Ilmiy tadqiqotlar va ularning yechimlari jurnali
Aoka guvohnoma № 323759 ISSN: 3060-4818

KRIPTOGRAFIK ALGORITMLARNING TESKARI MUHANDISLIKDA QO‘LLANILISHI VA SHIFRNI BUZISH USULLARI

Tolibov Begzod Anvar o‘g‘li
Toshkent axborot texnologiyalari universiteti, Kiberxavfsizlik fakulteti
4-bosqich talabasi, Toshkent, O‘zbekiston
E-mail: begzodtolibov28@gmail.com, +998978006610
Olimov Iskandar Salimboyevich
Toshkent axborot texnologiyalari universiteti, Kriptologiya kafedrasi,
PhD, Toshkent, O‘zbekiston
https://doi.org/10.5281/zenodo.24334520

Annotatsiya. Maqolada kriptografik algoritmlarning zararli dasturlarni teskari muhandislik qilish amaliyotida qo‘llanilishi va shifrni buzish usullari tahlil qilinadi. Simmetrik va asimmetrik algoritmlarning teskari muhandislikdagi xarakterli belgilari solishtirilib, ikkilik kodda kriptografiyani aniqlashning besh usuli (entropiya, konstantalar, import va satrlar, strukturaviy tahlil, dinamik tasdiqlash) ko‘rib chiqiladi. Yetti xil namuna bilan o‘tkazilgan entropiya tajribasi shifrlangan va siqilgan ma’lumotlar entropiyasi taxminan 7,99 bit/bayt ekanini, bir baytli XOR esa entropiyani o‘zgartirmasligini ko‘rsatdi. Kriptotahlil usullari tasniflanib, zararli dasturlar tahlilida eng samarali yo‘nalish algoritmning o‘zini buzish emas, balki uning realizatsiyasidagi xatolarni va kalit materialini tahlil qilish ekani WannaCry va PayBreak misollarida asoslanadi.
Kalit so‘zlar: kriptografiya, teskari muhandislik, kriptotahlil, entropiya, XOR, RSA, AES, ransomware, YARA.

ПРИМЕНЕНИЕ КРИПТОГРАФИЧЕСКИХ АЛГОРИТМОВ В ОБРАТНОЙ РАЗРАБОТКЕ И МЕТОДЫ ВЗЛОМА ШИФРОВ

Аннотация. В статье анализируется применение криптографических алгоритмов при обратной разработке (реверс-инжиниринге) вредоносных программ и методы взлома шифров. Сопоставляются характерные признаки симметричных и асимметричных алгоритмов при реверс-инжиниринге, рассматриваются пять способов выявления криптографии в бинарном коде: энтропия, константы, импорт и строки, структурный анализ и динамическое подтверждение. Эксперимент по энтропии на семи образцах показал, что энтропия зашифрованных и сжатых данных составляет около 7,99 бит/байт, а XOR с однобайтовым ключом энтропию не меняет. Методы криптоанализа классифицированы; на примерах WannaCry и PayBreak обосновано, что наиболее эффективное направление при анализе вредоносных программ — не взлом самого алгоритма, а анализ ошибок его реализации и ключевого материала.
Ключевые слова: криптография, обратная разработка, криптоанализ, энтропия, XOR, RSA, AES, программы-вымогатели, YARA.

APPLICATION OF CRYPTOGRAPHIC ALGORITHMS IN REVERSE ENGINEERING AND CIPHER-BREAKING METHODS

Abstract. The article analyses how cryptographic algorithms are used in the reverse engineering of malware and the methods of breaking ciphers. The characteristic reverse-engineering traits of symmetric and asymmetric algorithms are compared, and five ways of identifying cryptography in binary code are examined: entropy, constants, imports and strings, structural analysis, and dynamic verification. An

entropy experiment on seven samples showed that encrypted and compressed data have an entropy of about 7.99 bits per byte, while single-byte XOR leaves entropy unchanged. Cryptanalysis methods are classified, and the WannaCry and PayBreak cases show that the most effective approach in malware analysis is not breaking the algorithm itself but analysing flaws in its implementation and its key material.

Keywords: cryptography, reverse engineering, cryptanalysis, entropy, XOR, RSA, AES, ransomware, YARA.

Kirish

O‘zbekiston Respublikasi Prezidentining 2022-yil 28-yanvardagi PF-60-son Farmoni bilan tasdiqlangan “2022–2026-yillarga mo‘ljallangan Yangi O‘zbekistonning taraqqiyot strategiyasi”da fuqarolarning shaxsiy ma’lumotlari daxlsizligini ta’minlash va kiberjinoyatchilikning oldini olish tizimini yaratish alohida vazifa sifatida belgilangan [1]. “Raqamli O‘zbekiston – 2030” strategiyasi [3] raqamli infratuzilmani rivojlantirish bilan birga axborot xavfsizligi masalalarini ham qamrab oladi, 2022-yil 15-apreldagi “Kiberxavfsizlik to‘g‘risida”gi Qonun esa shu sohadagi munosabatlarni tartibga soladi [2].

Kriptografiya axborotning maxfiyligi, butunligi va haqiqiyligini ta’minlashning asosiy vositasidir. Biroq undan nafaqat himoyachilar, balki zararli dastur mualliflari ham keng foydalanmoqda. To‘lov talab qiluvchi (ransomware) dasturlar foydalanuvchi fayllarini shifrlaydi, troyanlar boshqaruv serveri (C2) bilan aloqani shifrlangan kanal orqali olib boradi, packer va crypter vositalari esa zararli kodning o‘zini yashirish uchun shifrlashdan foydalanadi. MITRE ATT&CK bilimlar bazasida bu holatlar “Data Encrypted for Impact” (T1486), “Encrypted Channel” (T1573) va “Obfuscated Files or Information” (T1027) texnikalari sifatida alohida tavsiflangan [26].

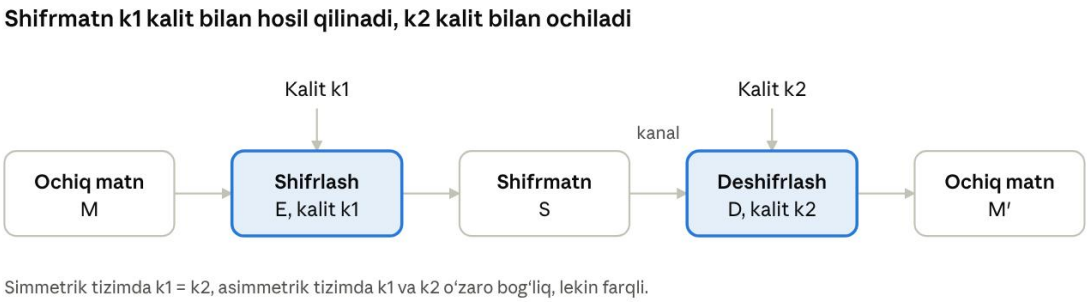
Mavzuning dolzarbligi shundaki, zararli dasturlarni tahlil qiluvchi mutaxassis teskari muhandislik jarayonida kriptografik algoritmni aniqlay olishi, uning kaliti va parametrlarini topishi hamda shifrlangan ma’lumotni tiklash imkoniyatini baholashi lozim. Topilgan kalit yoki realizatsiyadagi xatolik ko‘pincha jabrlanuvchilarning ma’lumotlarini to‘lovsiz tiklash imkonini beradi.

Ishning maqsadi – kriptografik algoritmlarning teskari muhandislikda qo‘llanilishini va shifrni buzish usullarini tahlil qilish hamda ularni zararli dasturlarni tahlil qilish amaliyoti bilan bog‘lashdir. Buning uchun simmetrik va asimmetrik algoritmlarning teskari muhandislikdagi belgilari o‘rganildi, ikkilik kodda algoritmlarni aniqlash usullari tahlil qilinib, entropiya bo‘yicha tajriba o‘tkazildi, kriptotahlil usullari tasniflandi va ransomware misolida shifrlash sxemasi tahlil qilindi. Ko‘rib chiqilgan usullar faqat himoya maqsadida va izolyatsiyalangan laboratoriya muhitida qo‘llanilishi nazarda tutiladi.

1. Kriptografik algoritmlar va teskari muhandislik asoslari

1.1. Algoritmlarning tasnifi va asosiy xususiyatlari

Shifrlash deganda ochiq axborotni shifrlangan axborotga o‘zgartirish, deshifrlash deganda esa teskari jarayon tushuniladi. Uzatiluvchi matn M kriptografik o‘zgartirish E yordamida shifrlanib, S = Ek1(M) shifrmatn hosil qilinadi, qabul qiluvchi tomonda esa M′ = Dk2(S) o‘zgartirish bajariladi [5, 6]. Simmetrik tizimlarda k1 va k2 kalitlari bir xil, asimmetrik tizimlarda esa o‘zaro bog‘liq, lekin turlicha bo‘ladi.



shifrlash sxemasi · 5 blok

1-rasm. Shifrlash jarayonining umumlashgan sxemasi

Zamonaviy kriptografiyaning muhim tamoyillaridan biri Kerkxoffs tamoyilidir (1883): tizim xavfsizligi algoritmning sirliligiga emas, faqat kalitning maxfiyligiga asoslanishi kerak [7]. Zararli dastur mualliflari bu tamoyilga ko‘pincha rioya qilmaydi: ular o‘zlari yozgan “maxfiy” algoritmlardan yoki oddiy XOR amalidan foydalanadi. Teskari muhandislik nuqtai nazaridan bu tahlilni osonlashtiradi, chunki bunday algoritmlar odatda kuchsiz bo‘ladi.

Simmetrik algoritmlar blokli va oqimli turlarga bo‘linadi. Blokli shifrlarga AES, DES va O‘z DSt 1105:2009 milliy standarti kiradi. AES 128 bitli bloklar bilan ishlaydi, kalit uzunligi 128, 192 yoki 256 bit bo‘lib, mos ravishda 10, 12 yoki 14 raunddan iborat; har bir raund SubBytes, ShiftRows, MixColumns va AddRoundKey o‘zgartirishlarini bajaradi [10, 24]. O‘z DSt 1105:2009 64 bitli blok, 256 bitli kalit va 32 raundli Feystel tarmog‘iga asoslangan [4]. Oqimli shifrlarga RC4 va ChaCha20 misol bo‘ladi: ular kalit asosida psevdotasodifiy oqim (gamma) hosil qilib, uni ochiq matn bilan XOR orqali birlashtiradi. Blokli shifrlar ECB, CBC, CTR kabi rejimlarda ishlaydi va rejim tanlovi xavfsizlikka bevosita ta’sir qiladi.

Asimmetrik algoritmlarda ochiq va maxfiy kalitlar juftligi ishlatiladi. Eng keng tarqalgan RSA 1978-yilda Rayvest, Shamir va Adleman tomonidan taklif qilingan bo‘lib, katta sonlarni tub ko‘paytuvchilarga ajratish masalasining murakkabligiga asoslanadi [15]. Kalitlar quyidagicha hosil qilinadi:

$$n = p \cdot q,\ \varphi(n) = (p-1)(q-1),\ e \cdot d \equiv 1 \pmod{\varphi(n)} \quad (1)$$

bu yerda p va q – katta tub sonlar, (e, n) – ochiq kalit, (d, n) – maxfiy kalit. Shifrlash va deshifrlash C = M^e mod n va M = C^d mod n formulalari bilan bajariladi. Elliptik egri chiziqlarga asoslangan kriptografiya (ECC) 256 bitli kalit bilan 3072 bitli RSA kalitiga teng xavfsizlik beradi, shu sababli kichik qurilmalar uchun qulay [9]. Asimmetrik algoritmlar sekin ishlagani uchun amalda gibrid sxema qo‘llanadi: katta hajmdagi ma’lumot simmetrik kalit bilan shifrlanadi, kalitning o‘zi esa asimmetrik algoritm bilan himoyalanadi. Aynan shu sxema zamonaviy ransomware dasturlarida ishlatiladi. Heshlash funksiyalari (MD5, SHA-1, SHA-256) shifrlash hisoblanmaydi, ammo kalitni paroldan hosil qilishda va API nomlarini yashirishda (API hashing) uchraydi, shuning uchun ularni ham aniqlay bilish zarur.

1-jadval. AES, RSA va ECC algoritmlarining teskari muhandislik nuqtai nazaridan solishtirilishi

Xususiyat	AES	RSA	ECC
Turi	Simmetrik, blokli	Asimmetrik	Asimmetrik
Muammo asosi	Almashtirish–o‘rin almashtirish tarmog‘i	Sonni faktorlash	Egri chiziqda diskret logarifm
Kalit	128, 192, 256 bit	2048 bit va	Odatda 256 bit

Xususiyat	AES	RSA	ECC
uzunligi		undan ortiq	
Tezlik	Yuqori	Past	O‘rtacha
Teskari muhandislikdagi belgisi	256 baytli S-box, Rcon, MixColumns	Katta sonlar arifmetikasi, modulli darajaga oshirish sikli	Nuqtalarni qo‘shish va ikkilantirish, egri chiziq parametrlari

1.2. Zararli dasturlarni tahlil qilishda kriptografiyaning o‘rni

Teskari muhandislik – dasturning manba kodi mavjud bo‘lmaganda uning tuzilishi va ishlash mantig‘ini ikkilik kod asosida tiklash jarayonidir [13, 14]. Statik tahlilda dastur bajarilmasdan uning PE sarlavhasi, import va eksport jadvallari, satrlari va bo‘limlari o‘rganiladi; dinamik tahlilda esa dastur nazorat qilinadigan muhitda (debugger yoki sandbox ichida) ishga tushirilib, xatti-harakati kuzatiladi [11]. Dizassemblerlash uchun IDA Pro [12] va NSA tomonidan ochiq kodli qilingan Ghidra [30] qo‘llanadi.



teskari muhandislik bosqichlari · 6 bosqich

2-rasm. Zararli dasturni teskari muhandislik qilish bosqichlari

Zararli dasturlarda kriptografiya to‘rt maqsadda ishlatiladi: satrlar va konfiguratsiyani yashirish (C2 manzillari, mutex va API nomlari); zararli yuklamani (payload) yashirish, ya’ni packer va crypter vositalarida asosiy kodni shifrlab, faqat xotirada ochish, bu imzoga asoslangan antivirusdan qochishga yordam beradi; tarmoq trafigini XOR, RC4, AES yoki TLS orqali himoyalash; hamda foydalanuvchi fayllarini shifrlab, to‘lov talab qilish.

Analitik uchun bu to‘rtta vazifani qo‘yadi: algoritmni aniqlash; kalit, boshlang‘ich vektor (IV) va boshqa parametrlarni topish; shifrlangan ma’lumotni deshifrlash; sxemaning zaif joylarini baholash. Muhimi, amaliyotda AES yoki RSA kabi standart algoritmning o‘zi deyarli hech qachon “buzilmaydi”. Analitik algoritmni emas, uning realizatsiyasini tahlil qiladi: kalit qayerda saqlanadi, qanday hosil qilinadi va xotirada qancha vaqt turadi. Windows muhitida kriptografik funksiyalar odatda CryptoAPI (advapi32.dll) yoki CNG (bcrypt.dll) orqali chaqiriladi [25], bunda algoritm import jadvalida ko‘rinadi. Zararli dastur OpenSSL kabi kutubxonani statik ulasa yoki algoritmni mustaqil yozsa, uni kod va konstantalar orqali aniqlashga to‘g‘ri keladi.

2. Binar kodda kriptografik algoritmlarni aniqlash usullari

Kriptografik kodni aniqlash uchun bir-birini to‘ldiruvchi bir nechta usul qo‘llanadi [11].

2.1. Entropiya tahlili

Shifrlangan yoki siqilgan ma’lumotlar tasodifiy ketma-ketlikka yaqin bo‘ladi. Baytlar ketma-ketligining Shannon entropiyasi quyidagicha hisoblanadi:

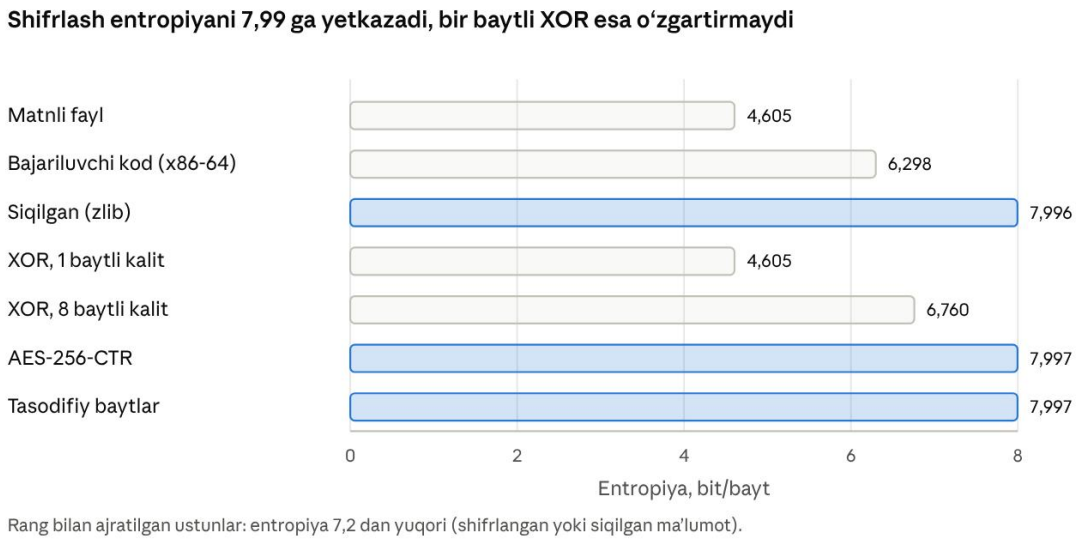
$$H = -\sum p_i \cdot \log_2 p_i, \quad i = 0, 1, \ldots, 255 \quad (2)$$

bu yerda pi – i qiymatli baytning uchrash ehtimoli. Entropiya 0 dan 8 bit/baytgacha o‘zgaradi: ochiq matn uchun taxminan 4–5, bajariluvchi kod uchun 5–6,5, shifrlangan yoki siqilgan ma’lumot uchun esa 7,2 dan yuqori bo‘ladi. Bu chegara shartli bo‘lib, tahlil vositalarida (masalan, PE bo‘limlari entropiyasini ko‘rsatishda) keng qo‘llanadi.

Usulni tekshirish uchun har biri 60 000 bayt hajmdagi yetti xil namuna bilan tajriba o‘tkazildi. Natijalar 2-jadvalda keltirilgan.

2-jadval. Turli namunalarning entropiyasi (tajriba natijalari)

Namuna	Hajmi, bayt	Entropiya, bit/bayt	Izoh
Matnli fayl	60 000	4,605	Ochiq matn
Bajariluvchi kod (x86-64)	60 000	6,298	.text bo‘limi
Siqilgan ma’lumot (zlib)	60 000	7,996	Chegaradan yuqori
XOR, 1 baytli kalit	60 000	4,605	Entropiya o‘zgarmagan
XOR, 8 baytli kalit	60 000	6,760	Oraliq qiymat
AES-256-CTR	60 000	7,997	Chegaradan yuqori
Tasodifiy baytlar	60 000	7,997	Chegaradan yuqori



namunalar entropiyasi · 7 ustun

3-rasm. Namunalarning Shannon entropiyasi

Natijalardan ikkita muhim xulosa kelib chiqadi. Birinchidan, AES bilan shifrlangan ma’lumot, siqilgan ma’lumot va tasodifiy baytlar entropiya bo‘yicha deyarli farq qilmaydi (taxminan 7,997), ya’ni entropiya faqat “bu ma’lumot shifrlangan yoki siqilgan” degan taxminni beradi, aniq algoritmni ko‘rsatmaydi. Ikkinchidan, bir baytli kalit bilan XOR qilingan matnning entropiyasi ochiq matnnikidan farq qilmaydi (4,605), chunki bu amal baytlar chastotasini faqat boshqa qiymatlarga ko‘chiradi. Shu sababli oddiy XOR bilan yashirilgan satrlarni entropiya orqali aniqlab bo‘lmaydi; ularni topish uchun ma’lum ochiq matn usuli kerak bo‘ladi (3.2-band).

2.2. Konstantalar bo‘yicha qidiruv

Ko‘pgina algoritmlar o‘ziga xos sobit konstantalarga (S-box, boshlang‘ich holat qiymatlari, raund konstantalari) ega bo‘lib, ular ikkilik kodda o‘zgarishsiz saqlanadi. Shu sababli bu usul eng tez va ishonchli usullardan biridir. Qidiruv uchun IDA Pro uchun findcrypt-yara plagini [28], YARA qoidalari [29] va Mandiant capa vositasi [27] ishlatiladi. Masalan, AES S-boxining birinchi baytlari (63 7C 77 7B F2 6B 6F C5) ketma-ketligini YARA qoidasida hex-satr sifatida yozib, dasturda AES jadvali borligini aniqlash mumkin.

3-jadval. Kriptografik algoritmlarning xarakterli konstantalari

Algoritm	Konstanta (o‘n oltilik)
AES	63 7C 77 7B F2 6B 6F C5 … (S-box); 52 09 6A D5 … (teskari S-box)
MD5	67452301 EFCDAB89 98BADCFE 10325476; D76AA478 …
SHA-1	67452301 EFCDAB89 98BADCFE 10325476 C3D2E1F0
SHA-256	6A09E667 BB67AE85 …; 428A2F98 71374491 …
TEA / XTEA	9E3779B9 (oltin kesim asosidagi δ)
Blowfish	243F6A88 85A308D3 … (π sonining kasr qismi)
ChaCha20 / Salsa20	“expand 32-byte k”
CRC-32	EDB88320 (yoki 04C11DB7); kriptografik emas, lekin tez-tez chalkashtiriladi

Shuni hisobga olish kerakki, konstantalarning topilishi algoritm ishlatilayotganini isbotlamaydi (jadval foydalanilmagan kutubxona kodida bo‘lishi mumkin), va aksincha, konstantalar obfuskatsiya yoki dinamik hisoblash orqali yashirilgan bo‘lsa, usul natija bermaydi [19].

2.3. Import va satrlar tahlili

Windows dasturlarida kriptografik funksiyalar import jadvali orqali ko‘rinadi. CryptGenRandom kalit va IV manbai bo‘lib, uning chaqiruvini kuzatish kalitni beradi; CryptGenKey va CryptDeriveKey funksiyalariga uzatilgan ALG_ID parametri algoritmni bevosita ko‘rsatadi: CALG_AES_256 (0x6610) qiymati AES-256 ga, CALG_RC4 (0x6801) esa RC4 ga mos keladi. CryptEncrypt va CryptDecrypt chaqiruvlariga breakpoint qo‘yib, ochiq va shifrlangan bufer juftligini olish mumkin. CNG da algoritm nomi satr ko‘rinishida beriladi (BCryptOpenAlgorithmProvider funksiyasida “AES”, “RSA”), rejim esa xususiyatdan o‘qiladi (masalan, ChainingModeCBC). Kalit blobi tarkibidagi “RSA1” belgisi ochiq RSA kaliti borligidan, “RSA2” esa maxfiy kalit borligidan dalolat beradi. Satrlar orasida “Microsoft Enhanced RSA and AES Cryptographic Provider” yoki “-----BEGIN PUBLIC KEY-----” kabi ifodalar ham muhim alomatdir [11, 25].

2.4. Strukturaviy va avtomatik usullar

Konstantalar bo‘lmaganda (masalan, RC4 da yoki obfuskatsiyalangan kodda) kod tuzilmasiga e’tibor beriladi: XOR, siklik siljitish (ROL/ROR) va qo‘shish amallarining ko‘p takrorlanuvchi sikllari kripto-kod belgisidir. RC4 da 256 elementli S massivi 0 dan 255 gacha to‘ldirilib, so‘ng kalit asosida elementlar o‘rni almashtiriladi; bu naqsh dizassemblerda oson tanib olinadi. Aniqlashni avtomatlashtirish bo‘yicha bir necha yo‘nalish rivojlangan. Aligot usuli bajarilish izidan sikllarni ajratib, ularning kirish–chiqish qiymatlarini ma’lum algoritmlar bilan taqqoslaydi va obfuskatsiyalangan kodda ham RC4, XTEA kabi algoritmlarni aniqlay oladi [19]. Lestringant va hammualliflarning usuli kodning ma’lumotlar oqimi grafigini (DFG) normallashtirib, mos yozuv realizatsiyalarining grafigi bilan taqqoslaydi [20]. Keyinchalik bit-aniqlikdagi simvolik bajarilishga asoslangan usul [21] hamda DFG

yondashuvini noma’lum (xususiy) algoritmlarga kengaytirgan “Where’s Crypto?” vositasi [23] taklif qilindi.

2.5. Dinamik tasdiqlash

Algoritm nomzodi aniqlangach, natija dinamik tahlilda tasdiqlanadi: debuggerda (x64dbg, OllyDbg, WinDbg) CryptEncrypt yoki BCryptEncrypt funksiyasiga breakpoint qo‘yilib, chaqiruv paytida kalit, IV va ochiq matn buferi xotiradan o‘qiladi. So‘ng shu kirish ma’lumotlari mos yozuv realizatsiyasiga (masalan, Python kutubxonasiga) uzatilib, chiqish natijasi dastur hosil qilgan shifrmatn bilan solishtiriladi. Natijalar mos kelsa, algoritm, rejim va kalit to‘g‘ri aniqlangan hisoblanadi. Shunday qilib, aniqlash ketma-ketligi quyidagicha: entropiya bo‘yicha shifrlangan hududlarni topish → konstanta va importlar bo‘yicha nomzod algoritmni tanlash → struktura tahlili bilan tasdiqlash → dinamik tekshiruv.



aniqlash ketma-ketligi · 4 qadam, 1 qaror
4-rasm. Kriptografik algoritmni aniqlash ketma-ketligi

3. Shifrni buzish (kriptotahlil) usullari

Kriptotahlil usullari hujumchi imkoniyatlariga qarab tasniflanadi: faqat shifrmatn ma’lum bo‘lgan hujum; ma’lum ochiq matn (known-plaintext) hujumi; tanlangan ochiq matn va tanlangan shifrmatn hujumlari [8, 9]. Zararli dasturlarni tahlil qilishda ko‘pincha ma’lum ochiq matn holati uchraydi, chunki analitik shifrlangan ma’lumot mazmunini taxminan biladi: masalan, shifrlangan PE fayl boshida “MZ” belgilari yoki “This program cannot be run in DOS mode” qatori bo‘lishi kerak.

3.1. Klassik shifrlar

Sezar va Vijener kabi klassik shifrlar chastota tahlili, Kasiski usuli va mos kelish indeksi yordamida buziladi: monoalfavitli almashtirishda til harflarining chastotasi saqlanib qolgani uchun shifrmatndagi eng ko‘p uchragan belgilar ochiq matndagi eng ko‘p uchraydigan harflarga mos qo‘yiladi [6, 8]. Bu usullar zamonaviy shifrlarga qarshi samarasiz.

3.2. XOR va ma’lum ochiq matn hujumi

Zararli dasturlarda eng ko‘p uchraydigan “shifr” – qisqa kalitli XOR. Shifrlash $C = P \oplus K$ ko‘rinishida bo‘lgani uchun kalit quyidagicha topiladi:

$K = C \oplus P$ (3)

Masalan, shifrlangan fayl boshidagi baytlar 17 00 bo‘lsa va fayl PE formatida bo‘lishi kerak bo‘lsa, ochiq matn “MZ” = 4D 5A ga mos keladi. Birinchi bayt uchun $K = 17 \oplus 4D = 5A$, ikkinchisi uchun $K = 00 \oplus 5A = 5A$. Ikkala qiymat mos tushgani bir baytli kalit 0x5A ekanini ko‘rsatadi, shundan so‘ng butun fayl shu kalit bilan deshifrlanadi. Kalit uzunroq bo‘lsa, avval uning uzunligi aniqlanadi (Hamming masofasi yoki mos kelish indeksi bo‘yicha), so‘ng kalitning har bir baytiga alohida bir baytli XOR usuli

qo‘llanadi [11]. Xuddi shunday, oqimli shifrlarda (RC4, CTR rejimi) bir xil kalit va IV ikki marta ishlatilsa, ikki shifrmatnning XOR i ochiq matnlarning XOR iga teng bo‘ladi:

$C1 \oplus C2 = P1 \oplus P2$ (4)

va bu gamma oqimini bilmasdan ham matnni tiklash imkonini beradi.

3.3. To‘liq qidiruv va chuqur kriptotahlil

To‘liq qidiruv (brute force) usulida barcha kalitlar sinab ko‘riladi, murakkabligi 2^k ga teng (k – kalit uzunligi, bit). 56 bitli DES kaliti 1998-yilda maxsus apparat qurilma yordamida bir necha kun ichida topilgan, 128 bitli kalit uchun esa to‘liq qidiruv hozirgi texnika bilan amaliy jihatdan mumkin emas [9]. Differensial [16] va chiziqli [17] kriptotahlil to‘liq qidiruvdan tezroq natija beradi: Matsui usuli DES ni taxminan 2^43 ta ma’lum ochiq matn bilan buzish mumkinligini ko‘rsatgan. AES esa bu usullarga chidamli qilib ishlab chiqilgan va ochiq adabiyotlarda uning to‘liq raundli variantiga qarshi amaliy hujum ma’lum emas [10].

3.4. Asimmetrik algoritmlarga hujum: faktorlash

RSA ning ishonchliligi $n = p \cdot q$ ko‘paytmani ko‘paytuvchilarga ajratishning murakkabligiga bog‘liq. Kichik misol: $p = 61$, $q = 53$ bo‘lsa, $n = 3233$, $\varphi(n) = 3120$; $e = 17$ uchun $d = 2753$ (chunki $17 \cdot 2753 = 46801 = 15 \cdot 3120 + 1$). $M = 65$ xabar $C = 65^{17} \bmod 3233 = 2790$ ga shifrlanadi. Hujumchi ochiq kalitdagi $n = 3233$ ni 2 dan boshlab bo‘luvchilarni sinab 53 va 61 ga ajratadi, $\varphi(n)$ ni hisoblaydi va d ni topadi [15]. Haqiqiy tizimlarda modul 2048 bit va undan katta bo‘lgani uchun bunday qidiruv imkonsiz: 512 bitli RSA modullari 1999-yilda, 768 bitli modul esa 2009-yilda faktorlangan, shu sababli hozir kamida 2048 bitli kalitlar tavsiya etiladi.

Yon kanal hujumlari algoritmning matematik zaifligiga emas, uning amaliy ishlashiga asoslanadi. Kocher taklif qilgan vaqt bo‘yicha hujumda shifrlash operatsiyalarining bajarilish vaqtidagi farqlardan kalit bitlari aniqlanadi [18]; quvvat iste’moli va elektromagnit nurlanish tahlili ham qo‘llanadi.

Zararli dasturlar tahlilida eng samarali yo‘nalish – realizatsiya xatolaridan foydalanish. Eng ko‘p uchraydigan xatolar: kalitning dastur ichiga qat’iy yozib qo‘yilishi (hardcoded key); psevdotasodifiy sonlar generatorining vaqtga bog‘liq boshlang‘ich qiymati tufayli kalitning taxmin qilinishi; bir xil kalit va IV ning qayta ishlatilishi; ECB rejimida bir xil bloklarning shifrmatnda ham takrorlanishi; kalitning paroldan tuzsiz (salt) oddiy heshlash orqali hosil qilinishi; kalit materialining xotiradan o‘chirilmasligi.

4-jadval. Shifrni buzish usullarining tasnifi

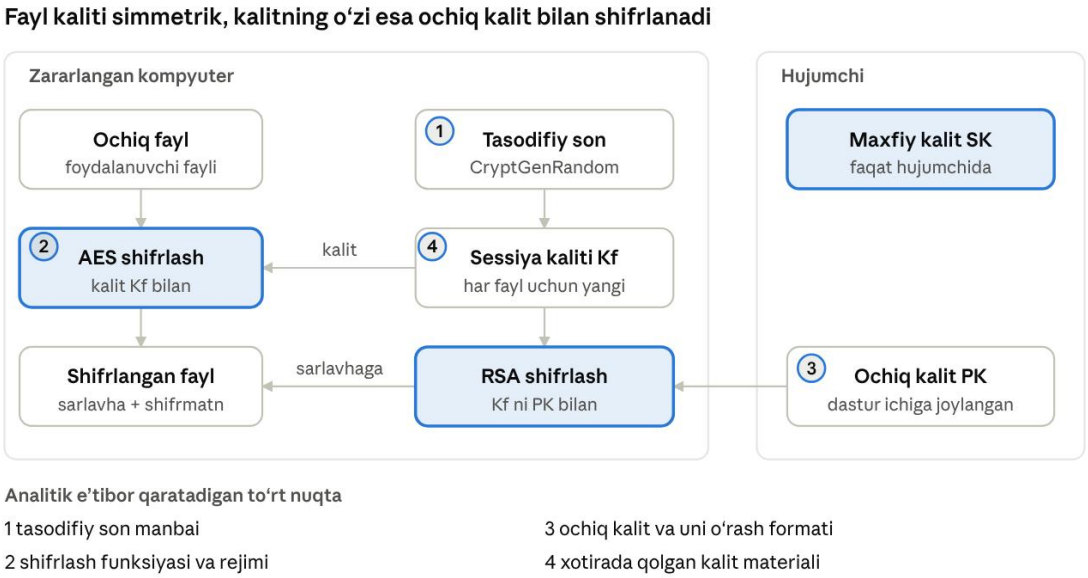
Usul	Qo‘llanish sharti	Qarshi chora
Chastota tahlili	Faqat shifrmatn; almashtirish shifri	Zamonaviy blokli va oqimli shifrlar
To‘liq qidiruv	Qisqa kalit (masalan, 56 bit)	Kamida 128 bitli kalit
Ma’lum ochiq matn (XOR)	Ochiq matnning bir qismi ma’lum	Standart shifr (AES) va tasodifiy kalit
Differensial va chiziqli kriptotahlil	Ko‘p ma’lum ochiq matn juftliklari	Yetarli raundlar soni (AES)
Modulni faktorlash	Kichik RSA moduli	Kamida 2048 bitli modul
Yon kanal hujumlari	Qurilmaga fizik yoki vaqt bo‘yicha kirish	Doimiy vaqtli realizatsiya
Realizatsiya xatolari	Xotira, kalit saqlash va IV tahlili	Xotirani tozalash, noyob IV, ishonchli CSPRNG

4. Ransomware misolida shifrlash sxemasini tahlil qilish va himoya choralari

4.1. Gibrid shifrlash sxemasi

Zamonaviy ransomware dasturlari odatda gibrid sxemada ishlaydi. Dastur har bir fayl uchun kriptografik tasodifiy sonlar generatori (masalan, CryptGenRandom) yordamida tasodifiy sessiya kaliti Kf hosil qiladi va faylni AES kabi tezkor simmetrik algoritm bilan shifrlaydi. So‘ngra bu kalit dastur ichiga joylangan hujumchining ochiq kaliti PK (odatda RSA) bilan shifrlanib, faylning sarlavhasiga yoki oxiriga yoziladi. Maxfiy kalit SK faqat hujumchida bo‘lgani uchun, sxema to‘g‘ri amalga oshirilgan bo‘lsa, fayllarni kalitsiz tiklab bo‘lmaydi [22, 26].

Teskari muhandislik analitigi to‘rtta nuqtaga diqqat qaratadi. Birinchisi – tasodifiy son manbai: generator zaif bo‘lsa (masalan, vaqtga asoslangan boshlang‘ich qiymat), kalit taxmin qilinishi mumkin; generator kuchli bo‘lsa ham, uning chaqiruvlarini kuzatish orqali sessiya kalitini ushlab olish mumkin. Ikkinchisi – shifrlash funksiyasi: algoritm va rejim 2-bo‘limda ko‘rilgan usullar bilan aniqlanadi. Uchinchisi – ochiq kalit va uni o‘rash formati: kalit bloblari (“RSA1”) yoki PEM qatorlari topiladi, shifrlangan kalitning fayldagi o‘rni aniqlanadi. To‘rtinchisi – xotirada kalit qoldiqlari, ya’ni kalit materialining dastur ishlagandan keyin ham xotirada qolishi.



ransomware gibrid sxemasi · 4 nuqta

5-rasm. Ransomware dasturlarining gibrid shifrlash sxemasi

4.2. WannaCry misoli: kalit materialini boshqarishdagi xato

To‘rtinchi nuqtaga 2017-yilgi WannaCry hodisasi yaqqol misol bo‘ladi. Dastur RSA kalit juftligini zararlangan kompyuterning o‘zida hosil qilgan, ammo Windows CryptoAPI funksiyalari (CryptDestroyKey va CryptReleaseContext) tub sonlarni xotiradan o‘chirmasdan bo‘shatgan. Fransiyalik tadqiqotchi Adrien Guinet bu holatni aniqlab, xotiradagi p va q tub sonlarni topish orqali maxfiy kalitni qayta hisoblaydigan WannaKey vositasini yaratdi; Benjamin Delpy esa uning asosida avtomatlashtirilgan WanaKiwi vositasini ishlab chiqdi [32, 33]. Usul faqat kompyuter qayta yuklanmagan va xotira boshqa jarayonlar tomonidan qayta yozilmagan taqdirdagina ishlagan va Windows XP dan Windows 7 gacha bo‘lgan tizimlarda sinalgan [33]. Bu misolda algoritmlar (RSA va AES) to‘g‘ri tanlangan, xato esa kalit materialini boshqarishda bo‘lgan.

4.3. Himoya choralari

Himoya tomonidan taklif etilgan yondashuvlardan biri – PayBreak tizimi. U kriptografik funksiyalarni dinamik hooking usuli bilan kuzatib, fayl sessiya kalitlarini ushlab oladi, ularni ochiq kalit bilan shifrlab “kalit omborida” saqlaydi; zararlanish yuz bersa, foydalanuvchi omborni o‘z maxfiy kaliti bilan ochib, fayllarni to‘lovsiz tiklaydi. Mualliflar tizimni yigirmata real ransomware oilasi bo‘yicha sinab, ulardan o‘n ikkitasida barcha fayllarni tiklash mumkinligini ko‘rsatgan [22]. Shuningdek, Europol, Gollandiya milliy politsiyasi va yetakchi kompaniyalar tashabbusi bilan yaratilgan No More Ransom loyihasida turli ransomware oilalari uchun bepul deshifrlash vositalari to‘planadi [31].

Tahlil natijalari quyidagi himoya choralarini asoslaydi:

- muhim ma’lumotlarning tarmoqdan ajratilgan (offline) zaxira nusxalarini yuritish;
- fayl tizimida ommaviy o‘zgartirish va yuqori entropiyali yozuvlarni (2.1-banddagi usul) kuzatadigan xatti-harakat monitoringini joriy etish;
- tizim va ilovalarni muntazam yangilab borish;
- YARA qoidalari va capa kabi vositalar yordamida kriptografik konstantalar bo‘yicha zararli namunalarni aniqlash [27, 29];
- dasturchilar uchun kalit materialini ishlatgandan so‘ng xotiradan albatta o‘chirish (masalan, SecureZeroMemory orqali) va tasdiqlangan kutubxonalardan foydalanish.

Bunday tahlillar faqat izolyatsiyalangan laboratoriya muhitida, amaldagi qonunchilikka, jumladan “Kiberxavfsizlik to‘g‘risida”gi Qonun talablariga rioya qilgan holda olib borilishi shart [2]

Xulosa

Mazkur maqolada kriptografik algoritmlarning teskari muhandislikda qo‘llanilishi va shifrni buzish usullari tahlil qilindi, quyidagi natijalarga erishildi:

- simmetrik (AES, O‘z DSt 1105:2009) va asimmetrik (RSA, ECC) algoritmlar tasniflandi, ularning teskari muhandislikdagi xarakterli belgilari (S-box, katta sonlar arifmetikasi) solishtirildi;
- algoritmlarni ikkilik kodda aniqlashning besh usuli (entropiya, konstantalar, import va satrlar, strukturaviy va avtomatik tahlil, dinamik tasdiqlash) ko‘rib chiqilib, aniqlash ketma-ketligi umumlashtirildi;
- entropiya tajribasi shifrlangan va siqilgan ma’lumotlar entropiyasi taxminan 7,99 bit/bayt ekanini, bir baytli XOR esa entropiyani o‘zgartirmasligini ko‘rsatdi;
- kriptotahlil usullari tasniflandi va zararli dasturlar tahlilida eng samarali yo‘nalish algoritmning o‘zini buzish emas, balki realizatsiya xatolarini va kalit materialini tahlil qilish ekani asoslandi;
- WannaCry va PayBreak misollarida gibrid shifrlash sxemasining zaif nuqtalari hamda himoya choralari ko‘rsatildi.

Olingan natijalardan zararli dasturlarni teskari muhandislik qilish bo‘yicha amaliy mashg‘ulotlarda, shuningdek, ransomware hujumlariga qarshi choralarni ishlab chiqishda foydalanish mumkin.

Foydalanilgan adabiyotlar

1. O‘zbekiston Respublikasi Prezidentining 2022-yil 28-yanvardagi “2022–2026-yillarga mo‘ljallangan Yangi O‘zbekistonning taraqqiyot strategiyasi to‘g‘risida”gi PF-60-son Farmoni.
2. O‘zbekiston Respublikasining 2022-yil 15-apreldagi “Kiberxavfsizlik to‘g‘risida”gi O‘RQ-764-son Qonuni. URL: https://lex.uz/docs/5960604 (murojaat sanasi: 09.11.2026).
3. O‘zbekiston Respublikasi Prezidentining 2020-yil 5-oktabrdagi “«Raqamli O‘zbekiston – 2030» strategiyasini tasdiqlash va uni samarali amalga oshirish chora-tadbirlari to‘g‘risida”gi PF-6079-son Farmoni.

4. O‘z DSt 1105:2009. Axborot texnologiyasi. Axborotning kriptografik muhofazasi. Ma’lumotlarni shifrlash algoritmi. – Toshkent: O‘zstandart agentligi, 2009.
5. Ganiyev S.K., Ganiyev A.A., Xudoyqulov Z.T. Kiberxavfsizlik asoslari: O‘quv qo‘llanma. – Toshkent, 2020.
6. Akbarov D.Y. Axborot xavfsizligini ta’minlashning kriptografik usullari va ularning qo‘llanilishi. – Toshkent: O‘zbekiston markasi, 2009.
7. Schneier B. Applied Cryptography: Protocols, Algorithms, and Source Code in C. 2nd ed. – New York: John Wiley & Sons, 1996.
8. Menezes A.J., van Oorschot P.C., Vanstone S.A. Handbook of Applied Cryptography. – Boca Raton: CRC Press, 1996.
9. Stallings W. Cryptography and Network Security: Principles and Practice. 7th ed. – Boston: Pearson, 2017.
10. Daemen J., Rijmen V. The Design of Rijndael: AES – The Advanced Encryption Standard. – Berlin: Springer, 2002.
11. Sikorski M., Honig A. Practical Malware Analysis: The Hands-On Guide to Dissecting Malicious Software. – San Francisco: No Starch Press, 2012.
12. Eagle C. The IDA Pro Book: The Unofficial Guide to the World’s Most Popular Disassembler. 2nd ed. – San Francisco: No Starch Press, 2011.
13. Dang B., Gazet A., Bachaalany E. Practical Reverse Engineering: x86, x64, ARM, Windows Kernel, Reversing Tools, and Obfuscation. – Indianapolis: Wiley, 2014.
14. Rivest R.L., Shamir A., Adleman L. A Method for Obtaining Digital Signatures and Public-Key Cryptosystems // Communications of the ACM. – 1978. – Vol. 21, No. 2. – P. 120–126.