Criptarea homomorfă

*Care este soluția pentru a realiza operații pe seturi mari de date și să le păstrezi confidențialitatea în tot acest timp?*

Organizațiile stochează într-o formă criptată informații vulnerabile, precum date cu caracter personal sau date financiare. Cu ajutorul sistemelor de criptare moderne, decriptarea de către un atacator este prea costisitoare pentru a fi fezabilă. Problema apare când datele trebuie să fie manipulate. Acestea trebuie decriptate, devenind vulnerabile pentru potențialii atacatori. Criptarea homomorfă este o soluție ce permite procesarea informațiilor direct în forma criptată, fără a fi necesară decriptarea lor în prealabil. Rezultatele obținute în urma operațiilor vor fi criptate, iar conținutul poate fi decriptat numai de cel ce deține cheia secretă.

Cum funcționează criptarea homomorfă

f

Nu știe nimic despre X

X

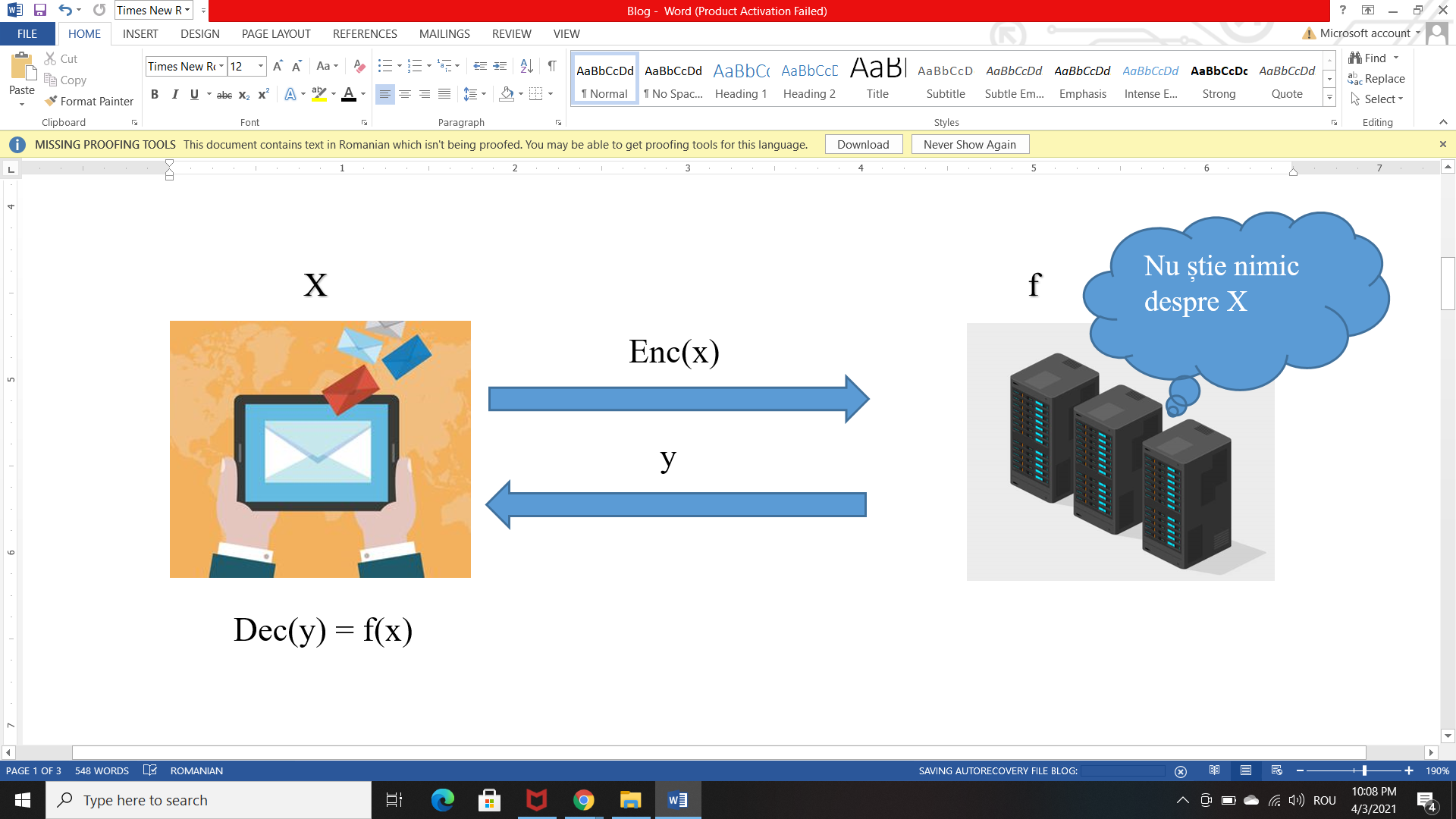


y

Enc(x)

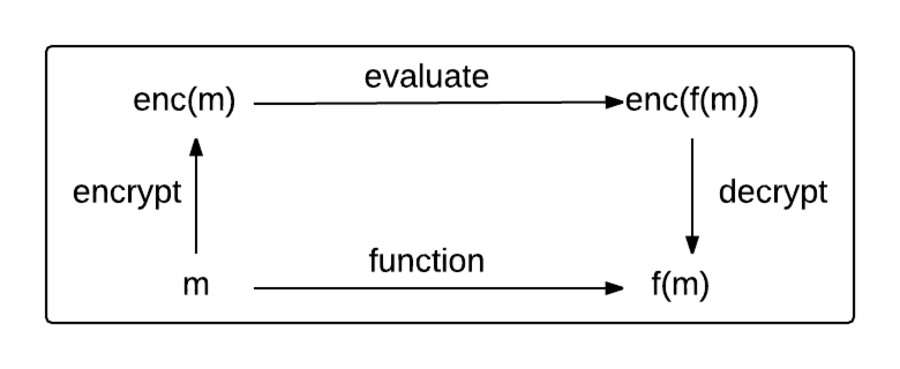
Nu știe nimic despre X

Dec(y) = f(x)



Ceea ce căutăm este o funcție de evaluare homomorfă:

Procesarea începe cu un anumit mesaj în clar m, presupunem că acesta este “John Smith”. Scopul este aplicarea unei funcții f pe textul clar, de exemplu transformarea literelor în litere mari. Acest lucru este posibil, dar este mai sigur să criptăm textul înainte pentru ca cel ce va aplica funcția să nu cunoască conținutul mesajului. Dacă mesajul criptat este 567123, iar funcția este f(x) = x + 134, atunci outputul va fi 567257, care se va decripta în “JOHN SMITH”.



Tipuri de criptare homomorfă

1. **Partially homomorphic encryption (PHE)** – permite aplicarea unei singure operații matematice (ex. adunarea) de un număr nelimitat de ori pe textul criptat.
2. **Somewhat homomorphic encryption (SHE)** – permite aplicare numai a anumitor operații (ex. fie înmulțire, fie adunare) de un număr limitat de ori pe textul criptat.
3. **Fully homomorphic encryption (FHE)** – se află încă în faza de dezvoltare și ar permite orice tip de operație de un număr nelimitat de ori, fără a fi necesară accesarea cheii de decriptare.

Probleme pe care le-ar rezolva criptarea homomorfă

* **Securizarea datelor stocate în Cloud** – utilizând criptarea homomorfă, datele ar fi stocate în Cloud în siguranță, permițând posibilitatea de a modifica și căuta informațiile criptate. Într-un mediu ideal, numai utilizatorul ce deține datele ar putea să decripteze informația inițială și cea rezultată în urma transformărilor homomorfe.
* **Analiza datelor în industrii reglementate** – criptarea homomorfă ar putea permite criptarea și externalizarea datelor în medii Cloud cu scopul cercetarii, protejând în același timp confidențialitatea datelor**.** De exemplu, se pot obține analize predictive folosind Machine Learning din dosarele de caz ale pacienților ce pot ajuta medicii să identifice tratamente pentru boli. Folosind criptarea homomorfă, datele pacienților ar rămâne confidențiale, iar algoritmii ar fi adaptați pentru a lucra pe datele criptate și ar produce rezultatul în formă criptată.

Implementari ale criptării homomorfice totale

Microsoft a creat [SEAL(Simple Encrypted Arithmetic Library)](https://www.microsoft.com/en-us/research/project/microsoft-seal/), un set de biblioteci ce permit operații direct pe datele criptate. De asemenea, Google și-a anunțat sprijinul în dezvoltarea criptării homomorfe și a realizat un tool criptografic open-source, [Private Join and Compute](https://github.com/google/private-join-and-compute).

În 2016 IBM a lansat prima versiune a [*Helib C++ library*](https://github.com/homenc/HElib), dar s-a raportat că viteza de rulare era de *100 de miliarde de ori* mai mare decât operațiile pe textul clar. Ulterior, IBM a venit cu o versiune care este de 75 de ori mai rapidă, însă este în continuare prea lentă în comparație cu operațiile directe pe textul clar.

Bibliografie:

* <https://www.thesslstore.com/blog/what-is-homomorphic-encryption/>
* <https://www.csoonline.com/article/3572381/homomorphic-encryption-deriving-analytics-and-insights-from-encrypted-data.html>
* <https://www.venafi.com/blog/homomorphic-encryption-what-it-and-how-it-used>
* <https://brilliant.org/wiki/homomorphic-encryption/#:~:text=Homomorphic%20encryption%20is%20a%20cryptographic,to%20obtain%20the%20desired%20output>.