диске, приложенном к методическим указаниям.

**Бесключевое чтение**

Пусть два пользователя выбрали одинаковый модуль *N* и разные экспоненты *e*1 и *e*2. Если один пользователь посылает им некое цирку­лярное сообщение *x*, то криптоаналитик противника может получить в свое распоряжение два шифрованных текста  и  В таком случае криптоаналитик может получить исходное сообщение, используя расширенный алгоритм Евклида, находим  такие, что . Отсюда получаем: 

*Пример 9*. Два пользователя применяют общий модуль *N* = 137759, но разные взаимно простые экспоненты *e*1 = 191 и *e*2 = 233. Пользователи получили шифртексты *y*1 = 60197 и *y*2 = 63656, которые содержат одно и то же сообщение. Найдем исходное сообщение методом бесключевого чтения. Так как *e*1 и *e*2 взаимно просты, то найдем такие *r* и *s*, что  С помощью расширенного алгоритма Евклида находим *r* = 61, *s* = –50. Искомое сообщение 

**Пример выполнения лабораторной работы**

**c помощью программы ВCalc**

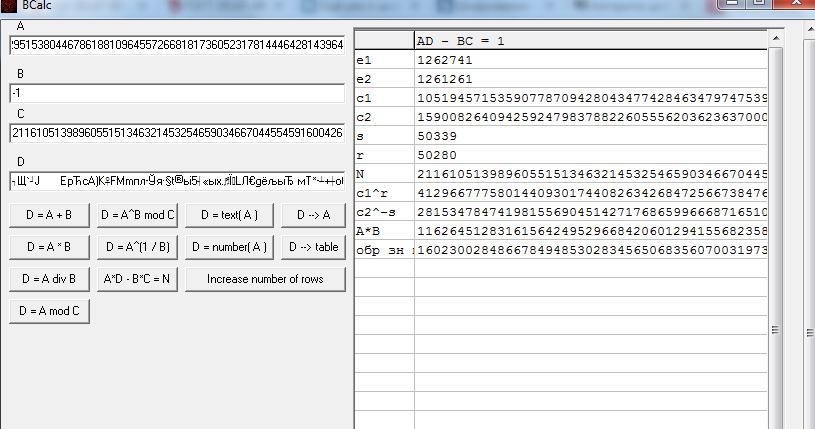
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Вариант | Модуль, *N* | Экспонента | | Блок зашифрованного текста | |
| e1 | e2 | *C*1 | *C*2 |
| 24 | 21161051398960551513463214532546590346670445545916004267495165748565101745603862439298880552912230730334777373963013322923140587388908881594979819507047860575665638614963963507148814595273060955150995937304875327692231140206965129462803585280373710788062899681052999767833035811568860409391172815018093039449757458346139366188988961050611777769714719199466730120521004297709240575754286977625902486053067862475017136045271474334297259169056177624557746463330785812569921072302740815360104065134948894688700814490513303529841108700648560688511743029542518911817962163087411300416783812914459320842461903797905549733619 | 1262741 | 1261261 | 10519457153590778709428043477428463479747539825217605241385927795187263207743266743443877333378598672119760454042486790354544308433468466159901769974560050275863977929919511496858366120844038243544189642345925291475290267533504981870903132865852666575417938367383446001769368782527289073621397121426524449984355373325448083979105176423172100309256057309620078145482786126537665697509980849865203263731605412271431637570089734169213543295915884375148855599492612261395331229518020362588819870813856959896291057276764880320007763625789950707356796938177243859825781794898700340124823452854420874754379180778727947400498 | 15900826409425924798378822605556203623637000718285220570038021402886185584460336288051579714085607370889388273357167877810500432739066795422730796569130946906092148721190938406441298208025933011151362331663394255744059262159576614819307362900313061369187548177575384259740872574783737765978485413459775347992452664455646562791113782046514567018606341323274678257723344658401685605472261460378640231409044081183003011699434439782077504165347437447577509512041803236421986228714138173816415591662779472430326510739464269037780998831231797081144579174782639716574125012295122099233285663724335421429954268742134731021154 |
|

1. Решаем уравнение *e*1 ∙ *r* – *e*2 ∙ *s* = ±1. Для этого в поле *A* помещаем значение *e*1, а в поле *B* – значение *e*2. Нажимаем кнопку «*A* ∙ *D* – *B* ∙ *C* = *N*». После этого *C* = *s* = 50339; *D* = *r* = 50280;.

2. Производим дешифрацию: *c*1 возводим в степень *r*, а *c*2 – в степень –*s* по модулю *N*.

После этого перемножаем результаты и получаем, что *m*^(*e*1 ∙ *r* – *e*2 ∙ *s*) .

Далее находим обратное значение по модулю: (*m*^ – (*e*1 ∙ *r*– *e*2 ∙ *s*) mod *N*) и преобразуем в текст.



## **ЗАДАНИЕ № 7**

**Атака на алгоритм шифрования RSA,**

**основанный на китайской теореме об остатках**

*Цель работы*: изучить атаку на алгоритм шифрования RSA посредством китайской теоремы об остатках.

*Ход работы*:

– ознакомиться с теорией, («Атака на основе Китайской теоремы об остатках»);

– Экспонента для всех вариантов *е* = 3;

– используя Китайскую теорему об остатках, получить исходный текст;

– результаты и промежуточные вычисления значений для любых трех блоков шифрованного текста оформить в виде отчета

Исходные данные:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Вариант | Блок зашифрованного текста | | | Модуль | | |
| *C*1 | *C*2 | *C*3 | *N*1 | *N*2 | *N*3 |
| 24 | 20775332638749664557004736875132834828744022215213833845624657631034453015531989435577525173575423695102299715684944268323491936171945949698351565608933635900725411263476991396071974885537580703304133968280364318634139046075903429714830308759185301474632373585730653062470979397583600690596793626001886251077020371214057196867551822311919304824375801766366222078406984929471386841631049860706485145840899965774166051359621410633664870297027965706048018985235845820108039019989880536618025605534688854727587824765333345083006098945214653002998587370327301986586821705011714766123534376240780449179297845474467119744338 | 7744112297355677701796446368194095790206104765509659936827092629264168635040093708317039781507165980736873184793958417016290087641621181024325146225186591629025040850331544786960019432230287244830375764895043429329223678383795214070455430448408745540111053074055812191584915395201330382052283681405276683246508570826576816902174319379341541425598891966179516227526467199260972116896402074445620564382612207191521376598392527528350120504050766388039952937163600637186678166621717826065976920245002496742901903547313434597437512865285709673976753880226211498644741645055246865094911913357093006560982533973322838129095 | 14033932628703216215657042755946436452648298031389136846641685388838250591590127162868202974678677802792042701570462524830270320748021665829924339449763409149297879526097463291936681511932799310621290876000339268616971848688867090839323145544713420388574433006842472585440866412305425923552016484853385221268415108450227765376408562794415958404494112178746320870936101365061812934110027748247491671242652767706384335583468317005624361389936751453809228558623529511982605670801904488363507281350633487960768795419262556489457948737573152266818962071910968619490362677476681766104706999793628007251877396001412818868837 | 22691689027995939705153841021855678960173838326289747035417950544649702808328527711494129635544086428891742752745043941655240628899789274298921666463344491456957597422287517752901848098039440060521074203549898235881206435302053769824651496044156122159880477673633785141381975811884455021755879298638661803579355399953247737768586727795546693788178294558246374565868515240262962882940323426567251916459471133519112321688561957216093408166337635260915984878510443626329312520717008682797834068907186426688770288476495381002681351589956551300047616622811582230293360553751103857290566892549327344580977185944255215312603 | 23691062990905505587562174097247100064127509698916860558365023953560511080857190988632969184328146679456354533741793925413770336691338396150198436209017517792395258658844206464560060473230471753828425406730334788914597690826846612348412096331493611840686018202943622158040566563353353365968973582644291676984839879270455870313082857242072889078760692665558870511012934289071583355929164332916023328087781276578479826922242572392977807936401924474194959652120380750578776502438939511318675767437292009009628626013043359480204513991306033254594488700802650641223527634012134927566773196657264159427397729307602277309803 | 14533683355346527292364179620551340767177176782191628021645983546111778698011023764352383766488374715736987712143952845216439705763550448879641515884938689371245338384603290677243879431862762703924453708495674270068355852797236688662510149086322158384680457124254087727686869862662842830004518278557584015229912114935730183780094896352069511127128379133983830958883594276825741723691188380278948266748017302758322266800054133088120573342120233254934497797915237200797543344161609926370699468565055137793097677985103627093303159019101465887500343942687295856334607177032666739745445828607986937964655919709980875965371 |

**Пример выполнения лабораторной работы**

**c помощью программы ВCalc**

Последовательно вычисляем:

*M*0 = *N*1 ∙ *N*2 ∙ *N*3;

*m*1 =  *N*2 ∙ *N*3;

*m*2 = *N*1 ∙ *N*3;

*m*3 = *N*1 ∙ *N*2;

*n*1 = *m*1^ (–1) mod *N*1;

*n*2 = *m*2^ (–1) mod *N*2;

*n*3 = *m3*^ (–1) mod *N*3;

*S* = *c*1 ∙ *n*1 ∙ *m*1 + *c*2 ∙ *n*2∙ *m*2 + *c*3 ∙ *n*3 ∙ *m*3 ;

*S* mod *M*0 = 67675640795094503562173784000;

*M* = (*S* mod *M*0)^(1/*e*);

text(*M*).

