ANDROID OPERATSION TIZIMINING XAVFSIZLIGINI TA'MINLASH USULLARI

Habibjonova Guldofarid Murodjon qizi
Muxammad al – Xorazmiy nomidagi TATU "Kiberxavfsizlik va kriminalistika" kafedrasi stajyor o'qituvchi
Aplidinov Abrorbek Ibolitdin o'g'li
Toshtemirov Ulug'bek Kamolovich
Ko'chimova Oyshabonu O'tkirjon qizi
Muxammad al – Xorazmiy nomidagi TATU talabalari

Annotatsiya: Ushbu maqolada Android operatsion tizimidagi mobil ilovalarga xos bo'lgan asosiy tahdidlar va zaifliklar tahlil qilinadi. Statistik ma'lumotlar asosida Android va iOS platformalarida aniqlangan yuqori xavfli zaifliklar, ularning kelib chiqish sabablari hamda hujumchilar tomonidan foydalanish imkoniyatlari ko'rib chiqiladi. Mijoz va server tomonidagi zaifliklar, xavfsiz ma'lumotlarni saqlash, autentifikatsiya mexanizmlaridagi kamchiliklar va zararli dasturlar tahdidi yoritilgan. Shuningdek, Android operatsion tizimida xavfsizlikni ta'minlash usullari, jumladan shifrlash texnologiyalari, Application Sandbox, SELinux, tasdiqlangan yuklash (Verified Boot) va kriptografik mexanizmlar tahlil qilinadi. Maqola mobil ilovalar xavfsizligini oshirish bo'yicha amaliy tavsiyalar berish bilan yakunlanadi.

Kalit so'zlar: Android, mobil ilovalar xavfsizligi, zaifliklar, kiberxavfsizlik, shifrlash, Application Sandbox, SELinux, IPC, zararli dasturlar, autentifikatsiya, mobil tahdidlar.

To'liq fayl tizimini shifrlashni qo'llab-quvvatlovchi va xavfsiz aloqa kanallarini ta'minlovchi ma'lumotlardan tashqari, Android kriptografiya yordamida ma'lumotlarni himoya qilish uchun keng doiradagi algoritmlarni taqdim etadi.

Kalit generatori tomonidan ishlab chiqarilgan har qanday kriptografik kalitlarni ishga tushirish uchun ishonchli tasodifiy sonli generator, SecureRandom-dan foydalaning. Xavfsiz tasodifiy sonli generator bilan yaratilmagan kalitni ishlatish algoritmning kuchini sezilarli darajada zaiflashtiradi va oflayn hujumlarga yo'l qo'yishi mumkin.

Kalitni takroriy foydalanish uchun saqlash kerak bo'lsa, KeyStore kabi mexanizmlardan foydalaning, bu kriptografik kalitlarni uzoq muddat saqlash va olish mexanizmini ta'minlaydi.

Shifrlash. Shifrlash simmetrik shifrlash kalitlari yordamida Android qurilmasidagi barcha foydalanuvchi ma'lumotlarini kodlash jarayonidir. Agar qurilma shifrlangan bo'lsa, foydalanuvchi tomonidan yaratilgan barcha ma'lumotlar diskka yozilishdan oldin avtomatik ravishda shifrlanadi va barcha o'qish ma'lumotlarni chaqirish jarayoniga qaytarishdan oldin avtomatik ravishda uni deshifrlaydi. Shifrlash, ruxsatsiz shaxslar ma'lumotlarga kirishga harakat qilsa, uni o'qib chiqa olmasligini ta'minlaydi.

Android qurilmalarda shifrlashning ikkita usuli bor: to'liq disk shifrlash va faylga asoslangan shifrlash.

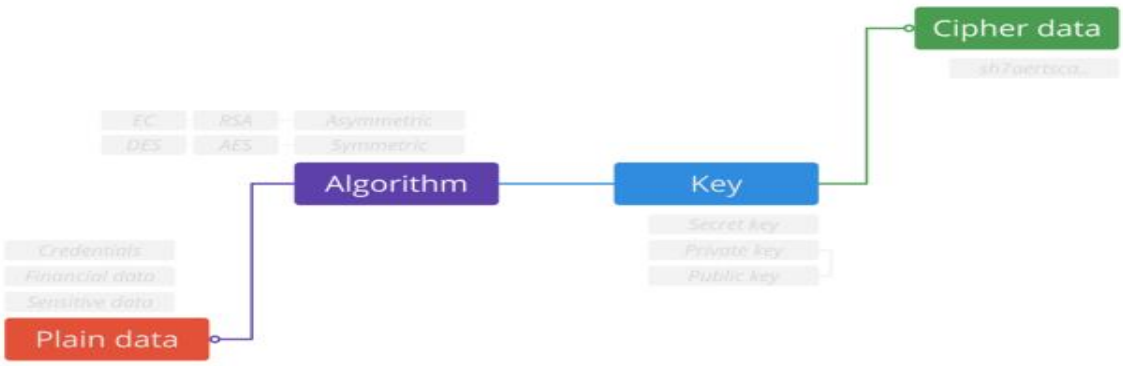
To‘liq disk shifrlash. Android 5.0 va undan yuqori muhitda to‘liq diskda shifrlashni qo‘llab-quvvatlaydi. To‘liq diskda shifrlash bitta kalit-foydalanuvchining qurilmadagi parol bilan himoyalanganligi-qurilmadagi foydalanuvchi ma’lumotlarini butunicha himoyalashdan foydalanadi. Yuklashda, foydalanuvchi diskka kirish uchun ruxsatnomalarini taqdim etishdan oldin o‘zlarining ishonchini komil qilishi kerak.

Bu xavfsizlik uchun yaxshi bo‘lsa-da, demak, telefonning asosiy funktsiyalarini ko‘pchilik foydalanuvchilar o‘z qurilmalarini qayta ishga tushirganda darhol foydalanishlari mumkin emas. Chunki ma’lumotlariga kirish signallar funksiyalari ishlamasligi, kirish xizmatlari mavjud emasligi va telefonlar qo‘ng‘iroqlarni qabul qila olmasligi kabi ularning yagona foydalanuvchi identifikatorlari ortida saqlanadi.

Faylga asoslangan shifrlash. Android 7.0 va undan yuqori muhitda faylga asoslangan shifrlashni qo‘llab-quvvatlaydi. Faylga asoslangan shifrlash turli fayllarni mustaqil ravishda ochilishi mumkin bo‘lgan turli xil kalitlar bilan shifrlanishiga imkon beradi. Faylga asoslangan shifrlashni qo‘llab-quvvatlaydigan qurilmalar, shuningdek, shifrlangan qurilmalarning qulflangan ekranga to‘g‘ri yuklash imkonini beruvchi "Direct Boot" deb nomlangan yangi xususiyatni qo‘llab-quvvatlay oladi, shuning uchun kirish xizmatlari va signallarni kabi muhim qurilmalarga tez kirish imkonini beradi.

Dasturlarni shifrlashdan xabardor qilish uchun faylga asoslangan shifrlash va yangi APIlarni kiritish bilan, bu ilovalar cheklangan kontekstda ishlashi mumkin. Foydalanuvchilar shaxsiy ma’lumotlarini himoya qilishda o‘z hisob ma’lumotlarini taqdim etish vaqtida sodir bo‘lishi mumkin.

1.1-rasm. Shifrlash algoritmi



Keyinchalik, ba’zi bir **algoritm** asosida maxsus **kalit** yaratiladi va uni **shifrlash ma’lumotlarini** yaratish uchun **ishlatiladi**.

Algoritm turlari. Yuqorida biz shifrlashning eng oddiy namunasini ko‘rdik. Bugungi kunda algoritmlar yanada murakkab va simmetrik va assimetrik (alohida kalitni talab qilmaydigan va ushbu maqolada ko‘rib chiqilmaydigan *xesh funktsiyalari ham mavjud) ajratiladi.*

Simmetrik - eng qadimgi va eng mashhur texnika. Shifrlash kaliti va parol hal qilish tugmasi bir xil. Bundan tashqari, odatda, "Stream Cipher" yoki "Blok-parol" deb nomlanadi.

Eng keng tarqalgan nosimmetrik **AES** - Kengaytirilgan shifrlash standarti (AES) AQSh hukumati va ko‘plab tashkilotlar tomonidan standart sifatida tasdiqlangan

algoritmdir.

Asimmetrik - kriptografiyaning zamonaviy bo‘limi. Shuningdek, algoritmlarda juft kalitlarni (ochiq kalit va shaxsiy kaliti) ishlatadigan va juftning boshqa komponentini algoritmning turli bosqichlarida ishlatadigan ochiq kalit kriptografiya deb ham ataladi.

Eng keng tarqalgan assimetrik algoritm - **RSA** - umumiy kalit shifrlash algoritmi va internet orqali yuborilgan ma’lumotlarni shifrlash uchun standart.

Stream cipher - bu simvolli shifrlash algoritmi, ma’lumotlar bir vaqtning o‘zida bir yoki bir baytda ishlaydigan kalit bilan birga tasodifiy shriftlar yoki tekis ma’lumotlar.

Bloklarni shifrlash - **to‘siq** deb ataladigan qattiq uzunlikdagi bit guruhlarida ishlaydigan deterministik algoritm. Blokirovka shifrlar juda ko‘p kriptografik protokollarni loyihalashda muhim elementlar bo‘lib, ommaviy ma’lumotlarni shifrlashda keng qo‘llaniladi.

Tartiblar va to‘ldirishlar. Blokirovka shifrlashi turli darajalarda va himoyalash darajasiga ega bo‘lib, uni himoya darajasini oshiradi.

Modlar - operatsion usuli bir blokdan kattaroq ma’lumotlarning xavfsizligini konvertatsiya qilish uchun shifrlarni bir martalik operatsiyani qayta-qayta qo‘llashni ta’riflaydi.

Padding - blokli shifrlash qattiq o‘lchamdagi birliklarda ishlaydi (blok o‘lchami deb nomlanadi), ammo xabarlar turli uzunliklarda bo‘ladi. Shunday qilib, ayrim usullar (ya’ni, ECB va CBC) oxirgi blokni shifrlashdan oldin to‘ldirishni talab qiladi.

Eng ko‘p tarqalgan usullar:

ECB - elektron shifrlash usullarining eng oddiy kodi. Xabar bloklarga bo‘lingan va har bir blok alohida-alohida shifrlangan.

CBC - shifrlash blokirovka zanjirida, har bir shifrlash ma’lumot bloklari shu nuqtaga qadar qayta ishlangan barcha ma’lumotlar bloklariga bog‘liq. Har bir xabarni yagona qilish uchun birinchi blokda ishga tushirish vektori qo‘llanilishi kerak.

Lekin algoritm nosimmetrik emas, chunki bu usullar va to‘ldirishlar bo‘lishi mumkin emas. Masalan, RSA algoritmi ECB rejimi va PKCS1Padding bilan ishlatilishi mumkin.

Asosiy turlari. Uchta kalit turi mavjud: Yashirin kalit, Xususiy kalit va ochiq kalit.

Yashirin kalit - xabarni shifrlash va shifrlash uchun an’anaviy *simmetrik* shifrlashda ishlatiladigan bitta maxfiy kalit.

Xususiy kalit - *assimetrik* kriptografiya uchun parol hal qilish uchun ishlatiladigan juft kriptografik kalitlarning maxfiy tarkibi.

Umumiy kalit - *assimetrik* kriptografiya uchun shifrlash uchun foydalaniladigan juft kriptografik kalitlarning ommaviy komponenti.

Operatsion tizim darajasida Android platformasi Linux yadrosi xavfsizligini ta’minlaydi, shuningdek turli jarayonlarda ishlaydigan ilovalar o‘rtasida xavfsiz aloqani ta’minlash uchun xavfsiz jarayonlararo aloqa (IPC) muhitini ta’minlaydi. OS darajasidagi ushbu xavfsizlik xususiyatlari, hatto mahalliy kodni Application Sandbox tomonidan cheklanganligini kafolatlaydi. Ushbu kod dasturiy ta’minotning xatti-harakatlari yoki dasturning zaiflikdan foydalanish natijasi bo‘ladimi-yo‘qmi, tizim nayrang dasturining

boshqa ilovalarga, Android tizimiga yoki qurilmaning o‘ziga zarar yetkazishiga yo‘l qo‘ymaslik uchun mo‘ljallangan.

Linux xavfsizligi. Android platformasining asosi Linux yadrosidir. Linux yadrosi yillar davomida keng tarqalgan bo‘lib foydalanilgan va millionlab xavfsizlikka sezgir muhitlarda ishlatilgan. Linux muntazam ravishda o‘rganilayotgan, hujum qilingan va minglab ishlab chiqaruvchilar tomonidan aniqlangan tarixi tufayli, Linux ko‘plab korporatsiyalar va xavfsizlik mutaxassislari tomonidan ishonchli va xavfsiz yadroga aylandi.

Mobil hisoblash muhiti uchun asos sifatida Linux yadrosi Android-ni quyidagi asosiy xavfsizlik xususiyatlari bilan ta’minlaydi:

- Foydalanuvchiga asoslangan ruxsatnomalar modeli.
- Jarayon izolyatsiyasi.
- Xavfsiz IPQ uchun kengaytirilgan mexanizm.
- Kernenin keraksiz va potensial xavfli qismlarini olib tashlash qobiliyati.

Linux yadrosi asosiy xavfsizlik maqsadi, foydalanuvchi resurslarini bir-biridan ajratishdir. Linux xavfsizligi foydalanuvchi resurslarini bir-biridan himoya qilishdir. Shunday qilib, Linux:

- A foydalanuvchi B foydalanuvchining fayllarini o‘qishiga yo‘l qo‘ymaydi.
- A foydalanuvchisi B foydalanuvchining xotirasidan chiqmasligini ta’minlaydi.
- A foydalanuvchisi B foydalanuvchining CPU resurslarini ishlatmasligini ta’minlaydi.
- A foydalanuvchining B foydalanuvchi qurilmalarini ishlatmasligini ta’minlaydi (masalan, telefoniya, GPS, Bluetooth).

Dastur sandboxi. Android platformasi dasturni ishlatish manbalarini identifikatsiya qilish va ajratish vositasi sifatida Linux foydalanuvchilariga asoslangan himoyadan foydalanadi. Android tizimi har bir Android ilovasiga noyob foydalanuvchi identifikatorini (UID) tayinlaydi va u foydalanuvchini alohida jarayonda boshqaradi. Ushbu yondashuv bir nechta ilovalar bir xil foydalanuvchi ruxsatnomalari bilan ishlaydigan boshqa operatsion tizimlardan (an’anaviy Linux konfiguratsiyasi, shu jumladan) farqlanadi.

Bu yadro darajasidagi Application Sandbox-ni o‘rnatadi. Dastur yadrosi ilovalarga tayinlangan foydalanuvchi va guruh identifikatorlari kabi standart Linux funktsiyalari orqali ilovalar va tizimlar o‘rtasida xavfsizlikni ta’minlaydi. Odatiy bo‘lib, ilovalar bir-biri bilan ta’sir o‘tkaza olmaydi va ilovalar operatsion tizimiga cheklangan kirish imkoniyatiga ega. Agar A ilovasi o‘qilgan dastur B singari zararli narsa qilishga harakat qilsa yoki ruxsatsiz telefonni teradi (alohida dastur bo‘lsa), operatsion tizim bu dasturdan himoyalanadi, chunki A ilovasi tegishli foydalanuvchi imtiyozlariga ega emas. Sandbox oddiy, tekshirilishi mumkin va o‘n yilliklar davomida UNIX tarzi foydalanuvchi jarayonlari va fayl ruxsatini ajratib turadi.

Application Sandbox yadroda bo‘lgani uchun, ushbu xavfsizlik modeli mahalliy kodga va operatsion tizim ilovalariga amal qiladi. Operatsion tizim kutubxonalari, dastur doirasi, dasturni ishga tushirish vaqti va barcha ilovalar kabi yadro ustida joylashgan barcha dasturlar amaliy sandbox ichida ishlaydi. Ba’zi maydonchalarda ishlab chiquvchilar

xavfsizlikni ta’minlash uchun ma’lum bir rivojlanish doirasiga, API yoki tilga to‘siq qo‘yishadi. Androidda xavfsizlikni ta’minlash uchun zarur bo‘lgan dasturni qanday yozish mumkinligi haqida hech qanday cheklovlar yo‘q; Bu borada, mahalliy kod, tarjima kodi kabi xavfsizdir.

Ba’zi operatsion tizimlarda, bir amalda xotira buzilishidagi xatolar bir xil xotira maydoniga joylashtirilgan boshqa ilovalarda korruptsiyaga olib kelishi mumkin, bu esa qurilmaning xavfsizligini to‘liq ta’minlashga olib keladi. Barcha ilovalar va ularning resurslari OS darajasida qumli bo‘lganligi sababli, xotira buzilishi xatosi operatsion tizim tomonidan o‘rnatilgan ruxsatnomalar bilan faqat ushbu ilovaning kontekstida o‘zboshimchalik bilan kod bajarilishiga imkon beradi.

Barcha xavfsizlik xususiyatlari kabi, Application Sandbox buzilmaydi. Shu bilan birga, to‘g‘ri tuzilgan qurilma ichida Application Sandboxdan chiqish uchun Linux yadrosi xavfsizligini buzish kerak.

Tizim bo‘limi va xavfsiz rejim. Tizim bo‘limida Android yadrosi, operatsion tizim kutubxonalari, dastur ish vaqti, dastur doirasi va ilovalar mavjud. Ushbu bo‘lim faqat o‘qish uchun o‘rnatiladi. Agar foydalanuvchi qurilmani xavfsiz rejimga tushirganda, uchinchi tomon ilovalari qurilma egasi tomonidan qo‘lda ishga tushishi mumkin, lekin sukut bo‘yicha ishga tushirilmaydi.

Fayl tizimi ruxsatnomalari. UNIX tarzi muhitda fayl tizimi ruxsatnomalari bitta foydalanuvchining boshqa foydalanuvchining fayllarini o‘zgartira olmaydi yoki o‘qimasligini ta’minlaydi. Androidda, har bir dastur o‘z foydalanuvchisi sifatida ishlaydi. Ishlab chiquvchilar boshqa ilovalar bilan ochiq fayllarni ulashmagan ekan, bitta dastur tomonidan yaratilgan fayllar boshqa dastur tomonidan o‘qilmaydi yoki o‘zgartirilmaydi.

Xavfsizlikni yaxshilash uchun Linux. Android kirishni boshqarish siyosatini amalga oshirish va jarayonlarga majburiy kirishni boshqarish (Mac) o‘rnatish uchun Xavfsizlikni kengaytiruvchi Linuxni (SELinux) ishlatadi.

Tasdiqlangan yuklash. Android 6.0 va undan keyingi versiyalari tasdiqlangan yuklash va qurilmaning xaritasi-verity-ni qo‘llab-quvvatlaydi. Tasdiqlangan yuklash tizimning dasturiy ta’minotning yaxlitligini kafolatlaydi, tizimning ishonchli ildizidan tizim qismiga qadar. Yuklash vaqtida, har bir bosqich kriptografik jihatdan uni ijro etishdan oldin keyingi bosqichning yaxlitligi va haqiqiyligini tekshiradi.

Android 7.0 va undan keyingi versiyalari qat’iy majburiy tasdiqlangan yuklashni qo‘llab-quvvatlaydi, ya’ni buzilgan qurilmalar ocholmaydi.

Kriptografiya. Android ilovalar tomonidan ishlatilishi uchun kriptografik API-larni taqdim etadi. Bunga AES, RSA, DSA va SHA kabi standart va keng tarqalgan ishlatiladigan kriptografik ibtidoiy qo‘llanmalar kiradi. Bundan tashqari, API-lar SSL va HTTPS kabi yuqori darajadagi protokollarga ega.

Android 4.0 ilovalar shaxsiy kalitlar va sertifikat zanjirlari uchun tizim hisob ma’lumotlarini saqlash uchun ruxsat berish uchun KeyChain sinfini taqdim etdi.

Qurilmalarni rootlash. Odatiy bo‘lib, Android-da faqat yadro va yadro ilova kichik to‘plami ildiz ruxsatlarni bilan ishlaydi. Android operatsion tizimni, yadroni yoki boshqa ilovani o‘zgartirish uchun ildiz ruxsatnomalari bilan foydalanuvchi yoki ilovaga to‘sqinlik

qilmaydi. Umuman olganda, ildiz barcha ilovalarga va barcha ilovalarga to‘liq kirish imkoniyatiga ega. Android qurilmasidagi ilovalarni ildizga kirish huquqini o‘zgartiradigan ruxsatnomalarni o‘zgartiradigan foydalanuvchilar zararli dasturlarga va potentsial amaliy kamchiliklarga qarshi xavfsizlik ta’sirini oshiradi.

Android qurilmasini o‘zgartirish qobiliyati Android platformasi bilan ishlaydigan ishlab chiquvchilar uchun muhimdir. Ko‘pgina Android qurilmalarida foydalanuvchilarga alternativ operatsion tizimni o‘rnatishga ruxsat berish uchun bootloaderning qulfini ochish imkoniyati mavjud. Ushbu muqobil operatsion tizimlar egalariga ilovalar va tizim tarkibiy qismlarini disk raskadrovka maqsadlarida yoki Android API-lariga ilovalarga taqdim qilinmagan xususiyatlarga kirish uchun rootdan foydalanish huquqini beradi.

Ba’zi qurilmalarda qurilmani jismoniy boshqarish va USB kabeliga ega bo‘lgan foydalanuvchi foydalanuvchiga root privilegiyalarini ta’minlaydigan yangi operatsion tizim o‘rnatishi mumkin. Mavjud foydalanuvchi ma’lumotlarini uzilishdan himoya qilish uchun bootloaderni qulfni ochish mexanizmi bootloaderning mavjud bo‘lgan foydalanuvchi ma’lumotlarini qulfni ochishning bir qismi sifatida o‘chirishini talab qiladi. Kernel bug yoki xavfsizlik teshiklaridan foydalanish orqali olingan root erkin foydalanish bu muhofazani chetlab o‘tishi mumkin.

Qurilmada saqlangan kalit bilan ma’lumotlarni shifrlash ilova ma’lumotlarini ildiz foydalanuvchilardan himoya qilmaydi. Ilovalar serverda yoki foydalanuvchi parolida saqlangan off-qurilma saqlanadigan kalit bilan shifrlashdan foydalanib ma’lumotlarni himoya qilish qatlami qo‘shishlari mumkin. Bunday yondashuv kaliti mavjud bo‘lganda vaqtinchalik himoyani ta’minlashi mumkin, lekin ba’zi hollarda dasturga kalitni kiritish kerak va u keyin root foydalanuvchilarga kirish imkoniyatiga ega bo‘ladi.

XULOSA

Mobil ilovalar kundalik hayotning ajralmas qismiga aylanganligi sababli ularning xavfsizligini ta’minlash dolzarb masalalardan biri hisoblanadi. Tadqiqot natijalari shuni ko‘rsatadiki, Android platformasidagi ilovalarning katta qismida yuqori va o‘rta darajadagi xavfli zaifliklar mavjud bo‘lib, ularning aksariyati mijoz tomonida joylashgan. Xavfsizlik mexanizmlaridagi xatolar, ma’lumotlarni himoyasiz saqlash va autentifikatsiya jarayonidagi kamchiliklar foydalanuvchi ma’lumotlarining o‘g‘irlanishiga olib kelishi mumkin.

Android operatsion tizimi Linux yadrosi, sandbox mexanizmi, faylga asoslangan shifrlash va kriptografik API’lar orqali yuqori darajadagi himoyani ta’minlasa-da, ishlab chiquvchilar tomonidan xavfsizlik talablariga rioya qilinmasligi umumiy xavfsizlik darajasini pasaytiradi. Shuning uchun mobil ilovalarni loyihalash va ishlab chiqish jarayonida xavfsizlikni birlamchi omil sifatida ko‘rib chiqish, zamonaviy kriptografik usullardan foydalanish va server hamda mijoz tomonidagi himoya mexanizmlarini kuchaytirish zarur.

FOYDALANILGAN ADABIYOTLAR:

1. Juniper Research. *Mobile Banking Users Worldwide* – Statistik hisobot, 2018.
2. Positive Technologies. *Mobile Application Security Research*, 2018.

3. OWASP Foundation. *OWASP Mobile Top 10 – 2016*.
4. McAfee Labs. *Mobile Malware Threat Report*, 2018.
5. Android Developers. *Android Security Overview*.
6. Android Developers. *WebView Security Best Practices*.
7. Google. *Android Encryption and KeyStore Documentation*.
8. Linux Foundation. *Linux Kernel Security Overview*.