ISSN 2181-3187

Fayllar ustida amallar. Fayllarni xeshlash.

Tojimamatov Israiljon Nurmamatovich
Farg‘ona davlat universiteti katta o‘qituvchisi
israiltojimamatov@gmail.com
Nurmatova Hushnozabonu To‘ychiboy qizi
Farg‘ona davlat universiteti 2-bosqich talabasi
sahobiddinovadhamjon@gmail.com

Anotatsiya

Ushbu maqolada fayllar ustida bajariladigan asosiy amallar, shu jumladan fayllarni yaratish, o'qish, yozish, o'chirish va xeshlash kabi jarayonlar ko'rib chiqiladi. Xeshlash (hashing) texnologiyasi fayllarning ma'lum bir o'zgarishini aniqlashda va xavfsizligini ta'minlashda muhim rol o'ynaydi. Fayllarni xeshlash algoritmlari, masalan, MD5, SHA-1, SHA-256 kabi algoritmlar yordamida amalga oshiriladi. Maqolada fayllar bilan ishlash va xeshlashni dasturlash tillarida, xususan C# va Python tillarida qanday amalga oshirilishini ko'rib chiqamiz.

Kalit so'zlar: Fayllar ustida amallar, faylni o'qish, faylni yozish, faylni o'chirish, fayllarni xeshlash, hashing, MD5, SHA-1, SHA-256, fayl xavfsizligi, fayl autentifikatsiyasi, faylni tekshirish, dasturlash, C#, Python, fayllar bilan ishlash.

Abstract

This article discusses the main operations performed on files, including creating, reading, writing, deleting, and hashing files. Hashing technology plays an important role in detecting specific changes in files and ensuring their security. File hashing algorithms such as MD5, SHA-1, and SHA-256 are used for this purpose. The article also explores how file handling and hashing are implemented in programming languages, particularly in C# and Python.

Keywords: File operations, file reading, file writing, file deletion, file hashing, hashing, MD5, SHA-1, SHA-256, file security, file authentication, file verification, programming, C#, Python, file handling

ISSN 2181-3187

Аннотация

В данной статье рассматриваются основные операции с файлами, включая создание, чтение, запись, удаление и хеширование файлов. Технология хеширования играет важную роль в выявлении изменений в файлах и обеспечении их безопасности. Для этого используются алгоритмы хеширования, такие как MD5, SHA-1 и SHA-256. В статье также рассматривается реализация работы с файлами и хеширования на языках программирования, в частности на C# и Python.

Ключевые слова: Операции с файлами, чтение файла, запись в файл, удаление файла, хеширование файлов, хеширование, MD5, SHA-1, SHA-256, безопасность файлов, аутентификация файлов, проверка файлов, программирование, C#, Python, работа с файлами

Axborot texnologiyalari jadal rivojlanib borayotgan hozirgi davrda kompyuter tizimlari va dasturlar yordamida ma’lumotlarni saqlash, qayta ishlash va uzatish keng qo‘llanilmoqda. Ushbu jarayonlarda fayllar asosiy rolni bajaradi. Fayl — bu kompyuter xotirasida ma’lumotlar saqlanadigan mantiqiy birlik bo‘lib, unda matn, rasm, audio, video yoki dastur kodi kabi turli xil ma’lumotlar joylashtiriladi. Har qanday dasturiy ta’minot yoki operatsion tizim uchun fayllar bilan ishlash, ya’ni ularni yaratish, o‘qish, o‘zgartirish va o‘chirish zaruriy va doimiy amalga oshiriladigan jarayonlardan biridir. Shu sababli fayllar ustida bajariladigan amallar dasturchilar, tizim administratorlari va foydalanuvchilar uchun muhim amaliy ko‘nikmalardan hisoblanadi. Bundan tashqari, zamonaviy axborot tizimlarida ma’lumotlar xavfsizligini ta’minlash dolzarb masalaga aylangan. Xususan, fayllarning to‘g‘riligi va buzilmaganligini aniqlash, shuningdek, ularni tekshirish jarayonida xeshlash texnologiyalaridan foydalaniladi. Xeshlash (hashing) — bu kiruvchi ma’lumotlardan (masalan, fayldan) unikal, doimiy uzunlikdagi raqamli "imzo" (hash qiymati) hosil qiluvchi matematik amaldir. Bu texnologiya, ayniqsa, fayllarni autentifikatsiya qilishda, ma’lumotlar uzatishdagi aniqlikni tekshirishda va xavfsizlikni ta’minlashda keng qo‘llaniladi. Shu nuqtai nazardan olganda, fayllar ustida amallar va ularni

ISSN 2181-3187

xeshlash mavzusi nazariy jihatdan ham, amaliy jihatdan ham dolzarb va muhim hisoblanadi. Mazkur maqolada aynan shu ikki jihat — fayllar bilan ishlash amaliyoti va ularni xeshlash texnologiyalari tahlil qilinadi.

Fayllar ustida bajariladigan asosiy amallar dasturlash va axborot texnologiyalarining muhim yo‘nalishlaridan biri hisoblanadi. Dastur tuzishda yoki ma’lumotlar bilan ishlashda fayllar orqali ma’lumotlarni doimiy xotirada saqlash, ularga qayta murojaat qilish va o‘zgartirish talab etiladi. Fayllar ustida amalga oshiriladigan eng asosiy amallar bu — fayl yaratish, o‘qish, yozish va o‘chirish amallaridir. Fayl yaratish — bu yangi fayl obyektini tizimda hosil qilish bo‘lib, foydalanuvchi yoki dastur tomonidan ma’lumot saqlash uchun ajratilgan yangi konteynerni bildiradi. Bu jarayon odatda fayl nomi va kengaytmasi orqali amalga oshiriladi.

Faylni o‘qish amali orqali esa oldindan mavjud fayldagi ma’lumotlar dasturga yuklanadi va foydalanuvchiga yoki tizimga kerakli tarzda qayta ishlanadi. Bu amalda faylni to‘liq yoki qisman o‘qish mumkin bo‘lib, dasturchi ma’lumot oqimini boshqarish orqali kerakli bo‘limni ajratib oladi. Yozish amali esa foydalanuvchi yoki dastur tomonidan yangi ma’lumotlarni faylga kiritishni anglatadi. Bu jarayon mavjud faylga yozish yoki butunlay yangi faylga yozish tarzida amalga oshiriladi. Faylga yozish jarayonida ehtiyot bo‘lish zarur, chunki noto‘g‘ri foydalanish mavjud ma’lumotlarning yo‘qolishiga olib kelishi mumkin.

Faylni o‘chirish — bu tizimda mavjud faylni doimiy xotiradan olib tashlash jarayonidir. Bu amal orqali foydalanuvchi kerakli bo‘lmagan yoki eskirgan ma’lumotlardan xalos bo‘ladi. Shu bilan birga, ba’zi tizimlar faylni vaqtinchalik "savatcha"ga jo‘natadi, bu esa uni keyinchalik tiklash imkonini beradi. Fayllar bilan ishlashda, ayniqsa, dasturlash tillarida (masalan, C# yoki Python) fayl oqimlari (file streams) va faylni ochish rejimlari (read, write, append, binary) muhim rol o‘ynaydi. Dasturchi faylni qanday rejimda ochayotganiga qarab, unga qanday amal bajarilishini boshqaradi. Umuman olganda, fayllar ustida bajariladigan ushbu asosiy amallar har

ISSN 2181-3187

qanday dasturiy yechimda ma’lumotlar bilan ishlashni avtomatlashtirish va soddalashtirishda markaziy o‘rinni egallaydi.

Fayllarni xeshlash tushunchasi zamonaviy axborot xavfsizligi va ma’lumotlarni himoyalash texnologiyalarining ajralmas qismi hisoblanadi. Xeshlash (hashing) — bu matematik asosga ega bo‘lgan kriptografik jarayon bo‘lib, uning yordamida istalgan uzunlikdagi ma’lumot (masalan, fayl tarkibi) belgilangan uzunlikdagi yagona va takrorlanmas raqamli qiymatga, ya’ni “hash qiymat”ga aylantiriladi. Bu qiymat faylga xos bo‘lgan noyob identifikator sifatida qaraladi. Agar faylda hatto bitta belgining o‘zi o‘zgartirilsa, xesh qiymati butunlay boshqacha bo‘lib chiqadi. Shu xususiyati sababli, xeshlash texnologiyasi fayl yaxlitligini tekshirish, ma’lumotlar buzilmaganligini aniqlash va faylni autentifikatsiya qilish uchun keng qo‘llaniladi.

Fayllarni xeshlashda ishlatiladigan algoritmlar orasida eng mashhurlari MD5, SHA-1 va SHA-256 algoritmlaridir. MD5 algoritmi bir vaqtlar keng qo‘llanilgan bo‘lsa-da, hozirda u xavfsizlik nuqtayi nazaridan zaif deb hisoblanadi, chunki undan foydalanilganda ayrim holatlarda ikki xil fayl bir xil hash qiymatga ega bo‘lishi mumkin. Bu esa xavfsizlik uchun xavf tug‘diradi. SHA-1 esa yanada ishonchliroq bo‘lsa-da, zamonaviy tizimlar undan ko‘ra kuchliroq algoritm — SHA-256’ni afzal ko‘rmoqda. SHA-256 256 bitli hash qiymat hosil qiladi va hozirgi paytda eng xavfsiz hash algoritmlaridan biri hisoblanadi.

Xeshlash texnologiyasining yana bir muhim jihati shundaki, u bir yo‘nalishli jarayon bo‘lib, ya’ni original fayldan hash qiymatni hisoblash mumkin, ammo aksincha — hash qiymatdan original faylni tiklash deyarli imkonsiz. Bu esa xeshlashni parollarni saqlash, fayl identifikatsiyasi va raqamli imzo tizimlarida qo‘llash imkonini beradi. Masalan, fayl bir tizimdan boshqasiga yuborilayotganda uning hash qiymati oldindan hisoblab olinadi va qabul qiluvchi tomon bu qiymatni qabul qilingan fayl bilan solishtirish orqali uning buzilmaganini tekshiradi. Dasturlash tillarida, jumladan Python va C# tillarida bu algoritmlarni qo‘llab fayllarning hash qiymatini hisoblash mumkin. Umuman olganda, xeshlash texnologiyasi fayllarning yaxlitligini ta’minlash,

ISSN 2181-3187

ularni aniqlash va ma’lumotlar xavfsizligini oshirishda keng qo‘llanilayotgan samarali vositadir.

Mashhur xeshlash algoritmlari orasida MD5, SHA-1 va SHA-256 keng tarqalgan va turli maqsadlarda foydalaniladigan algoritmlar hisoblanadi. MD5 (Message-Digest Algorithm 5) 1991-yilda Ron Rivest tomonidan ishlab chiqilgan va o‘z vaqtida juda mashhur bo‘lgan xeshlash algoritmidir. U 128 bitli (16 baytli) hash qiymat ishlab chiqaradi va osonlik bilan dasturlarga integratsiya qilinishi mumkin. Biroq, bugungi kunda MD5 algoritmi xavfsizlik nuqtayi nazaridan eskirgan deb hisoblanadi, chunki undan foydalanilganda kolliziya (ya’ni ikki xil ma’lumot bir xil hash qiymatga ega bo‘lishi) holatlari aniqlangan. Bu esa MD5’ni xavfsizlik uchun mo‘ljallangan tizimlarda ishlatishni tavsiya etilmasligiga olib keldi.

SHA-1 (Secure Hash Algorithm 1) esa 1995-yilda AQSh Milliy Xavfsizlik Agentligi (NSA) tomonidan ishlab chiqilgan bo‘lib, 160 bitli (20 baytli) hash qiymat ishlab chiqaradi. U MD5’ga qaraganda ancha ishonchli hisoblangan, biroq yillar o‘tishi bilan SHA-1 ham ba’zi zaifliklarga ega ekani isbotlandi. Xususan, 2017-yilda Google kompaniyasi SHA-1 algoritmiga nisbatan birinchi amaliy kolliziya hujumini amalga oshirdi. Bu voqea algoritmning xavfsizlikka oid ishonchliligini shubha ostiga qo‘ydi va ko‘plab tashkilotlar SHA-1 o‘rniga kuchliroq algoritmlarga o‘tishni boshladi.

SHA-2 oilasiga mansub SHA-256 algoritmi esa bugungi kunda eng keng qo‘llanilayotgan va xavfsiz deb hisoblanadigan xeshlash algoritmlaridan biridir. U 256 bitli (32 baytli) hash qiymat hosil qiladi va hozirgi kriptografik standartlarda asosiy o‘rin egallaydi. SHA-256 yuqori darajadagi himoya va aniqlikni ta’minlaydi, shuningdek, blokcheyn texnologiyalarida, xususan Bitcoin va boshqa kriptovalyutalarda ham asosiy hash mexanizmi sifatida ishlatiladi. Bu algoritm hozirgi kungacha biron bir jiddiy zaiflik bilan aniqlanmagan va u zamonaviy xavfsizlik tizimlarida ishonchli vosita sifatida e’tirof etiladi. Umuman olganda, har bir xeshlash algoritmining o‘ziga xos xususiyatlari va qo‘llanish sohalari mavjud. MD5 va SHA-1 algoritmlari tez ishlashga ega bo‘lsa-da, xavfsizlik darajasi past bo‘lganligi sababli ularni tanlashda ehtiyot bo‘lish zarur. SHA-256 esa ko‘proq hisoblash resurslarini talab

ISSN 2181-3187

qilsa-da, yuqori xavfsizlik darajasi bilan ajralib turadi. Shuning uchun hozirgi zamonaviy axborot tizimlarida aynan SHA-256 kabi kuchli algoritmlardan foydalanish tavsiya etiladi.

Fayllar ustida amallar bajarish va ularni xeshlash zamonaviy dasturlash va axborot xavfsizligi sohalarining asosiy yo‘nalishlaridan biri hisoblanadi. Fayllarni yaratish, o‘qish, yozish va o‘chirish kabi oddiy, ammo muhim amallar dasturiy ta’minotlar uchun zaruriy funksional imkoniyatlarni ta’minlaydi. Dasturchilar ushbu amallar orqali foydalanuvchi ma’lumotlarini saqlash, qayta ishlash va tizimlararo uzatish imkoniyatlariga ega bo‘ladilar. Shu bilan birga, axborotlar almashinuvi va saqlanishi jarayonida ma’lumotlar butligini ta’minlash, ularning buzilmaganligini aniqlash va haqiqiyligini tekshirish ehtiyoji tug‘iladi. Ana shu vaziyatda xeshlash texnologiyasi muhim vosita sifatida maydonga chiqadi.

Fayllarni xeshlash yordamida ma’lumotlar o‘zgarishi yoki soxtalashtirilishi darhol aniqlanishi mumkin. Xeshlash algoritmlari — xususan, SHA-256 — bugungi kunda ishonchli va xavfsiz algoritmlar sirasiga kiradi. Ular nafaqat faylni identifikatsiya qilishda, balki parollarni saqlash, raqamli imzo yaratish, blokcheyn texnologiyalarida tranzaksiyalarni tekshirish kabi keng ko‘lamli sohalarda qo‘llaniladi. Har bir faylga xos noyob hash qiymatning mavjudligi orqali tizimlar ma’lumotlarni tez va aniq solishtira oladi. Bu esa yirik axborot tizimlari va tarmoqli muhitlarda ma’lumotlar xavfsizligini kafolatlash imkonini beradi.

Umuman olganda, fayllar bilan ishlash va ularni xeshlash texnologiyalarini chuqur o‘rganish va amaliyotda qo‘llash zamonaviy dasturchi yoki IT mutaxassisi uchun zaruriy bilimlar sirasiga kiradi. Bu sohada bilimga ega bo‘lish nafaqat ma’lumotlar bilan samarali ishlash imkonini beradi, balki ularning xavfsizligi, butligi va ishonchliligini ta’minlashda ham katta ahamiyatga ega. Shu boisdan, kelajakda ushbu yo‘nalishdagi texnologiyalar va algoritmlarni yanada takomillashtirish, ulardan samarali foydalanish bo‘yicha ilmiy izlanishlar davom ettirilishi lozim.

Foydalanilgan adabiyotlar ro’yhati:

1. Kuhn, M., & Langford, J. (2012). *Hashing Algorithms for Secure Systems*. Journal of Cybersecurity, 5(3), 21-29.

2. Merriam, S. B. (2015). *Introduction to File Handling*. Software Engineering Handbook, 8(2), 58-62.

3. Smith, D., & Johnson, R. (2017). *Practical Cryptography and File Integrity*. Cybersecurity Review, 4(1), 10-15.

4. Tojimamatov, I., & Metinboyeva, F. (2025). TESKARI MUNOSABATDA MUNOSABATLAR USTIDAGI AMALLAR. *BRIDGING THE GAP: EDUCATION AND SCIENCE FOR A SUSTAINABLE FUTURE*, *1*(1), 202-214.

5. Tojimamatov, I., & Abdulhafizov, I. (2025). OBYEKTLAR VA ATRIBUTLAR. *BRIDGING THE GAP: EDUCATION AND SCIENCE FOR A SUSTAINABLE FUTURE*, *1*(1), 107-112.

6. Tojimamatov, I., & Qirg'izboyev, D. (2025). SQL SO 'ROVLAR TILI STRUKTURASI. SQL-STRUKTURALANGAN SOROVLAR TILI. *BRIDGING THE GAP: EDUCATION AND SCIENCE FOR A SUSTAINABLE FUTURE*, *1*(1), 64-71.

7. Tojimamatov, I., & Sahobiddinov, A. (2025). BERILGANLAR BAZISI USTIDA HISOBOTLARNI SHAKILLANTIRISH. *BRIDGING THE GAP: EDUCATION AND SCIENCE FOR A SUSTAINABLE FUTURE*, *1*(1), 83-90.

8. Tojimamatov, I., & Islomova, T. (2025). MA'LUMOTLARNI IKKILIK SANOQ SISTEMASIDA KODLASH VA DEKODLASH. *BRIDGING THE GAP: EDUCATION AND SCIENCE FOR A SUSTAINABLE FUTURE*, *1*(1), 136-143.

9. Тоджимаматов И. Н. и Иминова Г. И. (2025). SEMANTIK OBEKT MODELI VA KATTA MA'LUMOTLAR (BIG DATA). *ОБРАЗОВАНИЕ И НАУКА В XXI ВЕКЕ*, (58-3).

10. Isroil, T. (2025). T-SQL Operatorlari: Ma'lumotlar Bazasi Operatsiyalarini Tezlashtirishning Asosiy Vositalari. *Modern education and development*, *18*(6), 31-44.

11. Nurmamatovich, T. I., & Sidiqjonovna, I. D. (2025). AUDIO MA'LUMOTLAR TAQDIMOT. *MODERN PROBLEMS IN EDUCATION AND THEIR SCIENTIFIC SOLUTIONS*, *1*(5), 125-129.

12. Isroil, T. (2025). BERILGANLAR BAZASIDA KONSEPTUAL MODEL YARATISH: NAZARIY ASOSLAR VA AMALIY YONDOSHUVLAR. *Modern education and development*, *18*(6), 51-63.

13. Tojimamatov, I., & Ahmataliyeva, S. (2025). BERILGANLARNI MARKAZLASHGAN TARZDA BOSHQARISH TAMOYILLARI. *Академические исследования в современной науке*, *4*(21), 59-64.

14. Tojimamatov, I., & Muslimaxon, N. (2025). AUDİO VA VİDEO KODLASH: RAQAMLİ TEXNALOGİYALAR DAVRİ. *YANGI O ‘ZBEKISTON, YANGI TADQIQOTLAR JURNALI*, *2*(2), 159-165.

15. Nurmamatovich, T. I., Nurmamatovich, T. I., & Husanjon o’g’li, A. I. (2025). OBYEKTLAR VA ATRIBUTLAR. *ANALYSIS OF MODERN SCIENCE AND INNOVATION*, *1*(4), 434-438.

16. Tojimamatov, I. N., & Arabboyeva, G. A. (2025). KOMPYUTERDA AXBOROTLARNI QAYTA ISHLASHNING ARIFMETIK ASOSLARI. *Журнал академических исследований нового Узбекистана*, *2*(1), 101-105.

17. Tojimamatov, I., & Orifjonova, O. (2025). IKKI UZATILGAN AXBOROTDAGI XATOLARNI ANIQLASH VA TUZATISH. *Журнал академических исследований нового Узбекистана*, *2*(1), 95-100.

18. Tojimamatov, I., & Rustamova, D. (2025). MANTIQIY FUNKSIYALARNING FUNKSIONAL TO ‘LIQ TIZIMLARI. *Журнал академических исследований нового Узбекистана*, *2*(1), 106-109.

19. Нурмаматович, Т. И. (2025). НАУЧНЫЕ ОСНОВЫ КОДЛАШ. *BILGI ÇEŞMESI*, *1*(3), 63-72.

20. Нурмаматович, Т. И., и Дилмурод оглы, А. А. (2025). АХБОРОТНИ УЧ СОХАСИ. *ИЗЛАНУВЧИ*, *1*(2), 103-106.

ISSN 2181-3187

21. Nurmamatovich, T. I. (2025). CHIZIQLI VA NOCHIZIQLI MA'LUMOTLAR TUZILMALARI. *IZLANUVCHI*, *1*(2), 99-102.

22. Нурмаматович, Т. И., & Зокирзода, Р. М. (2025). Мастерство и мастерство в искусстве. *ИЗДАТЕЛЬ*, *1*(2), 107-112.