CRYPTO CHALLENGE 2:

Public key cryptography: ElGamal

[Publish Date]

European Cyber Security Challenge 2019
Bucharest, Romania

Version	Name	Comments	Date
0.1	Initial version	Adrian Belmonte	01/08/2019
1.0	Final challenge	Kimmo Halunen	13/09/2019

1. Initial Write-Up

Description:

You are working in a multinational company and you have learned that the bookkeeping has been changing some key figures in reporting the revenues, taxes etc. to the authorities. In your position in the company you have been tasked with submitting the figures to the proper authorities. You would like to report the correct figures instead of the fake ones, but the numbers have been encrypted using ElGamal encryption. You have access to the public key parameters (parameters.txt) and the encrypted values (ciphertexts.txt). Each of the lines in the ciphertext file corresponds to the encryption of a single number in the reporting.

You know that the differences between the fake report and the original are the following:

1. The number on the first line has been multiplied by 1/6
2. The number on the third line has been halved
3. The number on the fourth line has been multiplied by 4
4. The number on the sixth line is the product of the first and fourth line
5. The number on the seventh line is the product of the 2nd, 3rd and 5th lines.
6. The number on the eighth line is the product of the fourth and the seventh line.

Your challenge is to generate a new encrypted file that decrypts the numbers to their original values instead of the fake ones, while conforming to the other rules in the report.

All values in the ciphertexts should be in the range from 1 to p (the ElGamal modulus given in the parameters.txt file.

2. Challenge specifications

- Category: Crypto
- Difficulty : Hard
- Expected time to solve: 4-6 hours

3. Technical specifications

Description:

Challenge Technical Specification, data to set up and access to the environment.

- Software used
  - Python and PyCrypto
- Important files to have in mind
  - ciphertexts.txt includes the encryptions of the numbers
  - parameters.txt contains the public key information for the ElGamal encryption

4. Questions and answers

1. CTF Specific questions:

Question:

What is the encryption of the correct number in the first line?

Answer:
(24534732695585738471935491722880078220834244993390317051296102360039982575415374653
96151985215797700481863880612720937336585393205826283437098566858793073778023927429
27618534895741744078781918791772448777300407083032673157692619593316510040723502239

43448245999810914039763130797420442651066384547555942848835,
360219053704479281564405177716626625561033730354165894847452895572468289553100326600
1895622314759189583650604295257108204812187537615581635274426762715583712165639755
18282895837289224539890150151953650461765253492063829892182917014016571533677137057
16401454477427559595892804206475134901890558492818087816912)

Question:

What is the encryption of the correct number in the third line?

Answer:

(93222545201007553522034519041042169078560330171885907066073791386810795522620686831
13337345682563581319041595681799535007712712432440157733912258520310110161221157650
69059871976349928283388015931546225073263013171829361994129668111124558625999067679
94841172670371505426278507574874990323844501412450819451854,
93334860439821065402243907523931593331027336783297157383661996777682048682721543200
52975582663557198385743693712496602226350126063825281563221317884937701955349199077
26796669105550857465594044693189254170141401582120640288222752496866211066210108552
08293217358477380976095071366239229466807416713764936281618)

Question:

What is the encryption of the correct number in the fourth line?

Answer:
(63248775688869304246393739089096929474008706878398836045687872201355113294318873564
83622244428603660056810147647880807622487129837478105455962203887218791708189692426
79030136622485867438546546443953941519142986368101993293047365599093567570991909625
98364780052067197015381484517497462152615630732076134991941,
31992882911595594504307167480192108644014519009563608755731647187908442759655646426
66979111859497924798938063594750792082171908531922031302412128458881603267208717622
27614812472213066418116144629263275737589124529876432672307233157833725219557620911
85560780681296602485772669155823023083886360937755408513747)

Question:

What is the encryption of the correct number in the sixth line?

Answer:
(43598757356761116202823522131555661255890788912698808864282934343894317213433738710
89167406135746956652289653827122449352563019148047735384846838756189242256845982973
11155441757594344665134771371300165009210197223145319956208787968692630968591286994
51138492240762015145589554367657771389438848216548206342444,
74338990723881754639484266446426610687774538515717818122929218844117181855103759165
53748138849307072775458075687037818183418319483348356821258349635642315400646461845

66143654164935379670158900498932068115322835718405949680285707077346089456551657946
00920526864957600664085638539586088075696420900610315021864)

Question:

What is the encryption of the correct number in the seventh line?

Answer:
(33585291075913891310800793916965312934464928023304100678409512134116230837589796966
56227663667650150687022554187238263620656861592662140953387640128079666027918022786
06496041892984880043367695292598063861378435581583169812331788559193617863992623850
61004821192730538426579815058112978035905361352248813367002,
50288651327100948482494664866234493595670225099076747790458381718184833146702724340
88166618538243681050772225633287984417664345627192247890634132753374564762319403834
67224728296378964448728181417549941991257568853032609603243455430792292370386973847
41096872494846283535751323048507443938684372250945564164335)

Question:

What is the encryption of the correct number in the eighth line?

Answer:
(56341911664355232061908417161197891099551453765176228607319557745397765684174863102
73084293870647349412622067395738761038643656844032354150911857305463117062207923813
80302352599911471710327733062328807508565518623903810923782809416607706334507171931
41834189620221311675943593756101837711785718243545804155366,
91357820937834026726424959688804742633106215328828483244600030865783657300624519470
78567347347358333510613657520232015790227079885968859532069153411374279768151331203
66159733081655272485690276573502426842248001463167271036971403094202974435615162218
02131712121331467366751608819035556731838419755564912315843)

2. Non-Flag specific:

Open Questions: Answers that may need a minimum development. (No suitable for CTF but useful for training and for a better understanding of the challenge) Multiple choice answer

Question:

Why does simple integer division of the fourth line ciphertext produce wrong answers?

Answer:

There has been reduction modulo p and the ciphertext number is not divisible by four.

Question:

What is the correct method to achieve division by four for the fourth line?

Answer:

Finding the inverse of 4 modulo p. This can be done for example with Fermat's little theorem.

5. Attack Scenario

Description:

The attacker is able to change the ciphertexts to numbers at will and also provides ciphertexts that conform to the given "chekcsums" (products).

6. Installation instructions

Description:

Setup for the organizers:

Provide the attached text files (ciphertexts.txt and parameters.txt) to the contestants in some way (email, server, IRC channel, Slack etc.)

Organisers can also utilize the private_key.txt and Solution_ElGamal.py (Python 2.7 and PyCrypto needs to be installed) to compute solutions.txt file that contains all the correct 8 ciphertexts and provides also the decryptions of these. THESE FILES ARE NOT TO BE GIVEN TO THE PLAYERS UNDER ANY CIRCUMSTANCES!

Setup for players:

Retrieve the files (ciphertexts.txt and parameters.txt) from the media provided. Open them with a text editor of your choice. Enjoy.

7. Tools needed

Description:

- General linux tools

- Text editor
- Programming language(s) with large arithmetic support (e.g. Python)

8. Artefacts Provided

Description:

List of artifacts provided with checksums.

Name	Format	Comment	Checksum (SHA256)
ciphertexts.txt	Text file	Challenge ciphertexts	33 2c 02 ae 46 1e 0e 55 ac 14 16 4d 41 bb 4a a8 26 60 0d 7f fd 68 4e 4e 75 dd 26 09 ab dd 4f d6
parameters.txt	Text file	Public key parameters	f0 6e 58 35 f1 06 e1 cc 2d 2f 91 4a 71 0f 6e 39 6a 96 56 3f cb eb 65 d3 ff 07 ad d6 e1 bd 6f 6f
private_key.txt	Text file	Private key parameters	98 b0 25 f3 31 d7 27 c5 2c 6e 74 cd f3 cb 44 e1 66 af 4b c2 d0 49 35 f5 be c8 77 44 a0 7e bb a8
Solution_ElGamal.py	Python program	Program that generates solutions.txt file that contains the right solutions. Needs Python 2.7 and PyCrypto.	f3 c6 ad 30 cd 6d de 17 3c 17 1e 26 87 93 62 88 10 3e 43 65 0d e9 db 15 a9 d0 54 41 6a b8 7a a7

9. Walkthrough (writeup)

The contestants should familiarize themselves with the ElGamal cryptosystem. The malleability and homomorphic properties of ElGamal enable the arithmetic on the encrypted values. The contestants should then change the ciphertexts in a way that counteracts the mistakes introduced in the fake report and in a way that conforms to the format regarding the report (lines 6-8).

This means the second coordinate of the first ciphertext should be multiplied by six.

The second coordinate of the third ciphertext needs to be doubled.

The second coordinate of the fourth ciphertext needs to be multiplied by the inverse of 4 modulo p.

The second coordinates of the sixth, seventh and eighth ciphertexts need to be recomputed using the values computed above.

Utilizing private_key.txt and the ciphertexts.txt solutions can be generated with the Solution_ElGamal.py file by invoking: python Solution_ElGamal on the command line.

Reading

ElGamal, Taher. "A public key cryptosystem and a signature scheme based on discrete logarithms." IEEE transactions on information theory 31.4 (1985): 469-472.

ElGamal cryptosystem in Wikipedia https://en.wikipedia.org/wiki/ElGamal_encryption

Katz, Jonathan, and Yehuda Lindell. Introduction to modern cryptography. Chapman and Hall/CRC, 2014. Chapter 10.5

PyCrypto programming package https://www.dlitz.net/software/pycrypto/api/current/