O’ZBEKISTON RESPUBLIKASI OLIY TA’LIM, FAN VA INNOVATSIYALAR VAZIRLIGI

TERMIZ DAVLAT UNIVERSITETI

MAGISTRATURA BO’LIMI

Qo’lyozma huquqi asosida

UDK_____

MUSTAFOQULOV NORSOAT UMBAR O‘G‘LI

INTERNET MAKONIDA MAXFIY MA`LUMOTLARNI HIMOYA QILISHNI TA`MINLASH MODEL VA ALGORITMLARI

Mutaxassislik: 70610201 – Kompyuter tizimlari va ularning dasturiy ta’minoti (tarmoqlar va sohalar bo’yicha)

Magistr akademik darajasini olish uchun yozilgan

DISSERTATSIYA

Ilmiy rahbar: _____ t.f.f.d (PhD) Sh.R.G`ulomov

TERMIZ – 2024

Magistrlik dissertatsiyasi mavzusi Termiz davlat universiteti rektorining 2024-yil 19-yanvardagi №5-T/M sonli buyrug‘i asosida tasdiqlangan.

Magistrlik dissertatsiyasi Termiz davlat universiteti kafedrasida bajarilgan

Magistrlik dissertatsiyasi elektron nusxasi Termiz davlat universitetining rasmiy web sahifasiga joylashtirilgan.

Dissertatsiya manzilining QR-kodi:

Magistrlik dissertatsiyasi bilan Termiz davlat universitetining axborot-resurs markazida tanishish mumkin (_____ raqam bilan ro‘yxatga olingan. Manzil: Termiz shahri, Barkamol avlod ko‘chasi 43-uy.)

Ilmiy rahbar: _____t.f.f.d(PhD) Sh.R.G`ulomov

Kafedra mudiri: _____ i.f.d., prof. O.Q.Xatamov

Magistratura bo‘limi boshlig‘i: _____ D. Abduraximov

ANNOTATSIYA

Tayanch so’zlar: Veb-saytlar, internet xavfsizlik choralari, ma’lumotlarni himoya model, usul, algoritmlari, kommunal to‘lovlar.

Tadqiqot ob’yekti: Tadqiqot ishining ob’yekti internet tarmog‘ida ma’lumotlar xavfsizligiga nisbatan bo‘ladigan tahdidlar hisoblanadi.

Ishning maqsadi: Mazkur magistrlik dissertatsiyasining maqsadi internet tarmog‘ida maxfiy ma`lumotlarni himoya qilishni ta`minlashning model va algoritmlaridan foydalanish usullarini tadqiq va tahlil qilish.

Tadqiqot metodlari: internet veb-saytlar, veb-xizmatlar, model, usul, algoritm, elektron tijorat.

Olingan natijalar va ularning yangiligi:

1.Dissertatsiya ishida internet makonida maxfiy ma`lumotlarni himoya qilishni ta`minlash model va algoritmlari tadqiqi keltirilgan.

2. Internet makonida veb-saytlar xavfsizligi muammolari, internet tarmog‘i veb-saytlari uchun xavfsizlik choralarini ishlab chiqish, ma’lumotlarni himoya qilishning model, usul va algoritmlari.

3.Kommunal to‘lovlar xizmati veb-dasturiy ta’minoti misolida keltirib berilgan.

Amaliy ahamiyati: Kommunal to‘lov tizimi dasturiy taminotini yaratish bosqichlari keltirib o‘tildi. Kommunal to‘lov tizimi dasturiy taminotini server qismini yaratish boshqichlari va Kommunal to‘lov tizimi dasturiy taminotini client qismini yaratish bosqichlari yaratish texnologiyalari misolida keltirildi. Shuningdek, Kommunal to‘lov tizimi dasturiy taminotidan foydalanish yo‘riqnomasi keltirildi.

Tadbiq etish darajasi va iqtisodiy samaradorligi. . Kommunal to‘lov tizimi dasturiy taminotini yaratish bosqichlari keltirib o‘tildi. Kommunal to‘lov tizimi dasturiy taminotini server qismini yaratish boshqichlari va Kommunal to‘lov tizimi dasturiy taminotini client qismini yaratish bosqichlari yaratish texnologiyalari

misolida keltirildi. Shuningdek, Kommunal to‘lov tizimi dasturiy taminotidan foydalanish yo‘riqnomasi keltirildi.

Qo’llash sohasi: Internet makonida Kommunal to‘lov tizimlari ma’lumotlarini bazada himoyalash modellari va usullari keltirildi.

АННОТАЦИЯ

Ключевые слова: Веб-сайты, меры безопасности в Интернете, модель защиты данных, метод, алгоритмы, счета за коммунальные услуги.

Объект исследования: Объектом исследования являются угрозы безопасности информации в сети Интернет.

Цель работы: Целью данной магистерской работы является исследование и анализ методов использования моделей и алгоритмов для обеспечения защиты конфиденциальной информации в сети Интернет.

Методы исследования: интернет-сайты, веб-сервисы, модель, метод, алгоритм, электронная коммерция.

Полученные результаты и их новизна:

1. В диссертационной работе представлено исследование моделей и алгоритмов обеспечения защиты конфиденциальной информации в интернет-пространстве.

2. Проблемы безопасности веб-сайтов в Интернет-пространстве, разработка мер безопасности Интернет-сайтов, моделей, методов и алгоритмов защиты информации.

3. В качестве примера веб-программы приведен сервис оплаты коммунальных услуг.

Практическая значимость: Рассмотрены этапы создания программного обеспечения системы оплаты коммунальных услуг. В качестве примеров технологий приведены этапы создания серверной части программного обеспечения системы оплаты коммунальных услуг и этапы

создания клиентской части системы оплаты коммунальных услуг. Также были предоставлены инструкции по использованию программного обеспечения «Коммунальной платежной системы».

Уровень реализации и экономическая эффективность: Рассмотрены этапы создания программного обеспечения системы оплаты коммунальных услуг. В качестве примеров технологий приведены этапы создания серверной части программного обеспечения системы оплаты коммунальных услуг и этапы создания клиентской части системы оплаты коммунальных услуг. Также были предоставлены инструкции по использованию программного обеспечения «Коммунальной платежной системы».

Область применения: представлены модели и методы защиты данных систем оплаты коммунальных услуг в интернет-пространстве.

ANNOTATION

Keywords: Websites, Internet security measures, data protection model, method, algorithms, utility bills.

Object of research: The object of research is threats to information security on the Internet.

Purpose of the work: The purpose of this master's thesis is to research and analyze methods for using models and algorithms to ensure the protection of confidential information on the Internet.

Research methods: Internet sites, web services, model, method, algorithm, e-commerce.

The results obtained and their novelty:

1. The research of models and algorithms for ensuring the protection of confidential information in the Internet space is presented in the dissertation work.

2. Website security issues in the Internet space, development of security measures for Internet websites, data protection models, methods and algorithms.

3. Utility payment service is given as an example of web software.

Practical significance: The stages of creating software for a utility payment system are considered. As examples of technologies, the stages of creating the server part of the utility payment system software and the stages of creating the client part of the utility payment system are given. Instructions for using the Utility Payment System software were also provided.

Level of implementation and economic efficiency: The stages of creating software for a utility payment system are considered. As examples of technologies, the stages of creating the server part of the utility payment system software and the stages of creating the client part of the utility payment system are given. Instructions for using the Utility Payment System software were also provided.

Scope of application: models and methods for protecting data from utility payment systems in the Internet space are presented.

MUNDARIJA

KIRISH 3

I BOB. INTERNET MAKONIDA VEB-SAYTLAR XAVFSIZLIGI MUAMMOLARI 5

1.1. Xavfsiz veb-saytlarning tuzilishi arxitekturalari, xavfsizlikni ta’minlashdagi muammolar va ularni hal qilish usullari 5

1.3. Internet makonidagi Veb-saytlarning zaifliklarini aniqlash usullari tahlili ... 26

I bob bo‘yicha xulosalar 31

II BOB. INTERNET TARMOG‘I VEB-SAYTLARI UCHUN XAVFSIZLIK CHORALARINI ISHLAB CHIQISH 33

2.1. Internet makonida Veb saytlarga uyushtiriladigan hujum turlari tahlili 33

2.2. Veb saytlarni yaratishda xavfsiz kodlash usullari va algoritmlari 45

2.3. OWASP standartlaridan foydalanib veb saytlarni kiberhujumlardan himoya qilish choralarini samaradorligini oshirish algoritmlari 52

II bob bo‘yicha xulosalar 62

III BOB. KOMMUNAL TO‘LOVLAR XIZMATI VEB-DASTURIY TA’MINOTI MISOLIDA INTERNET TARMOG‘IDA MA’LUMOTLAR XAVFSIZLIGINI TA’MINLASH 63

3.1. Kommunal to‘lov tizimlari ma’lumotlarini bazada himoyalash modellari va usullari 63

3.2. Kommunal to‘lov tizimi dasturiy taminotini yaratish bosqichlari 71

3.3. Kommunal to‘lov tizimi dasturiy taminotidan foydalanish yo‘riqnomasi 81

III bob bo‘yicha xulosalar 88

XULOSA 89

FOYDALANILGAN ADABIYOTLAR 91

KIRISH

Dissertatsiya mavzusining asoslanishi va uning dolzarbligi. 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 Farmoniga muvofiq, shuningdek, axborot-kommunikatsiya texnologiyalari sohasini yangi bosqichga olib chiqish bo‘yicha ustuvor vazifalarni amalga oshirish maqsadida, 2022 yil 22 avgust kuni Prezidenting “2022–2023 yillarda axborot-kommunikatsiya texnologiyalari sohasini yangi bosqichga olib chiqish chora-tadbirlari to‘g‘risida”gi Qarori imzolandi. Unga asosan axborot-kommunikatsiya texnologiyalari sohasini 2022-2023 yillarda yanada rivojlantirishning asosiy vazifalari etib belgilandi. O'zbekistonda dunyoning boshqa mamlakatlari kabi elektron hukumatni shakllantirish va joriy etishga katta e'tibor qaratilmoqda. Bu esa dasturiy mahsulotlarning rivojlanishi ularga qo‘yiladigan xavfsizlik talablarining oshishiga olib keladi, chunki texnologiyaga o‘sishda davom etarkan va kibertahdidlar yanada murakkablashib boraveradi.[1]

Darhaqiqat bugungi kunda har qanday davlatning rivojlanishi, uning jahon hamjamiyatida tutgan o‘rni ushbu mamlakatda Axborot–komunikatsiya texnologiyalari xizmatlarining rivojlanish darajasi, aholining turli guruhlari tomonidan ulardan foydalanishdagi imkoniyatlariga bevosita bog‘liq desak, mubolag‘a bo‘lmaydi.

o

‘ **Ilmiy tadqiqot ishining ob’yekti va predmetining belgilanishi.**

o Tadqiqot ishining ob’ekti internet tarmog‘ida ma’lumotlar xavfsizligiga nisbatan bo‘ladigan tahdidlar hisoblanadi.

o Tadqiqot predmetini esa internet tarmog‘ida maxfiy ma`lumotlarni himoya qilishni ta`minlashning model, usul va algoritmlari tashkil qiladi.

o **Ilmiy tadqiqot ishining maqsadi va vazifalari.**

‘

o

‘

Mazkur magistrlik dissertatsiyasining maqsadi internet tarmog‘ida maxfiy ma`lumotlarni himoya qilishni ta`minlashning model va algoritmlaridan foydalanish usullarini tadqiq va tahlil qilish.

Ushbu maqsadga erishish uchun quyidagi vazifalar qo‘yildi:

- xavfsiz veb-saytlarning tuzilishi arxitekturalari, xavfsizlikni ta’minlashdagi muammolar va ularni hal qilish usullarini tahlil qilish;
- internet makonidagi Veb-saytlarning zaifliklarini aniqlash usullari tahlili;
- internet makonida Veb saytlarga uyushtiriladigan hujum turlari tahlili;
- veb saytlarni yaratishda xavfsiz kodlash usullari va algoritmlari tahlili;
- veb saytlarni kiberhujumlardan himoya qilish choralarini samaradorligini oshirish vositalari va algoritmlari tahlili;
- to‘lov tizimlari ma’lumotlarini bazada himoyalash modellari va usullari tahlili;
- kommunal to‘lov tizimi dasturiy taminotini yaratish;
- kommunal to‘lov tizimi dasturiy taminotidan foydalanish yo‘riqnomasini ishlab chiqish.

Ilmiy tadqiqot ishining asosiy masalalari va farazlari. Zamonaviy internet tarmog’ida maxfiy ma`lumotlarni himoya qilishni ta`minlashning model, usul va algoritmlari yordamida veb-xizmatlarni ishlab chiqish dolzarb masalalardan sanaladi.

Mavzu bo‘yicha qisqacha adabiyotlar tahlili.

Mazkur tadqiqot ishi doirasida ko‘plab ilmiy ishlar, maqolalar va Internet resurlari o‘rganildi. Bular orasida B. Sullivan va boshqalar [5], С.А. Бабин [6] va M. McDonald va J. Richardson [7], I. Ristic [8], shuningdek Phil Hughes va Ed Ferrett ning [11] ilmiy tadqiqotlari alohida o‘rin tutadi.

Bundan tashqari turli ilmiy jurnallarda nashr etilgan ilmiy maqola va Internet ma’lumotlaridan foydalanildi.

Ilmiy tadqiqot ishida qo‘llanilgan uslublarning qisqacha tavsifi. Ushbu tadqiqot ishida qiyosiy tahlil, matematik statistika, modellashtirish, ob’ektga yo‘naltirilgan dasturlash, ma’lumotlar bazasi xavfsizligini ta’minlash usullari, web-dasturlash, usullaridan foydalanilgan.

Ilmiy tadqiqot ishining ilmiy yangiligi. Tadqiqot ishining zamonaviy internet tarmog’ida maxfiy ma`lumotlarni himoya qilishni ta`minlashning model, usul va algoritmlarini tadqiq qilinganligi hisoblanadi.

Dissertatsiya tarkibining qisqacha tavsifi

Mazkur tadqiqot ishi kirish, 3 ta bob, xulosa, foydalanilgan adabiyotlar ro‘yxatidan iborat. Ilmiy tadqiqot ishining umumiy hajmi 97 sahifani tashkil qiladi.

I BOB. INTERNET MAKONIDA VEB-SAYTLAR XAVFSIZLIGI MUAMMOLARI

1.1. Xavfsiz veb-saytlarning tuzilishi arxitekturalari, xavfsizlikni ta’minlashdagi muammolar va ularni hal qilish usullari

Sayt yoki web-sayt inglizcha website: web – “o‘rgimchak to‘ri, tarmoq” va site – “joy, segment, tarmoqning bir qismi” degan ma’nolarni anglatadi. Sayt – bu bitta domen nomidan foydalanadigan va bir-biriga bog‘langan web-sahifalar to‘plamidir. Web-saytlar shaxs, guruh, korxona yoki tashkilot tomonidan turli maqsadlarda yaratilishi mumkin. Veb-saytlar HTML, CSS va JavaScript kabi turli veb-ishlab chiqish texnologiyalari va dasturlash tillari yordamida yaratiladi. Ushbu texnologiyalar veb-saytning tuzilishi, dizayni va funksionalligini yaratish uchun ishlatiladi. Barcha ommaviy veb saytlar butun dunyodagi internet to‘rini tashkil qiladi.

Dunyodagi birinchi web sayt info.cern.ch sayti 6-avgust 1991-yilda paydo bo‘ldi. Uning yaratuvchisi, Tim Berners-Li, HTTP aloqa protokolida, URI adreslash tizimi va gipermatnli belgilash tili HTML asoslangan, yangi texnologiyalar World

Wide Web ustida u tavsifi chop etildi. Veb-saytga kirish uchun foydalanuvchilar veb-brauzeriga uning URL manzilini (Uniform Resource Locator) kiritishlari kerak. Keyin veb-brauzer veb-saytni joylashtiradigan serverga so‘rov yuboradi, u so‘ralgan veb-sahifalarni foydalanuvchi brauzeriga qaytarib yuborish orqali javob beradi. Saytda shuningdek, serverlar va brauzerlarning o‘rnatilishi va ishlash tamoyillari tasvirlangan. Bu sayt dunyoning birinchi internet-katalogiga aylandi, chunki Tim Berners-Li keyinchalik boshqa saytlarga havolalar ro‘yhatini qo‘ydi. Birinchi sayt ishlashi uchun zarur bo‘lgan barcha vositalarini Berners-Li, hatto 1990 - yilda tayyorlab qo‘ygan edi, birinchi web-muharriri, birinchi server NeXTcube asoslangan va birinchi web-sahifasi bilan birinchi gipermatnli brauzer WorldWideWeb paydo bo‘ldi.

Saytlarni yaratish hozirgi vaqtda qiyin emas. Chunki kodni boshida yozib, saytning har bir detalini yaratishning hojati yo‘q. Hozirda internet saytlarida eng ko‘p ishlatilayotgan WordPress yoki Joomla kabi sayt platformalari sayt yaratishdagi ishning 99% qismini foydalanuvchi uchun bajaradi. Buning uchun WordPress yoki Joomla web dasturlarini hostinginga yuklash va ularni o‘rnatish kerak. WordPress va Joomla uchun saytlarning tayyor ko‘rinishlari ham internetda yuklab olish uchun juda ko‘plab topiladi[3].

Saytlarning mashhurligi, odatda, mehmonlar soni bo‘yicha aniqlanadi. Quyida Alexa saytlar va shunga o‘xshash web-saytlar ro‘yhatlari mavjud. Bu saytlar boshqa saytlarning ishtiroki haqida statistik to‘plangan saytlar. Top reyting Alexa to‘plamini o‘rnatgan foydalanuvchilarning to‘g‘ridan-to‘g‘ri ma’lumotlarini to‘playdi.

Natijada ushbu ma’lumotlarga asoslanib, sayt trafigi va tegishli havolalar haqida statistik ma’lumotlar to‘planadi.

Quyidagi 1.1.1-jadvalda 2022-yilda eng ko‘p tashrif buyurilgan saytlar ro‘yhati keltirilgan.

1.1 .1.jadval

Ommabop saytlat ro‘yhati

T/r	O‘zbekiston bo‘yicha	Dunyo bo‘yicha
1	Yandex.uz	Google.com
2	Google.uz	Youtube.com
3	OLX.uz	Facebook.com
4	Telegram.org	Instagram.com
5	Lex.uz	Baidu.com
6	Utube.uz	Wikipedia.org
7	Uzjobs.uz	Yahoo.com
8	Sputniknews.com	Amazon.com
9	Kinoprofi.org	Twitter.com
10	Sport.uz	Reddit.com

Misol uchun O‘zbekistondagi veb-saytlarda SimilarWeb saytining qayt etishicha tashrif buyuruvchilarning soni quyidagicha:

1. Yandex.uz: 14,2 million tashrif;
2. Google.uz: 9,9 million tashrif;
3. OLX.uz: 4,8 million tashrif;
4. Telegram.org: 4,4 million tashrif;
5. Sputniknews.com: 3,4 million tashrif.

Web sayt 2 qismdan iborat:

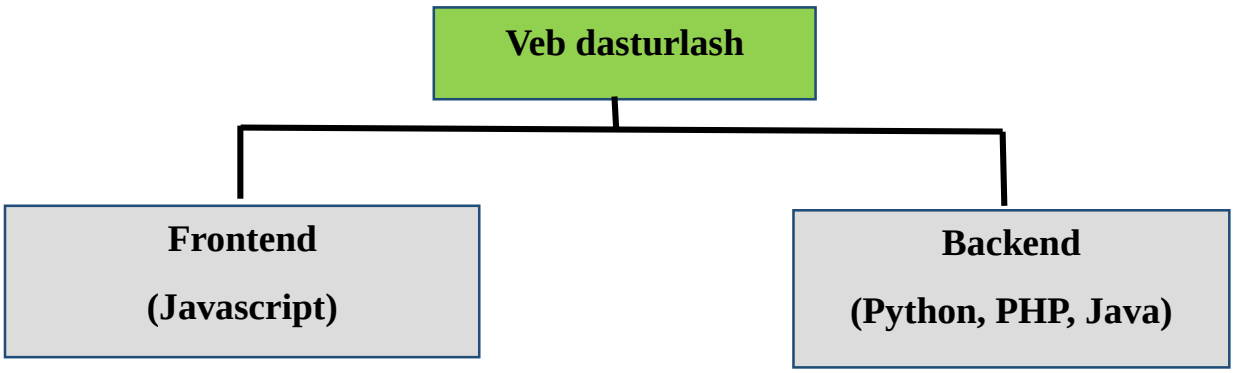
1) Frontend;
2) Backend.

Backend va *frontend* veb-ilovaning ikkita asosiy qismini tavsiflovchi atamalar bo‘lib, ularning har biri o‘ziga xos mas’uliyat va texnologiyalar to‘plamiga ega.

Backend veb-ilovaning server tomoniga ishora qiladi. U veb-serverda

ishlaydigan, veb-ilovani quvvatlaydigan server va dasturning mijoz tomoni o‘rtasidagi aloqani boshqaradigan koddan iborat. Backend odatda Python, PHP, Ruby va Java kabi server tomonidagi dasturlash tillaridan iborat bo‘lib, ular veb-ilovaning server tomoni mantiqini yaratish, ma’lumotlar bazalarini boshqarish va server tomoni skriptlarini boshqarish uchun ishlatiladi. Backend ishlab chiquvchilari veb-ilovalarning server tomoni komponentlarini, shu jumladan veb-server, ma’lumotlar bazasi va server tomonidagi ilovalarni yaratish va saqlash uchun javobgardir.

Boshqa tomondan, *frontend* veb-ilovaning mijoz tomoniga ishora qiladi. Bu veb-ilovaning foydalanuvchilar bilan bevosita muloqot qiladigan qismi. Frontend odatda HTML, CSS va JavaScript kabi veb-texnologiyalardan iborat bo‘lib, ular foydalanuvchi interfeysini yaratish, foydalanuvchi o‘zaro aloqalarini boshqarish va dasturning server tomoni bilan aloqa qilish uchun ishlatiladi. Frontend ishlab chiquvchilari veb-ilovalarning mijoz tomoni komponentlarini, jumladan foydalanuvchi interfeysi, foydalanuvchi tajribasi va front-end ilovalarini yaratish va saqlash uchun javobgardir. Frontend veb-ilovaning mijoz tomoni bo‘lib, foydalanuvchi interfeysi, foydalanuvchi tajribasi va server tomoni bilan aloqani boshqarish uchun javobgardir.ko‘rinib turgan qismi Front-End ga kiradi. Soddaroq qilib, restoran misolida.



1.1.1-rasm. Veb-dasturlash tillarining turlari

Backend dasturchisi - bu veb-ilovalar, ma’lumotlar bazalari va dasturiy ta’minot tizimlarining server tomoni komponentlarini yaratish va saqlashga

qaratilgan dasturiy ta’minot muhandisining bir turi. Ular Java, Python, Ruby va PHP kabi dasturlash tillari bilan ishlaydi, server tomoni mantig‘ini yaratadi, API (Application Programming Interfaces) ishlab chiqadi, ma’lumotlar bazalarini boshqaradi va server tomonidagi skriptlarni boshqaradi.

– backend dasturchining ba’zi mas’uliyatlariga quyidagilar kiradi;

– server tomonidagi ilovalar va APIlarni ishlab chiqish va saqlash;

– front-end funksiyalarini orqa tomon mantiqi bilan birlashtirish;

– ma’lumotlar bazasi so‘rovlari va sxemalarini yozish va optimallashtirish;

– PHP, Python va Ruby kabi server tomonidagi skript tillarini boshqarish;

– server tomonidagi ilovalarning miqyosi va ishlashini ta’minlash;

– yuqori sifatli dasturiy mahsulotlarni yetkazib berish uchun oldingi dasturchilar, UX dizaynerlari va loyiha menejerlari bilan hamkorlik qilish.

Backend dasturchi veb-ilovalarning mijoz tomoni komponentlarini yaratish uchun mas’ul bo‘lgan front-end dasturchilar bilan hamkorlikda ishlaydi. Ular birgalikda veb-ilovaning to‘liq to‘plamini yaratadilar.

Frontend dasturchi - bu veb-ilovalarning mijoz tomoni komponentlarini yaratish va saqlashga qaratilgan dasturiy ta’minot muhandisining bir turi. Ular foydalanuvchi interfeysini loyihalash va rivojlantirish, foydalanuvchilarning o‘zaro aloqalarini amalga oshirish va veb-sayt yoki ilovaning sezgir va ulardan foydalanish osonligini ta’minlash uchun javobgardir.

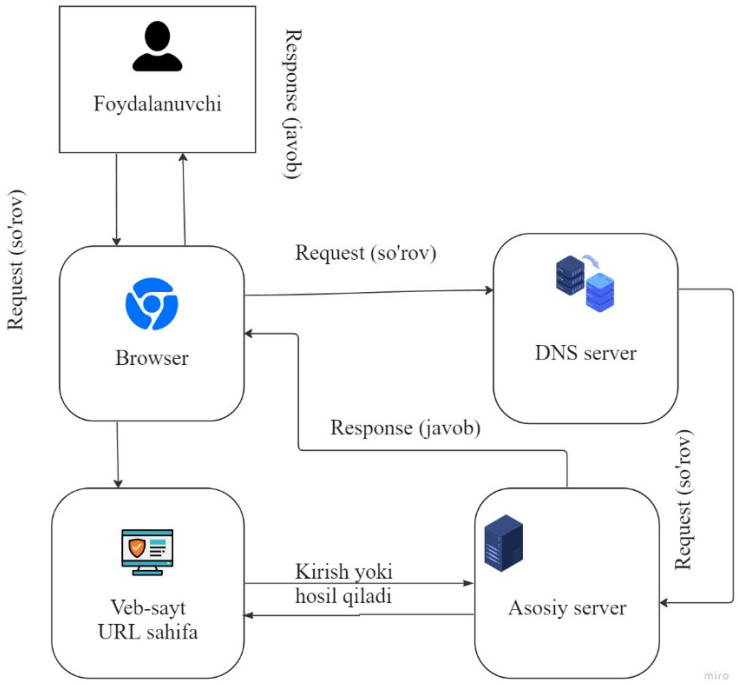
Keling endi veb-sayt qayday ishlashini bir korib chiqsak. Agar foydalanuvchi o‘z brauzeriga veb-sayt URL-manzilini kiritsa, brauzer domen nomlari tizimi (DNS) serveriga veb-saytning inson o‘qiy oladigan domen nomini IP-manzilga tarjima qilish uchun so‘rov yuboradi, bu har bir ulangan qurilmaga tayinlangan noyob raqamli identifikatordir.

DNS-server domen nomi bilan bogliq bo‘lgan IP-manzilni qidiradi va uni foydalanuvchi brauzeriga qaytaradi, so‘ngra veb-saytni joylashtirgan veb-server bilan aloqa o‘rnatish uchun IP-manzildan foydalanadi. Keyin veb-server so‘ralgan veb-sahifani foydalanuvchining brauzeriga yuboradi, u o‘z ekranida ko‘rsatadi.

DNS-serverlar internet uchun o‘ziga xos telefon kitobi vazifasini o‘taydi va foydalanuvchilarga o‘zlari tashrif buyurmoqchi bo‘lgan har bir sayt bilan bog‘langan IP manzillarni eslab qolish o‘rniga inson o‘qiy oladigan domen nomlaridan foydalangan holda veb-saytlarga kirish imkonini beradi. DNS Internet qanday ishlashining muhim qismidir va usiz foydalanuvchilar o‘zlarining IP manzillari yordamida veb-saytlarga kirishlari kerak bo‘ladi, bu esa ancha qulayroq va intuitiv bo‘ladi.

DNS taqsimlangan tizim bo‘lib, u domen nomini aniqlashni boshqarish uchun butun dunyo bo‘ylab joylashgan serverlar tarmogiga tayanadi. Agar foydalanuvchi brauzeri DNS serveriga so‘rov yuborsa, u server keshda so‘ralgan domen nomi uchun IP manzilga ega bo‘lmasligi mumkin. Bunday holda, server so‘rovni DNS ierarxiyasida yuqoriroq bo‘lgan va so‘rovni bajarish uchun yaxshiroq jihozlangan boshqa DNS serveriga yo‘naltiradi.

Bu jarayon so‘rov keshda so‘ralgan domen nomi uchun IP-manzilga ega bo‘lgan yoki uni qidirib, uni asl DNS serveriga qaytara oladigan DNS serveriga yetguncha davom etadi. DNS ierarxiyasi turli darajalarda tashkil etilgan, ildiz domeni tepada, undan keyin .com, .org va .net kabi yuqori darajali domenlar (TLD) va keyin example.com kabi ikkinchi darajali domenlar.



1.1.2-rasm. Veb-saytga murojaatning ishlash sxemasi

Veb-sayt yaratilganda, egasi odatda domen nomlarini tayinlashni boshqaruvchi kompaniya bo‘lgan registratorda domen nomini ro‘yxatdan o‘tkazadi. Ro‘yxatga oluvchi domen nomini egasiga tayinlaydi va DNS tizimida veb-saytni joylashtirgan serverning domen nomi va IP manzili o‘rtasidagi bog‘lanishni qayd qiladi. DNS foydalanuvchilarning brauzerlari va veb-serverlari o‘rtasida ulanishlarni o‘rnatish uchun ishlatilishi mumkin bo‘lgan IP-manzillarga inson o‘qiy oladigan domen nomlarini tarjima qilish orqali veb-saytlar qanday ishlashining muhim qismidir. Bu foydalanuvchilarga tanish domen nomlaridan foydalangan holda veb-saytlarga kirish imkonini beradi va internetning taqsimlangan tizim sifatida ishlashiga imkon beradi.

Bugungi kunda xavfsiz veb-saytlarni yaratish har qanday CIO yoki CTO uchun asosiy tashvishlardan biridir. Tarmoq ma’murlari, IT menejerlari, dasturiy ta’minot arxitektorlari va veb-ishlab chiquvchilar xavfsiz veb-saytlarni yaratish va saqlash uchun javobgardir. Xavfsiz veb-sayt tuzilish uchun quyidagi qoidalarga amal qilishimiz kerak.

Ma’lumotlarni shifrlash (Encrypt data) - ma’lumotlar xavfsizligi veb-sayt xavfsizligining eng muhim jihati hisoblanadi. Ma’lumotlar bazalarida saqlanadigan ma’lumotlarning aksariyati oddiy va ochiqdir. Ma’lumotlarning aksariyati oddiy saqlanishi mumkin bo‘lsa-da, maxfiy ma’lumotlar ma’lumotlar bazasida shifrlangan bo‘lishi kerak.

Shifrlanishi kerak bo‘lgan ba’zi umumiy ma’lumotlarga foydalanuvchi identifikatorlari, elektron pochta xabarlari, parollar, ijtimoiy xavfsizlik raqamlari, tug‘ilgan sanasi, kredit karta ma’lumotlari, parol bo‘yicha maslahatlar, shaxsiy sog‘liq yozuvlari, shaxsiy suhbatlar va xabarlar, moliyaviy yozuvlar va bank ma’lumotlari kiradi.

Buning ustiga, siz parollar, kredit karta ma’lumotlari, ijtimoiy xavfsizlik va qimmatli deb hisoblagan boshqa narsalar kabi eng nozik ma’lumotlarga ikki marta shifrlashni qo‘llashingiz mumkin. Parol va boshqa nozik ma’lumotlarni shifrlash uchun xeshlash tavsiya etiladi.

Veb-Saytlardagi aloqani shirflash - HTTPS (Hypertext Transfer Protocol Secure) yordamida veb-saytlarni shifrlash onlayn aloqani ta’minlash va maxfiy ma’lumotlarni himoya qilishda muhim qadamdir. HTTPS internet orqali xavfsiz aloqa uchun protokol bo‘lib, u foydalanuvchi brauzeri va veb-sayt o‘rtasida uzatiladigan ma’lumotlarni himoya qilish uchun shifrlashdan foydalanadi.

HTTPS oddiy HTTP protokoli ustiga SSL/TLS shifrlash qatlamini qo‘shish orqali ishlaydi, ya’ni foydalanuvchi brauzeri va veb-sayt o‘rtasida uzatiladigan barcha ma’lumotlar shifrlangan va uchinchi shaxslar tomonidan tutib bo‘lmaydi. Bu parollar, kredit karta ma’lumotlari yoki shaxsiy ma’lumotlar kabi nozik ma’lumotlar uzatilayotganda ayniqsa muhimdir.

Veb-sayt HTTPS dan foydalansa, URL “http://” o‘rniga “https://” bilan boshlanadi. Brauzer manzil satrida ulanish xavfsiz ekanligini ko‘rsatuvchi qulf belgisi ham bo‘ladi.

HTTPS bilan veb-saytlarni shifrlash tobora muhim ahamiyat kasb etmoqda, chunki kiber tahdidlar yanada murakkab va keng tarqalgan. Bu koʻplab veb-saytlar, xususan, maxfiy ma’lumotlarni qayta ishlovchi yoki foydalanuvchilarning tizimga kirishini talab qiladigan veb-saytlar uchun ham talabga aylanib bormoqda. Aslida, koʻplab veb-brauzerlar HTTPS bilan himoyalanmagan veb-saytga tashrif buyurganlarida, foydalanuvchilarga ulanishning uzilishini koʻrsatuvchi ogohlantirishlarni koʻrsatadi. Xavfsiz emas va ularning ma’lumotlari xavf ostida boʻlishi mumkin.

SQL injectionlardan himoyalanish - bu SQL soʻrovlariga zararli kodni kiritish orqali veb-ilovalarga qaratilgan xavfsizlik tahdidining bir turi. AOK qilingan kod maxfiy ma’lumotlarni oʻgirlash, ma’lumotlarni oʻzgartirish yoki oʻchirish yoki boshqa ruxsatsiz harakatlarni amalga oshirish uchun ishlatilishi mumkin. SQL in’ektsion hujumlarini oldini olishning ba’zi usullari:

Parametrlangan soʻrovlardan foydalaning: satrlarni birlashtirish orqali SQL soʻrovlarini qurish oʻrniga, SQL kodini foydalanuvchi kiritishidan ajratib turadigan parametrlangan soʻrovlardan foydalaning. Bu soʻrovga zararli kod kiritilishining oldini oladi. Foydalanuvchi kiritishini tozalash SQL soʻrovlarida foydalanishdan oldin barcha foydalanuvchi kiritgan ma’lumotlarni tekshiring va tozalang. Bunga maxsus belgilarni olib tashlash va ma’lum ma’lumotlar turlariga kirishni cheklash kiradi.

Tayyorlangan bayonotlardan foydalaning: Tayyorlangan bayonotlar SQL soʻrovlarini oldindan kompilyatsiya qilish va soʻrovni ma’lumotlardan ajratish uchun ishlatilishi mumkin. Bu tajovuzkorlar uchun soʻrovga zararli kodni kiritishni qiyinlashtiradi.

Ma’lumotlar bazasi imtiyozlarini cheklash: Ma’lumotlar bazasi foydalanuvchisining imtiyozlarini faqat zarur boʻlganlar bilan cheklang. Bu tajovuzkorlarning maxfiy ma’lumotlarni oʻzgartirishi yoki oʻchirishini oldini olishga

yordam beradi.

Veb-ilovaning xavfsizlik devoridan (WAF) foydalaning: WAF kiruvchi trafikni filtrlash va SQL in’ektsiyasi zaifliklaridan foydalanishga urinayotgan zararli so‘rovlarni bloklash uchun ishlatilishi mumkin.

Ushbu chora-tadbirlarni amalga oshirish orqali siz veb-ilovalaringizni va ma’lumotlar bazalaringizni SQL in’ektsion hujumlaridan himoya qilishingiz mumkin.

1.1.2.jadval

Veb-saytlarda uchraydigan zaifliklar ro‘yhati

T/r	Nomlanishi	Ta’rifi
1	Cross Site Scripting(XSS)	Saytlararo skriptlar.
2	SQL in’ektsiyasi	Veb-sayt ma’lumotlar bazasini tashqi SQL buyruqlariga bardoshsizligi.
3	Cross-Site Request Forgery(CSRF)	Autentifikatsiya qilingan foydalanuvchilarni veb-saytga so‘rov yuborishga majburlovchi hujumdir.
4	Ishonchsiz aloqa kanallari	Shifrlashdan foydalanmaydigan veb-saytlar.

Veb-server xavfsizligini saqlash- Ruxsatsiz kirish, ma’lumotlar buzilishi va boshqa xavfsizlik muammolarini oldini olish uchun veb-serverni xavfsiz saqlash juda muhimdir. Veb-serveringizni xavfsiz saqlash uchun quyidagi qadamlarni bajarishingiz mumkin:

– dasturiy ta’minotni yangilab turing: Serveringizda ishlaydigan barcha dasturiy ta’minot, jumladan, operatsion tizim, veb-server dasturiy ta’minoti va har qanday uchinchi tomon ilovalari muntazam yangilanib turishiga ishonch hosil qiling. Yangilanishlar ko‘pincha zaifliklarni bartaraf etadigan xavfsizlik tuzatishlarini o‘z ichiga oladi;

– kuchli parollardan foydalaning: Serverdagi barcha foydalanuvchi hisoblarida taxmin qilish yoki buzish qiyin bo‘lgan kuchli parollar mavjudligiga ishonch hosil

qiling. Xavfsiz parollarni yaratish va saqlash uchun parol menejeridan foydalanishni o‘ylab ko‘ring;

– kirishni cheklash: Serveringizga kirishni faqat unga muhtoj bo‘lganlar uchun cheklang. Turli foydalanuvchilar va xizmatlar uchun cheklangan imtiyozlarga ega alohida foydalanuvchi hisoblarini yarating. Foydalanilmayotgan xizmatlar va portlarni o‘chirib qo‘ying;

– xavfsizlik devoridan foydalaning: xavfsizlik devori ruxsatsiz trafikni serveringizga kirishini bloklashi mumkin. Xavfsizlik devoringizni faqat serveringiz ishlashi uchun zarur bo‘lgan trafikka ruxsat berish uchun sozlang;

– ma’lumotlarni shifrlash: Kirish hisob ma’lumotlari va kredit karta ma’lumotlari kabi nozik ma’lumotlarni himoya qilish uchun shifrlashdan foydalaning. HTTPS, SSL va TLS shifrlash protokollari bo‘lib, ular veb-trafikni himoya qilish uchun ishlatilishi mumkin;

– ma’lumotlarni muntazam ravishda zahiralash: Buzgunchilik yoki tizimdagi nosozliklar tufayli ma’lumotlarni yo‘qotishdan himoya qilish uchun server ma’lumotlarining muntazam zahira nusxalarini yarating va saqlang;

– xavfsizlik vositalarini qo‘llang: Xavfsizlik tahdidlarini aniqlash va oldini olishga yordam berish uchun hujumlarni aniqlash tizimlari (IDS), antivirus dasturlari va zararli dasturlar skanerlari kabi xavfsizlik vositalarini qo‘llang.

Veb-server xavfsizligini ta’minlash texnik va ma’muriy choralarning kombinatsiyasini talab qiladi. Server xavfsizligi bo‘yicha eng yaxshi amaliyotlarga rioya qilish orqali siz serveringizni xavfsizlik tahdidlaridan himoya qilishingiz va foydalanuvchilaringiz ma’lumotlarini xavfsiz saqlashingiz mumkin.

1

Bugungi raqamli dunyoda veb-saytlar kundalik hayotimizning ajralmas qismiga aylandi. Elektron tijoratdan onlayn-bankgacha, biz shaxsiy va professional

2

vazifalarimizni bajarish uchun koʻp veb-saytlarga tayanamiz. Biroq, veb-saytlarga boʻlgan ishonch ortib borishi bilan kiberxavfsizlik xavflari ham kattaligi va murakkabligi bilan oʻsdi. Kiberjinoyatchilar doimiy ravishda maxfiy ma’lumotlarni oʻgirlash, operatsiyalarni buzish va moliyaviy zarar etkazish uchun veb-sayt zaifliklaridan foydalanish yoʻllarini izlaydilar. Ushbu boʻlimda biz veb-saytlarda duch keladigan umumiy kiberxavfsizlik muammolarini va ushbu xavflarni kamaytirish uchun amalga oshirilishi mumkin boʻlgan yechimlarni muhokama qilamiz.

Veb-saytlar duch keladigan eng keng tarqalgan kiberxavfsizlik muammolaridan biri saytlararo skriptdir (XSS- Cross Site Scripting). XSS tajovuzkor veb-sahifaga zararli kodni kiritganda paydo boʻladi, keyin esa shubhali foydalanuvchilar tomonidan bajariladi. Bu tajovuzkorga parollar yoki kredit karta raqamlari kabi nozik ma’lumotlarni oʻgʻirlash imkonini berishi mumkin. Ushbu muammoning yechimi barcha foydalanuvchi kiritgan ma’lumotlarning tozalanganligini ta’minlash uchun kirish tekshiruvidan foydalanish va zararli kodni kiritishning oldini olish uchun chiqish kodlashidan foydalanishdir.

Veb-saytlar duch keladigan yana bir kiberxavfsizlik muammosi SQL in’ektsiyasidir. SQL in’ektsiyasi tajovuzkor veb-sayt ma’lumotlar bazasini boshqarish uchun SQL buyruqlaridan foydalanganda sodir boʻladi. Bu tajovuzkorga maxfiy ma’lumotlarga kirishga yoki hatto veb-saytni boshqarishga imkon beradi. Ushbu muammoni hal qilish SQL in’ektsion hujumlarini oldini olish uchun tayyorlangan bayonotlar yoki parametrlangan soʻrovlardan foydalanishdir.

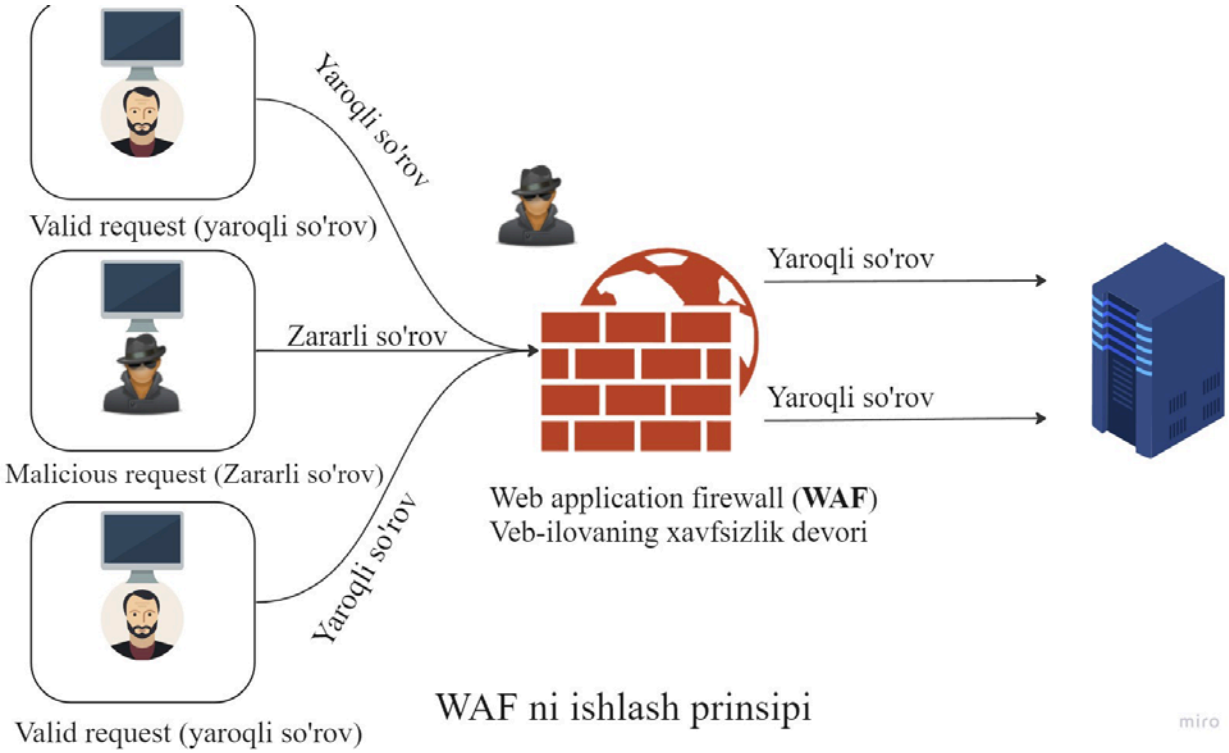
Saytlararo soʻrovlarni qalbakilashtirish (CSRF - Cross-Site Request Forgery) veb-saytlar duch keladigan yana bir kiberxavfsizlik muammosidir. CSRF tajovuzkor autentifikatsiya qilingan foydalanuvchilarni hozirda autentifikatsiya qilingan veb-ilovaga soʻrov yuborishga majburlovchi hujumdir. Bu tajovuzkorga maxfiy ma’lumotlarni oʻgirlash yoki veb-sayt nazoratini qoʻlga kiritish imkonini beradi.

Ushbu muammoning yechimi foydalanuvchi so‘rovlarining haqiqiyligini tekshirish uchun CSRF tokenlaridan foydalanishdir.

Zararli dastur veb-saytga zarar etkazishi va tashrif buyuruvchilar qurilmalariga tarqalishi, ularning tizimlariga zarar yetkazishi yoki ma’lumotlarini o‘girlashi mumkin bo‘lgan yana bir kiberxavfsizlik muammosidir. Ushbu muammoning yechimi zararli dasturlarni skanerlash uchun antivirus dasturidan foydalanish va ruxsatsiz kirishni oldini olish uchun xavfsizlik devorlarini ishlatishdir.

Zaif parollar tajovuzkorlarning veb-sayt va uning ma’lumotlariga kirishini osonlashtiradigan yana bir kiberxavfsizlik muammosidir. Kuchli parollardan foydalanish va foydalanuvchilardan kuchli parollarni tanlashni talab qiladigan parol siyosatini qo‘llash muhim. Qo‘shimcha xavfsizlik qatlamini qo‘shish uchun ikki faktorli autentifikatsiyadan foydalanish ham muhimdir.

Veb-saytlarda kiberxavfsizlikni ta’minlashning bir usuli bu veb-ilovalar xavfsizlik devorini (WAF) amalga oshirishdir. WAF kiruvchi trafikni filtrlash va zararli so‘rovlarni bloklash orqali SQL in’ektsiyasi va saytlararo skript kabi hujumlarning oldini olishga yordam beradi. WAF shuningdek, kiruvchi trafikni kuzatishi va qayd qilishi mumkin, bu potentsial hujumlarni aniqlashda foydali bo‘lishi mumkin.



1.2.1-rasm. WAF ni ishlash prinsipi

Ishonchsiz aloqa veb-saytlar duch keladigan yana bir kiberxavfsizlik muammosidir. Shifrlashdan foydalanmaydigan veb-saytlar tajovuzkorlarga login hisob ma’lumotlari yoki kredit karta raqamlari kabi nozik ma’lumotlarni tutib olishiga ruxsat berishi mumkin. Aloqalarni himoya qilish uchun HTTPS shifrlashdan foydalanish muhimdir.

Keling endi Python dasturlash tilidan foydalanib veb-ilovalar xavfsizlik devorini (WAF) yozishni korib chiqamiz

WAF_In_Python.py

```python
import re

# Oldindan belgilangan xavfsizlik qoidalari
rules = {
    "SQL Injection": r"(union|select|from|where|drop|delete)",
    "Cross-Site Scripting (XSS)": r"(<script>|javascript:)"
}

# Kiruvchi so'rov
request = "SELECT * FROM users"

# Xavfsizlik qoidalari asosida kiruvchi so'rovni filtrlang
for rule_name, rule_pattern in rules.items():
    if re.search(rule_pattern, request, re.IGNORECASE):
        print(f"So'rov {rule_name} hujumiga uchraydi shuning uchun rad etildi")
        break
else:
    print("So'rov amalga oshirildi")
```

Snipped

1.2.2-rasm. Sodda WAF tizimi.

```
PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE
[Running] python -u "e:\Masalalar\WAF_In_Python.py"
So'rov SQL Injection hujumiga uchraydi shuning uchun rad etildi

[Done] exited with code=0 in 0.186 seconds
```

1.2.3-rasm. Yaratilgan WAF orqali web-sayt xavfsizligini tekshirish.

Ushbu dastur kodimiz veb-saytimiz qabul qiladigan so‘rovlarni filter qilishda ishlatiladi.

Keling endi buni yanada mukammallashtiramiz. Endigi maqsad veb-saytimiz qabul qiladigan har bir so‘rov (request) larni filterlab chiqamiz agar XSS yoki SQL injection hujum turi aniqlansa dastur kodimiz foydalanuvchiga javob qaytarmasin va ushbu IP addresdagi foydalanuvchini qora royhatga joylab qo‘ysin. Uning uchun Python dasturlash tilining Django frameworkindan foydalamiz. Bunda so‘rovlarni

dasturga yetib borishdan oldin filter qiladigan qismi middleware da biznes logikani yozamiz.

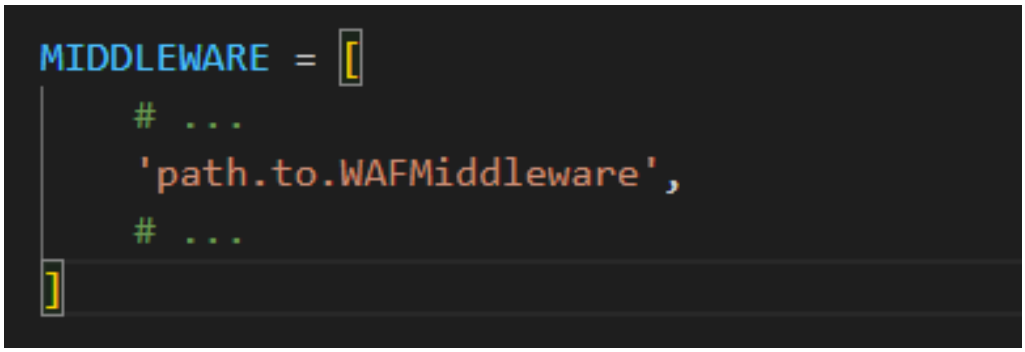
Dasturlash tillarida *middleware* - bu veb-server va veb-ilova o‘rtasida joylashgan dasturiy ta’minot komponenti bo‘lib, kiruvchi HTTP/HTTPS so‘rovlari va chiquvchi javoblarni o‘zgartirishga imkon beradi. O‘rta dastur odatda veb-ramkalarda autentifikatsiya, ro‘yxatga olish va xavfsizlik kabi barcha so‘rovlarga tegishli bo‘lgan o‘zaro bog‘liq muammolarni amalga oshirish uchun ishlatiladi.

middleware_with_WAF

```python
import re
from django.http import HttpResponse

class WAFMiddleware:
    def __init__(self, get_response):
        self.get_response = get_response

        # Oldindan belgilangan xavfsizlik qoidalari
        self.rules = {
            "SQL Injection": r"(union|select|from|where|drop|delete)",
            "Cross-Site Scripting (XSS)": r"(<script>|javascript:)"
        }

    def __call__(self, request):
        # Xavfsizlik qoidalari asosida kiruvchi so'rovni filtrlang
        for rule_name, rule_pattern in self.rules.items():
            if re.search(rule_pattern, request.path, re.IGNORECASE):
                return HttpResponse(f"Blocked request due to {rule_name} attack")
        # Keyingi o'rta dastur yoki ko'rish funktsiyasini chaqiring
        response = self.get_response(request)
        return response

```

Snipped

1.2.4-rasm. Mukammal WAF ni yaratish.



1.2.5-ram. WAF ni o‘rta dasturlar qatoriga qo‘shish.

Yaratgan middleware ni settingsdagi qolgan middlewarelar qatoriga qo‘shib qo‘yamiz.

Xodimlarni o‘qitish va o‘qitish kiberxavfsizlik hodisalarining oldini olishda ham muhim ahamiyatga ega. Xodimlar parollarni boshqarish, fishingdan xabardorlik va ma’lumotlar bilan ishlash tartib-qoidalari bo‘yicha eng yaxshi amaliyotlar bo‘yicha o‘qitilishi kerak. Shuningdek, ular kiberxavfsizlikning ahamiyati va xavfsizlik buzilishining mumkin bo‘lgan oqibatlari haqida ma’lumot olishlari kerak.

Ushbu chora-tadbirlarga qo‘shimcha ravishda, kiberxavfsizlik hodisasi sodir bo‘lgan taqdirda javob rejasiga ega bo‘lish ham muhimdir. Javob rejasi hodisaning oldini olish, sabablarini o‘rganish va manfaatdor tomonlar bilan muloqot qilish uchun qadamlarni o‘z ichiga olishi kerak.

Biroq, kirishni tekshirish, WAF’lar, muntazam zaxira nusxalari, xodimlarni o‘qitish va o‘qitish javobni rejalashtirish kabi tegishli xavfsizlik choralarini qo‘llash orqali xavflarni kamaytirish mumkin. Veb-sayt egalari va ishlab chiquvchilari kiberxavfsizlikka jiddiy yondashishlari va rivojlanayotgan tahdidlarga nisbatan hushyor bo‘lishlari juda muhimdir. Faqat birgalikda ishlash va egri chiziqdan oldinda bo‘lish orqali biz veb-saytlarimiz va ularga ishonadigan foydalanuvchilarning xavfsizligi va maxfiyligini ta’minlay olamiz.

Shuni ta’kidlash kerakki, kiberxavfsizlik bir martalik tuzatish emas, balki doimiy monitoring, sinov va takomillashtirishni talab qiladigan doimiy jarayondir. Xavfsizlik choralari muntazam ravishda yangilanishi va o‘zgaruvchan tahdid landshaftiga moslashtirilishi kerak. Xavfsizlik siyosatlari, protseduralari va amaliyotlarini ularning samarali va dolzarb bo‘lishini ta’minlash uchun muntazam ravishda ko‘rib chiqish va yangilab turish juda muhimdir.

Veb-sayt kiberxavfsizligining yana bir muhim jihati bu qoidalar va standartlarga rioya qilishdir. Sanoatga qarab, amal qilinishi kerak bo‘lgan turli xil qoidalar va standartlar bo‘lishi mumkin, masalan, Evropa Ittifoqidagi umumiy ma’lumotlarni himoya qilish to‘grisidagi nizom (GDPR) yoki kredit karta ma’lumotlari bilan ishlaydigan tashkilotlar uchun to‘lov kartalari sanoati ma’lumotlar xavfsizligi standarti (PCI DSS). Ushbu qoidalar va standartlarga rioya qilish nafaqat qonuniy muvofiqlikni ta’minlaydi, balki kiberxavfsizlik choralarini kuchaytirishga yordam beradi.

GDPR (General Data Protection Regulation) - umumiy ma’lumotlarni himoya qilish qoidalarini anglatadi. Bu 2018-yil 25-mayda kuchga kirgan Yevropa Ittifoqining (Yevropa Ittifoqi) reglamentidir. GDPR kompaniya va tashkilotlar Yevropa Ittifoqi fuqarolarining shaxsiy ma’lumotlarini qanday himoya qilishi kerakligi haqidagi qoidalarni belgilaydi. Bu shaxslarga shaxsiy ma’lumotlari ustidan ko‘proq nazorat qilish imkonini beradi va kompaniyalardan ma’lumotlar amaliyoti bo‘yicha shaffof bo‘lishini talab qiladi.

1.2.6-rasm. *GDPR* - umumiy ma’lumotlarni himoya qilish qoidalari.

GDPRga muvofiq, shaxsiy ma’lumotlar shaxsni aniqlash uchun ishlatilishi mumkin bo‘lgan ism, manzil, elektron pochta manzili yoki IP manzili kabi har qanday ma’lumotlarni o‘z ichiga oladi. Nizom kompaniyaning qayerda joylashganidan qat’i nazar, Yevropa Ittifoqi fuqarolarining shaxsiy ma’lumotlarini qayta ishlovchi har qanday kompaniya yoki tashkilotga nisbatan qo‘llaniladi.

GDPRning asosiy talablaridan ba’zilari quyidagilardan iborat:

– shaxsiy ma’lumotlardan foydalanish uchun jismoniy shaxslardan aniq rozilik olish;

– jismoniy shaxslarga shaxsiy ma’lumotlariga kirish va ularni o‘chirish huquqini ta’minlash;

– shaxsiy ma’lumotlarni himoya qilish uchun tegishli xavfsizlik choralarini qo‘llash;

– 72 soat ichida ma’lumotlarning buzilishi haqida hokimiyatga va zarar ko‘rgan shaxslarga xabar berish;

– ma’lumotlarni himoya qilish bo‘yicha ishlarni nazorat qilish uchun ma’lumotlarni himoya qilish bo‘yicha xodimni (DPO) tayinlash.

GDPRga rioya qilmaslik 20 million yevro yoki kompaniyaning global yillik daromadining 4 foizigacha, qaysi biri kattaroq bo‘lsa, katta miqdorda jarimaga olib kelishi mumkin.

PCI DSS (Payment Card Industry Data Security Standard.) - to‘lov kartalari sanoati ma’lumotlar xavfsizligi standartini anglatadi. Bu Visa, Mastercard, American Express va Discover kabi yirik kredit karta kompaniyalari tomonidan to‘lov kartalari bilan operatsiyalarni amalga oshirishda karta egasining nozik ma’lumotlarini himoya qilishni ta’minlash uchun yaratilgan xavfsizlik standartlari to‘plamidir.

1.2.7-rasm. Yevropa to‘lov tizimini nazorat standarti.

PCI DSS to‘lov kartasi tranzaksiyalarini qabul qiladigan yoki qayta ishlovchi har qanday tashkilot uchun amal qiladi. Ushbu kredit karta kompaniyalari PCI DSSga rioya qilishni talab qiladi va ularga rioya qilmaslik jarima va boshqa jazolarga olib kelishi mumkin.

PCI DSS tashkilotlar rioya qilishi kerak bo‘lgan 12 ta talabdan iborat, jumladan:

1. karta egasi ma’lumotlarini himoya qilish uchun xavfsizlik devori konfiguratsiyasini o‘rnating va saqlang;

2. standart parollar yoki xavfsizlik parametrlaridan foydalanmang;

3. saqlangan karta egasi ma’lumotlarini himoya qiling;

4. ochiq, umumiy tarmoqlarda karta egasi ma’lumotlarini shifrlash;

5. antivirus yoki dasturlardan foydalaning va muntazam ravishda yangilang;

6. xavfsiz tizimlar va ilovalarni ishlab chiqish va saqlash;

7. karta egasi ma’lumotlariga kirishni bilish kerak bo‘lganda cheklash;

8. kompyuterga kirish huquqiga ega bo‘lgan har bir shaxsga noyob identifikatorni tayinlang;

9. karta egasi ma’lumotlariga jismoniy kirishni cheklash;

10. tarmoq resurslari va karta egasi ma’lumotlariga kirishni kuzatib boring;

11. Xavfsizlik tizimlari va jarayonlarini muntazam ravishda sinab ko‘ring;

12. Xavfsizlik siyosatini saqlang va uni yangilab turing va chop eting.

PCI DSS muvofiqligi odatda o‘z-o‘zini baholash so‘rovnomalari va tashqi malakali xavfsizlik baholovchilari (QSA) tomonidan tekshirishlar kombinatsiyasi orqali baholanadi. Talab qilinadigan baholash darajasi tashkilotning tranzaksiya hajmiga va boshqa omillarga bog‘liq. PCI DSS muvofiqligi bir martalik harakat emas, balki doimiy jarayondir, chunki standart yangi tahdidlar va zaifliklarga moslashish uchun rivojlanadi. Tashkilotlar muvofiqlikni ta’minlash uchun xavfsizlik nazoratini muntazam ravishda ko‘rib chiqishlari va yangilashlari kerak.

PCI DSS standarti karta egasi ma’lumotlarining maxfiyligi, yaxlitligi va mavjudligini himoya qilish uchun mo‘ljallangan. Standartga rioya qilish orqali tashkilotlar ma’lumotlarning buzilishi, firibgarlik va boshqa xavfsizlik hodisalari xavfini kamaytirishi mumkin. PCI DSS ga muvofiqlik nafaqat kredit karta kompaniyalari tomonidan talab qilinadi, balki tashkilotlarga ham foyda keltiradi. Bu

mijozlar ishonchini oshirishi, obro‘siga putur yetkazish xavfini kamaytirishi va rioya qilmaslik natijasida moliyaviy jazolardan qochishi mumkin.

Shuni ta’kidlash kerakki, PCI DSS bilan muvofiqlik umumiy xavfsizlik strategiyasining faqat bir jihati hisoblanadi. Tashkilotlar xavflarni baholash, xodimlarni o‘qitish, hodisalarga javob berishni rejalashtirish va xavfsizlik nazoratini muntazam ravishda sinovdan o‘tkazish va monitoringini o‘z ichiga olgan xavfsizlikka kompleks yondashuvni qo‘llashlari kerak.

Bundan tashqari, xosting provayderlari va kontentni yetkazib berish tarmoqlari (CDN) kabi uchinchi tomon xizmatlari ham kiberxavfsizlikka xavf tug‘dirishi mumkin. Uchinchi tomon provayderlarini tekshirish va ularning tegishli xavfsizlik choralari mavjudligini ta’minlash muhimdir.

Tegishli xavfsizlik choralarini qo‘llash, so‘nggi tendentsiyalar va tahdidlardan xabardor bo‘lish, qoidalar va standartlarga rioya qilish va uchinchi tomon provayderlari, veb-sayt egalari va ishlab chiquvchilarini tekshirish orqali kiberxavfsizlik xatarlarini yumshatish va ularning xavfsizligi va maxfiyligini ta’minlashga yordam beradi. foydalanuvchilar ma’lumotlari. Kiberxavfsizlikka proaktiv yondashish orqali biz raqamli infratuzilmamiz va unga tayanadigan foydalanuvchilarni himoya qilishga yordam bera olamiz.

1.3. Internet makonidagi Veb-saytlarning zaifliklarini aniqlash usullari tahlili

Internetda taxminan 1,8 milliard veb-sayt mavjud. Ularning ko‘pchiligi kiber jinoyatchilar uchun oson o‘ljaga aylantiradigan zaifliklarga ega . Tadqiqotchilarning so‘nggi izlanishlariga ko‘ra, kontentni boshqarish tizimining (Content Management System) (*CMS*) o‘rnatilishining 56% dan ortig‘i eskirgan shuning uchun hujumga moyil. Boshqa bir tadqiqot shuni ko‘rsatadiki, veb-saytlarda ishlaydigan veb-ilovalarning 19 foizi zaifdir. Global kontekstda bu statistika ulkan hujum yuzasiga aylanadi . Veb-saytlarni zaifliklarini aniqlashni bir necha xil usullari

mavjud misol uchun veb-sayt skanerlari veb-ilovalarga kiberxavfsizlik hujumlarining oldini olishda muhim texnologiyadir . Va bu turdagi hujumlar katta muammo hisoblanadi. Forrester Research ma’lumotlariga ko‘ra, veb-saytlar eng ko‘p hujum uyushtiriladigan obyektlardir.

Buzg‘unchilar odatda veb-ilovalarning umumiy zaifliklarini nishonga olishadi, bu esa dastur konfiguratsiyasining xavfsizlik zaif tomonlaridan foydalanish va axborot tizimlariga chuqurroq kirib borishga urinishadi. Natijada, umumiy zaifliklarni aniqlash uchun xavfsizlik skaneridan foydalangan holda dastur zaifliklarini aniqlash mexanizmlarini sozlash muhimdir. Bundan tashqari, veb-ilovalar xavfsizligi siyosatini bilish har doim muammolarni oldini olishning yaxshi usuli hisoblanadi.

Tashkilotlar odatda *ethical* xakerlar , ixtisoslashtirilgan vositalar va xavfsizlik auditidan foydalanib, ilovalar xavfsizligi zaifliklarini protativ tarzda aniqlashadi. Buning asosiy maqsadi dastur kodini boyitish bo‘lsa-da, xavfsizlik muammolari yo‘q, bir qator boshqa tashkiliy darajadagi afzalliklar mavjud, masalan:

Zaiflikni uzluksiz skanerlash va sinovdan o‘tkazish pentesterchilar va ishlab chiquvchilarga bosimni engillashtirib, xavfsizlik operatsiyalarini soddalashtiradi. Uzluksiz dalillarga asoslangan skanerlash yordamida xavfsizlik mutaxassislari zaifliklarni birma-bir topib, hal qilishlari mumkin. To‘plamlarda zaifliklarni topish muntazam ravishda ish yukini samarali boshqarishni taklif qiladi va tegishli bo‘limlar o‘rtasidagi hamkorlikni rivojlantiradi.

Penetratsion test - bu xavfsizlik bo‘yicha mutaxassislar turli xil SQL in’ektsiyalari, saytlararo skriptlar, saytlararo so‘rovlarni soxtalashtirish va saytlararo so‘rovlar kabi zaifliklardan xavfsiz foydalanishga harakat qiladigan proaktiv xavfsizlik yondashuvidir . Zaifliklar aniqlangandan so‘ng, tashkilotlar tajovuzkorning harakatlarini taqlid qiladi va tushunadi. Xavfsizlik guruhlari xavfsizlik mexanizmlarining samaradorligi va xavfsizlik siyosatiga muvofiqligini

baholash uchun penetratsion testlarni o‘tkazadi. Bu soha vakillari Pentesterlar deb nomlanadi.

Eng yomoni, so‘nggi bir necha yil ichida bunday hujumlar doimiy ravishda o‘sib bormoqda. Va hatto katta hujum vektorini taklif qiluvchi dasturiy ta’minotning zaifliklaridan ham ko‘proq - bu tashqi kirishning odatiy yo‘li bo‘lgan veb-saytlardir.

Ushbu hujumlardan himoyalanish uchun keling, veb-sayt skanerlarini ko‘rib chiqamiz, so‘ngra yetakchi veb-sayt skaneri dasturiy ta’minotiga chuqur kirib boramiz. AT xavfsizligida xavfsizlik skanerlari haqida chalkashliklar mavjud. Masalan veb-sayt skaneri, zaifliklarni skanerlash vositasi, veb-sayt zaifliklari skaneri va veb-ilovalar skaneri kabi atamalar bir-birining o‘rnida ishlatiladi. Lekin bu xato.

Zaiflik skanerlari va veb-sayt zaiflik skanerlari bir-birdan farq qiladi. Veb-sayt skaneri veb-saytni masofadan skanerlaydi va ko‘pincha sayt skanerlanganligini ko‘rsatish uchun qo‘shilishi mumkin bo‘lgan grafikni taqdim etadi. Zaiflik skanerlari esa zaifliklarni qidirib, AT tarmog‘ini, so‘nggi nuqtalarni va infratuzilmani skanerlaydi.

Zaiflik skanerlari xavfsizlik zaifliklarini aniqlash uchun ilovalar va tarmoqlarni doimiy ravishda kuzatib boradi. Ular turli yo‘llar bilan ishlaydi. Ularning ko‘pchiligi ma’lum zaifliklarning so‘nggi ma’lumotlar bazasini saqlaydi va mumkin bo‘lgan xavf va ekspluatatsiyalarni aniqlash uchun skanerdan o‘tkazadi. Ular odatda AT tomonidan ilovalar va tarmoqlarni ma’lum muammolarga qarshi sinab ko‘rish, shuningdek, yangi zaifliklarni aniqlashga yordam berish uchun ishlatiladi. Shuningdek, ular ma’lum zaifliklar va potentsial yangi ekspluatatsiyalar tahlili asosida hisobotlarni taqdim etadilar.

Zaiflikni skanerlash, keyin xavfsizlik teshiklarini aniqlash uchun potentsial eksplutatsiya nuqtalarini tekshirish bilan shug‘ullanadi. Muntazam tekshiruvlar tizimning zaif tomonlarini aniqlaydi va tasniflaydi. Ba’zi hollarda, dastur qarshi

choralarning samaradorligi haqida bashoratlarni taklif qiladi. Skanerlash IT bo‘limi tomonidan yoki boshqariladigan xizmat orqali amalga oshirilishi mumkin.

Odatda, skanerlash xizmatlar va portlardagi ma’lum bo‘lgan xavfsizlik teshiklari, shuningdek, paket qurilishidagi anomaliyalar, yetishmayotgan qism (patch) va foydalaniladigan dasturlar yoki skriptlar uchun mavjud bo‘lishi mumkin bo‘lgan yo‘llar haqidagi ma’lumotlarni aniqlaydi.

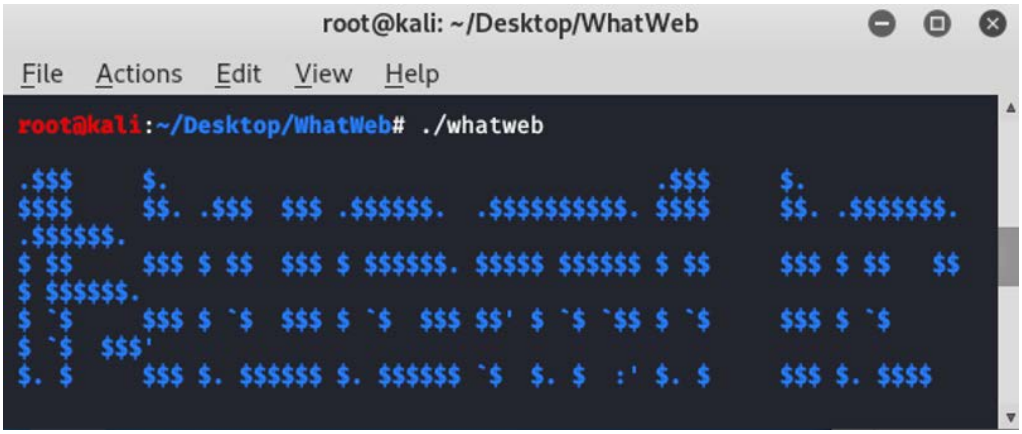
Ba’zi zaiflik skanerlari zaifliklarni aniqlaydi va mumkin bo‘lgan choralarni taklif qiladi. Boshqalar esa atrof-muhit bo‘ylab tuzatish va yumshatishga harakat qilishadi. Ba’zilari hisobot berish orqali audit va muvofiqlikni kuchli qo‘llab-quvvatlaydi yoki PCI DSS, Sarbanes-Oxley yoki HIPAA kabi xavfsizlik standartlariga qaratilgan. Boshqalar esa veb-ga asoslangan kamchiliklarni yoki autentifikatsiya hisob ma’lumotlari, kalitga asoslangan autentifikatsiya va hisob ma’lumotlari omborlari bilan bog‘liq muammolarni topishga ixtisoslashgan.

Veb-skanerlar tomonidan qo‘llaniladigan usullarga ilovalarni skanerlash, standart tarkibni tekshirish, shuningdek, umumiy tarkibni topish va umumiy zaifliklar uchun veb-ilovalarni tekshirish kiradi. Skanerlash faol yoki passiv amalga oshirilishi mumkin.

– passiv yondashuv foydali bo‘lgan, lekin ko‘pincha yetarlicha chuqur bo‘lmagan intruziv tekshiruvlarni amalga oshiradi;

– faol skanerlar veb-saytlar va veb-ilovalarga hujumlarni simulyatsiya qiladi. Ba’zi vositalar, shuningdek, boshqa zaifliklarni aniqlash mumkinmi yoki yo‘qligini ko‘rish uchun kirish ruxsatnomalaridan foydalanadi.

Saytni buzishga urinishdan oldin pentester (penetratsiya testeri) u haqida ma’lumot to‘plashi kerak. WhatWeb deb nomlangan vosita bu ishni qila oladi. U CMS va qo‘llaniladigan qo‘shimcha komponentlar haqidagi ma’lumotlarni oladi.



1.3.1-rasm. Kali linuxdagi whatweb interfeysi.

WhatWeb veb-sayt tomonidan qo‘llaniladigan texnologiyalarni aniqlash uchun ishlatilishi mumkin bo‘lgan veb-ilovalar skaneridir. Bu bepul va ochiq manbali vosita bo‘lib, u veb-saytning asosiy texnologiya steklari haqida batafsil ma’lumot berish uchun mo‘ljallangan.

Ushbu ko‘rib chiqishning maqsadi eng ko‘p ma’lumot beradigan eng funktsional va aniq skanerlarni aniqlash emas, balki xavfsizlik bo‘yicha mutaxassislar tomonidan tezkor xavfsizlikni baholash vazifalari uchun optimal bo‘lgan skanerlarni aniqlashdir. Quyida saytlarni xavfsizligini tekshiradigan bir nechta platformalar (baholash usullari) keltirilgan.

Web sayt xavfsizligini tekshirishning tasdiqlangan usullari:

1. Acunetix;
2. Detectix;
3. H-X Scanner;
4. Intruder;
5. ImmuniWeb;
6. Norton Safe Web;
7. Netsparker Cloud.

Bu platformalarni qulayligi quyidagicha:

– bepul foydalanish imkoniyati;

– foydalanuvchini ro‘yxatdan o‘tkazish zarurati mavjud emasligi;
– muammolarni aniqlash tezligi va bepul funksionallik miqdori.

I bob bo‘yicha xulosalar

Magistrlik dissertatsiyasi ishining birinchi bobining 1-bo‘limida Xavfsiz veb-saytlarning tuzilishi arxitekturalari, xavfsizlikni ta’minlashdagi muammolar va ularni hal qilish usullari tahlil qilindi. Tahlil natijasida Malumotlarni shifrlash (Encrypt data) - ma’lumotlar xavfsizligi veb-sayt xavfsizligining eng muhim jihati hisoblanishi aniqlandi. HTTPS (Hypertext Transfer Protocol Secure) yordamida veb-saytlarni shifrlash onlayn aloqani ta’minlash va maxfiy ma’lumotlarni himoya qilishda muhim qadamdir. HTTPS internet orqali xavfsiz aloqa uchun protokol bo‘lib, u foydalanuvchi brauzeri va veb-sayt o‘rtasida uzatiladigan ma’lumotlarni himoya qilish uchun shifrlashdan foydalanadi.

2-bo‘limda Internet makoni Veb-saytlarida kiberxavfsizlikni ta’minlash muammolari ham tahlil qilindi. Tahlil natijasida saytlararo so‘rovlarni qalbakilashtirish, Zararli dastur veb-saytga zarar etkazishi va tashrif buyuruvchilar qurilmalariga tarqalishi, Veb-saytlarda kiberxavfsizlikni ta’minlashning bir usuli bu veb-ilovalar xavfsizlik devorini (WAF) amalga oshirish ekanligi aniqlandi. WAF ni yaratish va undan foydalanish usullari ko‘rib chiqildi. Bundan tashqari, xosting provayderlari va kontentni yetkazib berish tarmoqlari (CDN) kabi uchinchi tomon xizmatlari ham kiberxavfsizlikka xavf tug‘dirishi mumkin. Uchinchi tomon provayderlarini tekshirish va ularning tegishli xavfsizlik choralari mavjudligini ta’minlash muhimligi ta’kidlandi.

3-bo‘limda esa Internet makonidagi Veb-saytlarning zaifliklarini aniqlash usullari tahlili amalga oshirildi. Zaiflikni uzluksiz skanerlash va sinovdan o‘tkazish pentesterchilar va ishlab chiquvchilarga bosimni engillashtirib, xavfsizlik operatsiyalarini soddalashtirishi aniqlandi. Veb-skanerlar tomonidan qo‘llaniladigan usullarga ilovalarni skanerlash, standart tarkibni tekshirish, shuningdek, umumiy

tarkibni topish va umumiy zaifliklar uchun veb-ilovalarni tekshirish kiradi. Web sayt xavfsizligini tekshirishning tasdiqlangan usullari ko‘rib chiqildi.

II BOB. INTERNET TARMOG‘I VEB-SAYTLARI UCHUN XAVFSIZLIK CHORALARINI ISHLAB CHIQISH

2.1. Internet makonida Veb saytlarga uyushtiriladigan hujum turlari tahlili

Kiberhujumlar ortib bormoqda. Zamonaviy texnologiyalar ko‘plab qulayliklar va imtiyozlarni taqdim etsa-da, undan noto‘gri foydalanadigan odamlar bor, bu esa butun dunyo bo‘ylab biznes va ma’lumotlar maxfiyligiga tahdid soladi. Web sayt xavfsizligi haqida gap ketganda foydalanuvchilarning 85% hech qachon o‘z ma’lumotlarini xavfsiz kanallardan ketayotganini bilishmaydi. Bundan ham yomonrog‘i 82% hech qachon xavfsiz bo‘lmagan saytlarga kirishayotganini sezishmaydilar ham.

Ma’lumotlar buzilishi sodir bo‘lganda, bu juda katta ta’sir ko‘rsatishi mumkin. Bu maqsadli kompaniya doirasidan tashqariga chiqadi, mijozlar, etkazib beruvchilar va boshqalarga ta’sir qiladi. Ajablanarlisi shundaki, ekspertlar 2025 yilga borib kiberjinoyatlarning narxi 10,5 trillion dollarga yetishini kutishmoqda. Tarixda ko‘plab davlat idoralari, korparatsiyalar va jismoniy shaxslar bundan ancha qiynalib kelishgan. Keling shunday hujumlar tarixini ko‘rib chiqamiz.

– Melissa virusi- eng birinchi va eng katta kiber tahdidlardan biri 1999 yilda dasturchi Devid Li Smit tomonidan Melissa virusidan kelib chiqqan. U foydalanuvchilarga virusni saqlaydigan Microsoft Word orqali ochish uchun fayl yubordi. Virus ochilgandan so‘ng faollashdi va yuzlab kompaniyalarga, shu jumladan Microsoftga jiddiy zarar etkazdi. Ta’kidlanishicha, zararlangan tizimlarni ta’mirlash uchun 80 million dollar kerak bo‘lgan;

– NASA kiber hujumi - 1999 yilda 15 yoshli Jeyms Jonatan NASA kompyuterlarini buzib, 21 kunga o‘chirib qo‘ydi! Hujum paytida taxminan 1,7 million dasturiy ta’minot yuklab olingan bo‘lib, bu kosmik gigantni ta’mirlash uchun taxminan 41 000 dollarga tushdi;

– Estoniya kiberhujumi - 2007 yil aprel oyida Estoniya butun bir okrugga birinchi kiber hujumga guvoh bo‘ldi. Asosan, 58 ga yaqin Estoniya veb-saytlari, jumladan hukumat, bank va ommaviy axborot vositalari veb-saytlari oflayn rejimga o‘tganini ko‘rdi;

– Adobe Cyber Attack - Adobe kiberhujum birinchi marta 2,9 million foydalanuvchining ma’lumotlarini buzgan deb taxmin qilingan. Bundan tashqari, u 38 million foydalanuvchining shaxsiy ma’lumotlarini buzdi! Adobe kompaniyasining ta’kidlashicha, faqat dastlabki 2,9 million foydalanuvchining parollari va kredit karta ma’lumotlari buzilgan, ammo qolgan 35,1 million foydalanuvchi parollari va foydalanuvchi identifikatorlaridan mahrum bo‘lgan;

– Yahoo ga kiribhujum - 2014-yilda Yahoo yilning eng yirik kiberhujumlaridan biriga duchor bo‘ldi, o‘shanda 500 million akkaunt buzilgan. Hujum paytida asosiy ma’lumotlar va parollar o‘girlangan, bank ma’lumotlari esa o‘girlanmagan;

– Marriott mehmonxonalariga kiberhujum - Marriott mehmonxonalari va Starwood mehmonxonalari guruhiga kiberhujum yillar davomida aniqlanmay qoldi, bu faqat 2018 yilda ma’lum bo‘ldi. Shunday qilib, ular hujumdan xabardor bo‘lgan vaqtga kelib, taxminan 339 million mehmonning ma’lumotlari buzilgan. Natijada, Buyuk Britaniyaning ma’lumotlar maxfiyligini nazorat qiluvchi tashkilot Marriott mehmonxonalarini 18,4 million funt sterling miqdorida jarimaga tortdi.

Bunday hujumlar yildan-yilga ortib bormoqda buning oldini olish uchun mutaxasislar uzluksiz ish yuritishmoqda. Hozirgi kunda keng tarqarqal hujum turlari haqida to‘xtalib o‘tsak. Bular:

1. Saytlararo skript yaratish (XSS): XSS hujumlari boshqa foydalanuvchilar tomonidan ko‘riladigan veb-sahifalarga zararli skriptlarni kiritishni o‘z ichiga oladi. Bu tajovuzkorga jabrlanuvchining brauzerida

o‘zboshimchalik bilan kodni bajarish, maxfiy ma’lumotlarni o‘girlash yoki veb-sayt tarkibini manipulyatsiya qilish imkonini beradi;

2. SQL Injection (SQLi): SQL in’ektsion hujumlari tajovuzkor veb-ilovaning ma’lumotlar bazasi so‘roviga zararli SQL bayonotlarini kiritganda sodir bo‘ladi. Bu ma’lumotlar bazasiga ruxsatsiz kirish, ma’lumotlarni manipulyatsiya qilish yoki maxfiy ma’lumotlarning oshkor etilishiga olib kelishi mumkin;

3. Saytlararo so‘rovlarni qalbakilashtirish (CSRF): CSRF hujumlari autentifikatsiya qilingan foydalanuvchilarni ularning bilimi yoki roziligisiz veb-saytda ko‘zda tutilmagan harakatlar qilish uchun aldaydi. Veb-sayt va foydalanuvchi o‘rtasidagi ishonchdan foydalanib, tajovuzkorlar parollarni o‘zgartirish, kiruvchi tranzaktsiyalarni amalga oshirish yoki foydalanuvchi sozlamalarini o‘zgartirish kabi amallarni bajarishi mumkin;

4. Taqsimlangan xizmat ko‘rsatishni rad etish (DDoS): DDoS hujumlari veb-sayt serverlari yoki tarmoq infratuzilmasini haddan tashqari ko‘p trafik bilan to‘ldirishga qaratilgan bo‘lib, saytni qonuniy foydalanuvchilarga kira olmaydi. Buzgunchilar odatda veb-sayt resurslarini to‘ldirish uchun botnetlardan yoki boshqa vositalardan foydalanadilar, bu esa xizmatda uzilishlarga olib keladi;

5. Shafqatsiz kuch hujumlari: Shafqatsiz kuch hujumida tajovuzkor veb-saytga yoki uning backend tizimlariga ruxsatsiz kirish uchun foydalanuvchi nomlari va parollarning turli kombinatsiyalarini muntazam ravishda sinab ko‘radi. Ushbu hujum zaif yoki osongina taxmin qilinadigan hisob ma’lumotlarining zaifligiga tayanadi;

6. Fayllarni qo‘shish ekspluatatsiyasi: Fayllarni qo‘shish hujumlari tajovuzkorga veb-serverga zararli fayllarni qo‘shish va amalga oshirish imkonini beradigan zaifliklardan foydalanadi. Bu masofaviy kod bajarilishiga, ruxsatsiz kirishga yoki serverning buzilishiga olib kelishi mumkin;

7. Server tomonidagi so‘rovlarni soxtalashtirish (SSRF): SSRF hujumlari veb-serverni tajovuzkor nomidan zararli so‘rovlar qilish uchun aldashni o‘z ichiga oladi. Bu xavfsizlik devori cheklovlarini chetlab o‘tish, ichki resurslarga kirish yoki boshqa tizimlarga qaratilgan hujumlarni amalga oshirish uchun ishlatilishi mumkin;

8. Clickjacking: Clickjacking hujumlari qonuniy veb-sayt tarkibiga zararli elementlarni joylashtirish orqali foydalanuvchilarni aldaydi, ularni yashirin tugmalar yoki havolalarni bosishga undaydi. Bu ko‘zda tutilmagan harakatlarga yoki maxfiy ma’lumotlarning oshkor etilishiga olib kelishi mumkin;

9. Remote Code Execution (RCE): RCE hujumlari tajovuzkorga maqsadli server yoki dasturda o‘zboshimchalik bilan kodni bajarishga imkon beradi. Zaifliklardan foydalangan holda, tajovuzkorlar tizim ustidan to‘liq nazoratni qo‘lga kiritishlari mumkin, bu esa ma’lumotlarning buzilishi yoki tizimning buzilishiga olib kelishi mumkin;

10. Fishing hujumlari: veb-saytlar bilan cheklanmagan bo‘lsa-da, fishing hujumlari foydalanuvchilarni kirish ma’lumotlari, kredit karta ma’lumotlari yoki shaxsiy ma’lumotlar kabi nozik ma’lumotlarni oshkor qilish uchun aldash uchun qonuniy saytlarni taqlid qiluvchi firibgar veb-saytlarni yaratishni o‘z ichiga oladi.

Endi shu hujum turlarini oldini olish bo‘yisha to‘xtalib otamiz. Misol uchun CSRF hujumlar

Saytlararo so‘rovlarni qalbakilashtirish (CSRF) hujumlari foydalanuvchilar va veb-saytlar o‘rtasida o‘rnatilgan ishonchdan foydalanib, veb-ilovalarga jiddiy tahdid soladi. Ushbu tahlil CSRF hujumlarini, ularning mexanizmlarini, mumkin bo‘lgan oqibatlarini va samarali yumshatish strategiyalarini chuqur o‘rganishni ta’minlaydi. Asosiy tamoyillarni tushunish va tegishli qarshi choralarni ko‘rish orqali ishlab chiquvchilar va xavfsizlik amaliyotchilari veb-ilovalarni CSRF zaifliklaridan himoya qilishlari mumkin. CSRF ruxsatsiz harakatlarni amalga oshirish uchun foydalanuvchilar va veb-saytlar o‘rtasidagi ishonchdan

foydalanadigan hujum turi sifatida aniqlanadi. Ushbu bo‘limda CSRF ning veb-ilovalar xavfsizligidagi ahamiyati ta’kidlangan va CSRF hujumlarining tarixiy misollari va ularning tashkilotlarga ta’siri keltirilgan.

Ushbu bo‘limda CSRF hujumlari qanday ishlashi haqida bosqichma-bosqich tushuntirish berilgan. U foydalanuvchilarning ishonchidan foydalanishda foydalaniladigan usullarni, jumladan, foydalanuvchi sessiyalari, cookie-fayllar va autentifikatsiya tokenlarining rolini o‘rganadi.

1. CSRF hujumidagi birinchi qadam jabrlanuvchi foydalanuvchining maqsadli veb-saytda o‘zini autentifikatsiya qilishidir. Muvaffaqiyatli autentifikatsiyadan so‘ng veb-sayt foydalanuvchi uchun seans yaratadi, odatda seans cookie-fayllaridan foydalanadi. Ushbu seans kukilari foydalanuvchi seansini o‘rnatuvchi va keyingi so‘rovlarni foydalanuvchi sessiyasi bilan boGlash imkonini beruvchi noyob identifikatorni o‘z ichiga oladi;

2. Jabrlanuvchi foydalanuvchi autentifikatsiya qilingandan so‘ng, ular maqsadli veb-sayt bilan o‘zaro aloqani davom ettiradilar va turli harakatlarni bajaradilar. Shu bilan birga, tajovuzkor foydalanuvchini o‘z nazorati ostidagi zararli veb-saytga kirish uchun aldaydi. Bunga aldamchi ijtimoiy muhandislik texnikasi, elektron pochta orqali fishing yoki boshqa vositalar orqali erishish mumkin;

3. Foydalanuvchi zararli veb-saytda bo‘lganda, tajovuzkorning maqsadi maqsadli veb-saytda ruxsatsiz harakatlarni amalga oshirishdir. Bunga erishish uchun tajovuzkor foydalanuvchi hisob sozlamalarini o‘zgartirish, moliyaviy operatsiyalarni amalga oshirish yoki shakllarni yuborish kabi kerakli amalni bajaradigan so‘rovni ishlab chiqadi;

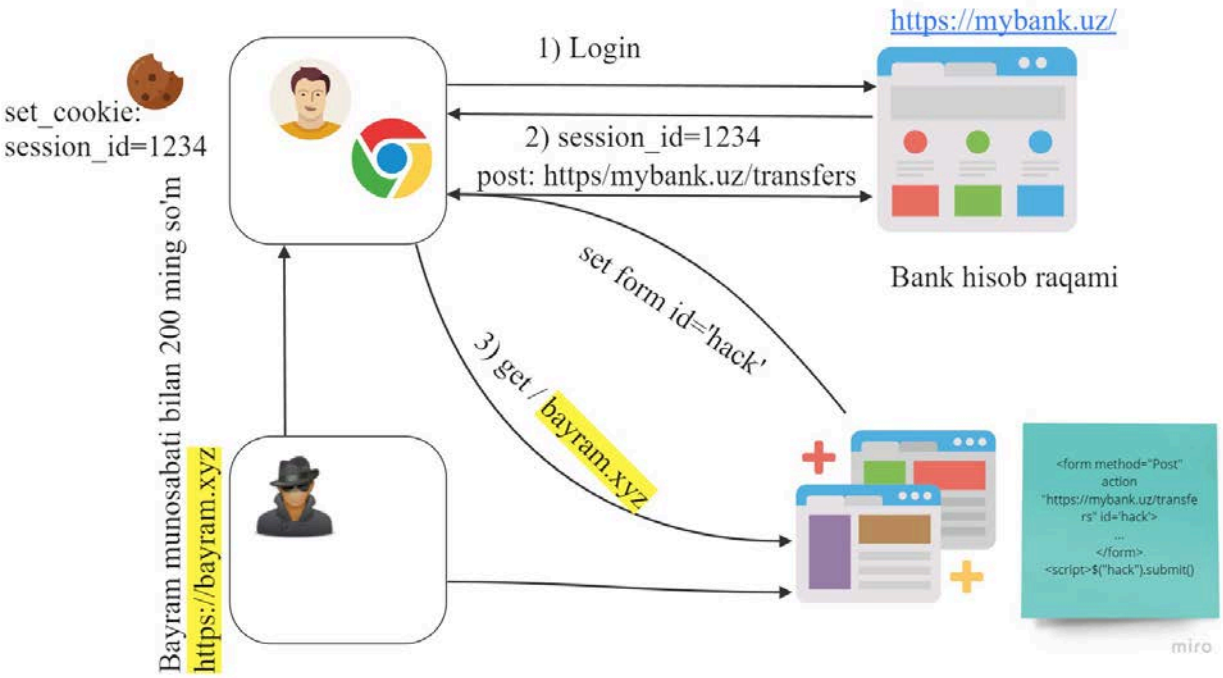
4. Keyingi qadam jabrlanuvchining brauzerini maqsadli veb-saytga tayyorlangan so‘rovni yuborishga undashdir. Bu foydalanuvchi brauzeri va maqsadli veb-sayt o‘rtasidagi ishonchdan foydalanish orqali amalga oshiriladi. Buzgunchi

odatda zararli veb-saytga HTML yoki JavaScript kodini o‘z ichiga oladi, bu esa foydalanuvchining xabarisiz avtomatik ravishda so‘rovni ishga tushiradi;

5. Maqsadli veb-saytda autentifikatsiya talablarini chetlab o‘tish uchun tajovuzkor zararli so‘rov seans kukilari yoki autentifikatsiya tokenlari kabi har qanday zarur hisob ma’lumotlarini o‘z ichiga olishini ta’minlaydi. Jabrlanuvchi foydalanuvchi allaqachon maqsadli veb-saytda autentifikatsiya qilinganligi sababli, ularning brauzeri avtomatik ravishda so‘rov bilan tegishli cookie yoki tokenlarni o‘z ichiga oladi;

6. Jabrlanuvchining brauzeri soxta so‘rovni maqsadli veb-saytga yuborganda, u jabrlanuvchining autentifikatsiya ma’lumotlarini o‘z ichiga oladi. Maqsadli veb-sayt, so‘rovning soxta ekanligini bilmagan holda, uni jabrlanuvchi foydalanuvchi tomonidan boshlangan qonuniy so‘rov kabi qayta ishlaydi. Natijada, potentsial zararli oqibatlarga olib keladigan ruxsatsiz harakat amalga oshiriladi;

7. CSRF hujumlarining muhim jihati shundaki, jabrlanuvchi foydalanuvchi o‘z brauzeri zararli so‘rov yuborayotganidan bexabar qoladi. So‘rov jabrlanuvchining shaxsiy brauzeridan kelib chiqqanligi va ularning hisob ma’lumotlarini o‘z ichiga olganligi sababli, maqsadli veb-sayt buni foydalanuvchi tomonidan amalga oshirilgan qonuniy harakat sifatida qabul qiladi.



2.1.1- rasm. CSRF hujumining ishlash prinspi.

Ho‘sh bunday hujumlardan qanday himoyalanish kerak degan savol tuGiladi. Endi bunday hujumlardan himoyalanish chora tadbirlarini ko‘rib chiqamiz.

– CSRF tokenlaridan foydalaning: har bir HTML formasi yoki AJAX so‘roviga noyob va oldindan aytib bo‘lmaydigan tokenni qo‘shing. Ushbu token foydalanuvchi seansiga bog‘langan bo‘lishi va so‘rovni qayta ishlashdan oldin server tomonida tekshirilishi kerak. Token so‘rov tajovuzkordan emas, baki veb-saytingizdan kelib chiqqanligini ta’minlashga yordam beradi;

– SameSite cookie-fayllarini qo‘llang: Cookie-fayllar doirasini cheklash uchun SameSite atributini o‘rnating. SameSite atributini "Strict" yoki "Lax" ga o‘rnatish orqali siz ko‘p hollarda CSRF hujumlarini samarali tarzda yumshatib, saytlararo so‘rovlarda cookie-fayllarni yuborilishining oldini olishingiz mumkin;

– CAPTCHA yoki reCAPTCHA-ni qo‘llang: CAPTCHA yoki reCAPTCHA sinovlaridan muhim harakatlar yoki shakllarda so‘rov

avtomatlashtirilgan skript emas, balki inson tomonidan amalga oshirilayotganiga ishonch hosil qilish uchun foydalaning. Bu CSRF hujumlari xavfini kamaytirishga yordam beradi, bu tajovuzkorlarga zararli so‘rovlarni avtomatlashtirishni qiyinlashtiradi;

– ilovangizni muntazam yangilang va tuzating: Veb-ilova va server dasturiy ta’minotini so‘nggi xavfsizlik yamoqlari bilan yangilab turing. Dasturiy ta’minotdagi zaifliklardan buzgunchilar CSRF himoyasini chetlab o‘tish uchun foydalanishlari mumkin;

– xavfsizlik sarlavhalaridan foydalaning: Veb-sahifalaringizdagi ruxsatsiz skriptlarning bajarilishini cheklash orqali CSRF hujumlarini yumshatishga yordam beradigan Content Security Policy (CSP) kabi xavfsizlik sarlavhalarini joriy qiling;

– xavfsizlik testini o‘tkazing: Potentsial zaifliklarni, shu jumladan CSRF zaifliklarini aniqlash va bartaraf etish uchun muntazam ravishda xavfsizlikni baholashni, shu jumladan zaifliklarni skanerlash va kirish testini o‘tkazing.

Web-saytlar bunday hujumlardan himoyalash uchun doimiy web-saytlarni himoya tizimini tekshirib boorish kerak. Buning uchun bir nechta xizmat turlari mavjud misol uchun H-X Scanneri. Bular orasidan eng avzali, H-X Scanner hisoblanadi. Sababi, qolgan platformalatdan o‘zining qulay interfeysi bilan ajralib turadi.

H-X Scanner - bepul onlayn zaiflik skaneri. U 2016 yildan beri ishlaydi va ikkita rejimga ega: tez va normal. Xavfsizlik hisobotiniolish uchun web saytning URL manzili va shaxsiy elektron pochta manzilini kiritishingiz kerak. Xizmatdan foydalanish juda oson va ro‘yxatdan o‘tishnitalabqilmaydi. Tez rejimda birinchi natijalarni real vaqtda ishga tushirilgandan keyin bir necha soniya ichida olish mumkin. Barcha tezkor skanerlash jarayoni 5 minut davom etadi. Oddiy rejimda skanerni ochiq saqlash shart emas. Skanerlash hisobotlari elektron pochta orqali

jo‘natiladi. Oddiy rejimda hisobot juda qulay. U jadval ko‘rinishidagi xulosa va tafsilotlarni o‘z ichiga qamrab oladi. Zaifliklarni qo‘lda tekshirish funksionalligi ta’minlangan.

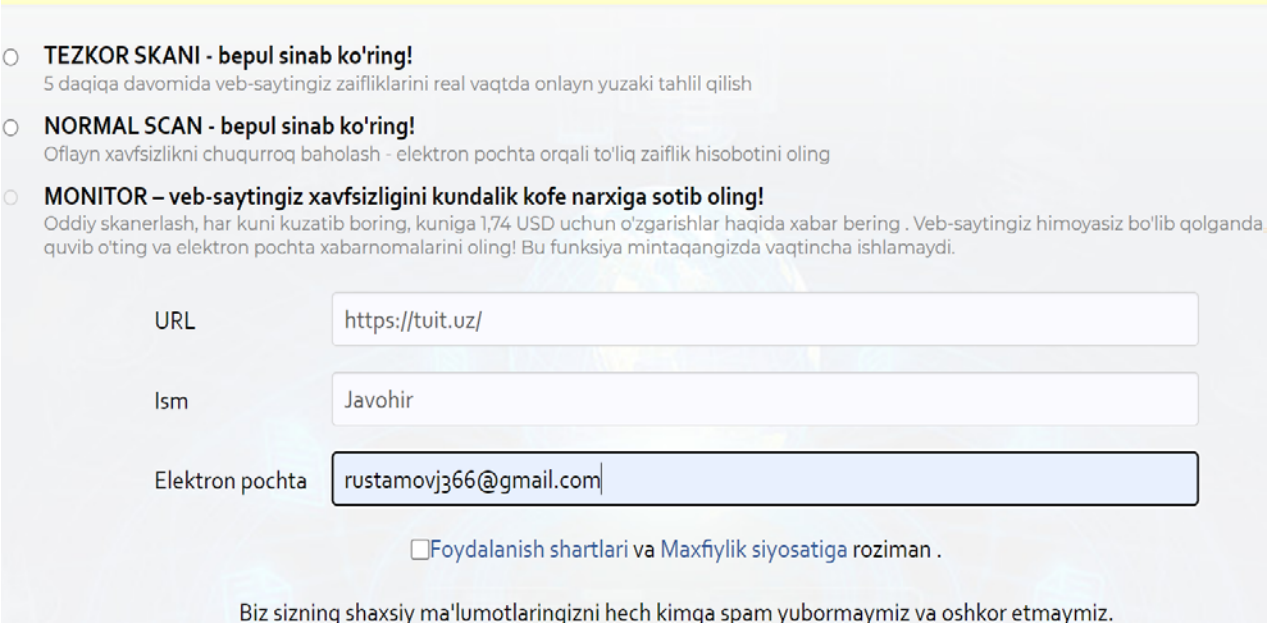
H-X Scanner yordamida *tuit.uz* va *daryo.uz* saytlarni xavfsizligini baholash quyidagicha amalga oshiriladi.

Tuit.uz misolida ko‘rib chiqamiz:

1 – qadam: https://service.h-x.technology/scan link orqali H-X Scanner platformasiga kiriladi. Natijada quyidagicha oyna hosil bo‘ladi.

2.1.2-rasm. H-X Scanner platformasi interfeysi.

2-qadam Sayt url manzilini kiritilgandan kerakli rejim tanlanadi



2.1.3-rasm. Veb-sayt manzili kiritish joylari.

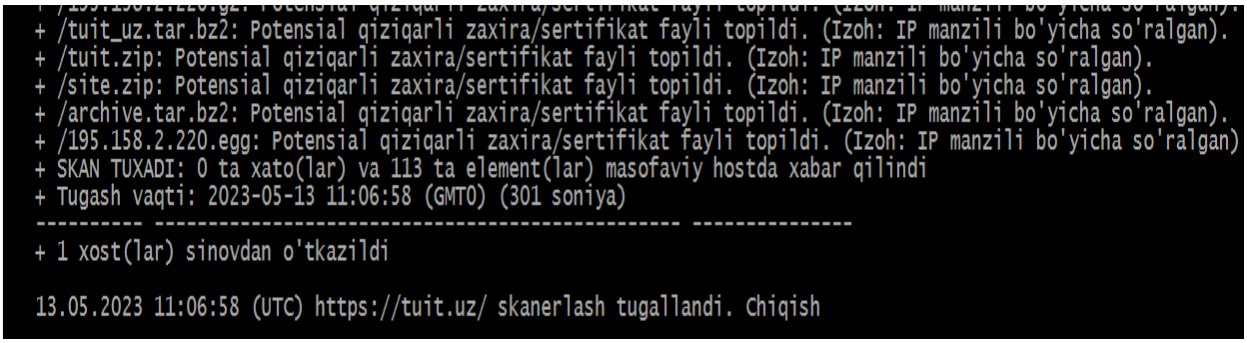
3-qadam Natijada, tuit.uz sayti skanerlash boshlanadi. Bu biroz vaqt oladi (5min). Sababi, tezkor skanerlash rejimi avval tanlangan. Agar Normal skanerlash rejimi tanlanganda, bu bir necha soat kutishni talab qilar edi. Lekin normal rejimda xavfsizlik chuqurroq va sifatliroq tekshirilishini aytib o‘tish joiz. Bu skanerlash usuli sayt haqida barcha ma’lumotlarni tekshirib ekranga chiqaradi.

```
QUICK SCAN ishga tushirilmoqda...

06.06.2023 08:48:37 (UTC) https://tuit.uz/ uchun tezkor skanerlash boshlandi

------------------ ------------------------------------------------ -----
+ Nishon IP: 195.158.2.220
+ Maqsadli xost nomi: tuit.uz
+ Maqsadli port: 443
------------------------ ------------------------------------------------
+ SSL ma'lumotlari: Mavzu: /CN=static.tuit.uz
Altnomlar: static.tuit.uz, tuit.uz, www.tuit.uz
Shifrlar: TLS_AES_256_GCM_SHA384
Emitent: /C=US/O=Let's Encrypt/CN=R3
+ Start Vaqt: 2023-06-06 08:48:40 (GMT0)
--------------------------------- ---------------------------------------
+ Server: nginx/1.24.0
+ Qabul qilingan kirish-nazorat-allow-origin sarlavhasi: *
+ Anti-clickjacking X-Frame-Options sarlavhasi mavjud emas.
+ Sayt SSL-dan foydalanadi va Strict-Transport-Security HTTP sarlavhasi aniqlanmagan.
+ X-Content-Type-Options sarlavhasi o'rnatilmagan. Bu foydalanuvchi agentiga sayt mazmunini MI
boshqacha tarzda ko'rsatishga imkon berishi mumkin.
+ PHPSESSID cookie xavfsiz bayroqsiz yaratilgan
+ Cookie _csrf xavfsiz bayroqsiz yaratilgan
```

2.1.4-rasm. 1-saytni skaner qilish jarayoni.

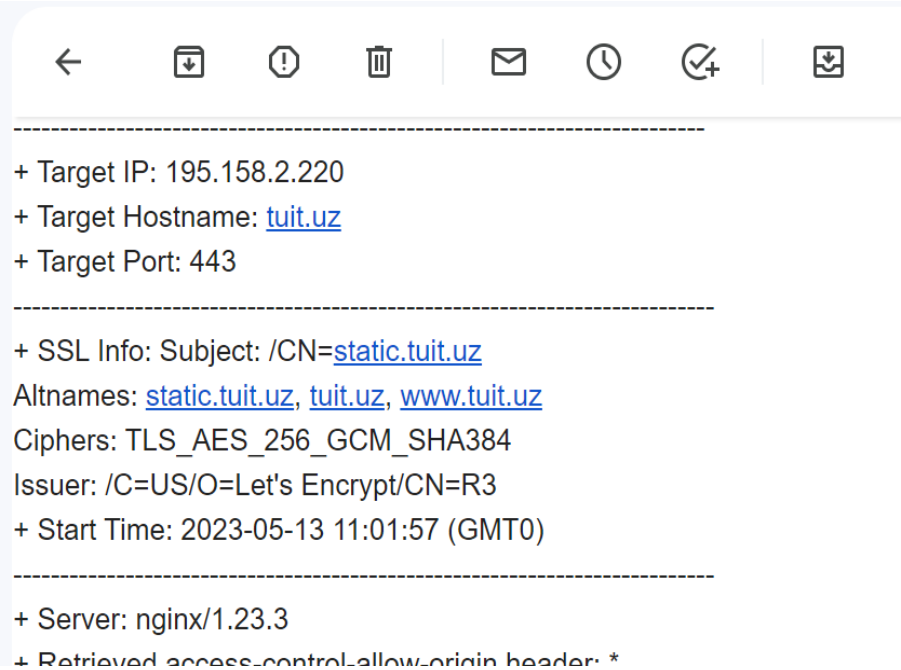
Tuit.uz sayti skanerlanmoqda

4-qadam: Skanerlash amalga oshirilib bo‘lingach, quyidagicha axborot taqdim etiladi



2.1.5-rasm. 1-saytni skaner natijasi.

5-qadam Skanerlash natijasi gmail ga keladi.



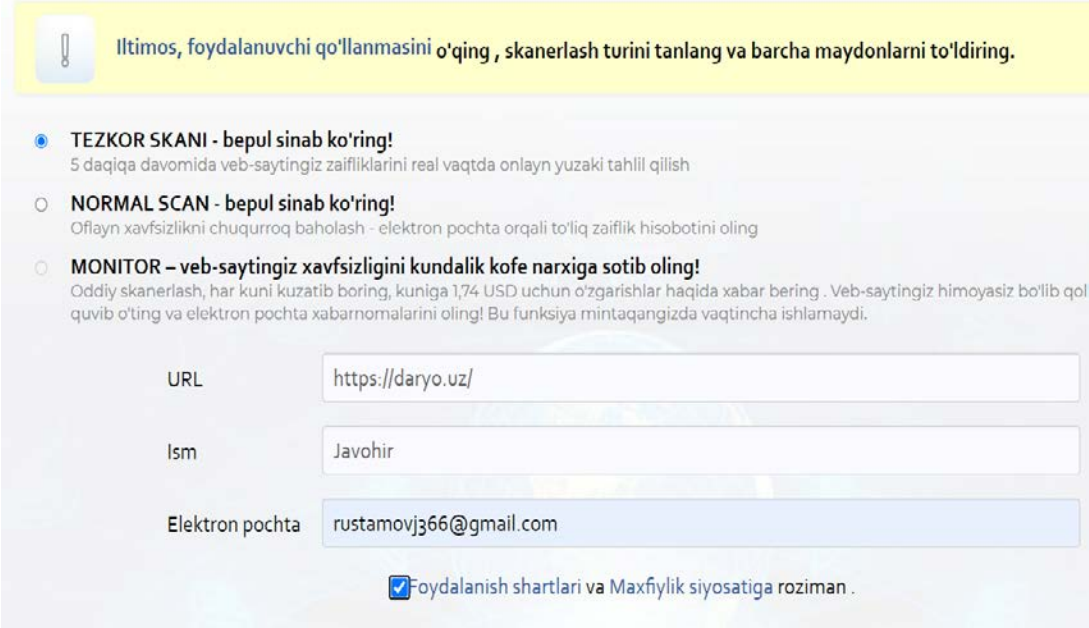
2.1.6-rasm. 1-saytni skaner natijasini electron pochtaga yuborilgan natijasi.

Ko‘rinib turibdiki, tuit.uz sayti xavfsiz va unda hech qanday zararli dasturlar mavjud emas. Skanerlanerlash muavafaqiyatli amalga oshirildi.

Daryo.uz sayti misolida ko‘rib chiqamiz:

1-qadam https://service.h-x.technology/scan link orqali H-X Scanner

platformasiga kiriladi. Natijada yana quyidagicha oyna hosil bo‘ladi.



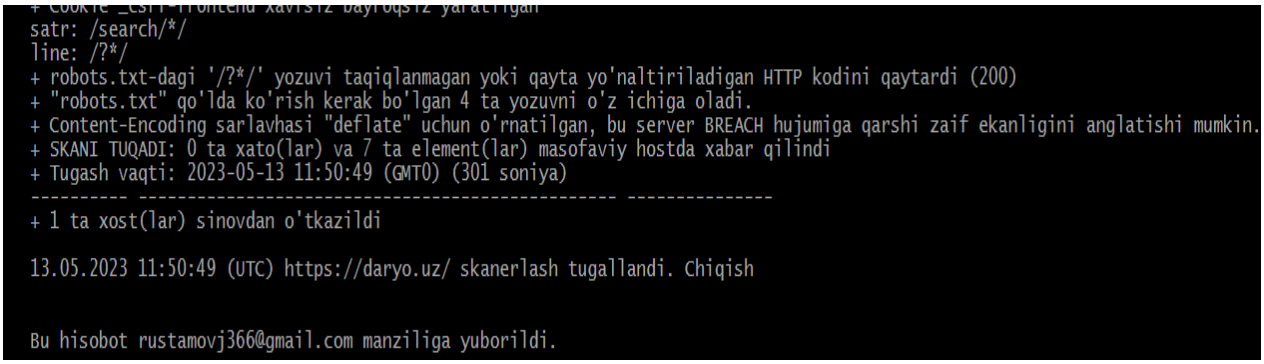
2.1.7-rasm. Daryo.uz veb-saytini tekshirish.

2-qadam: Natijada, *daryo.uz* sayti skanerlash boshlanadi. Bu skanerlash usuli sayt haqida barcha ma’lumotlarni tekshirib ekranga chiqaradi. Xuddi *tuiz.uz* sayti qanday skaner qilingan bo‘lsa, bu safar ham xuddi shu ish ketma-ketligi takrorlanadi

2.1.8-rasm. 2-saytni skaner qilish jarayoni.

3-qadam skanerlash natijasi taqdim etiladi. Bundan tashqari kiritgan gmail ga

skanerlash natijasi yuboriladi.



2.1.9-rasm. 2-saytni skaner natijasi.

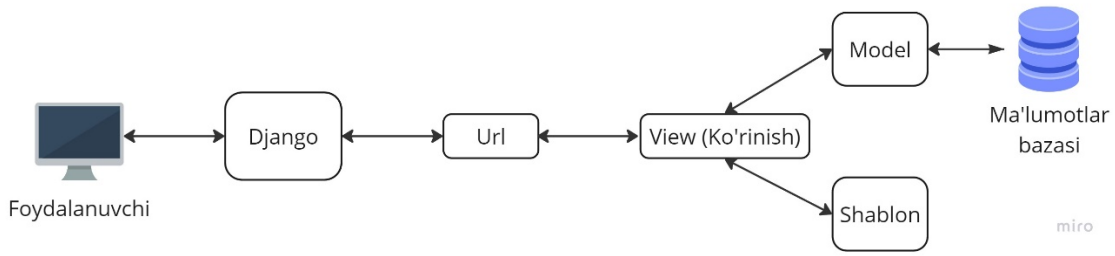
Xulosa o‘rnida shuni aytish kerakki *tuit.uz* veb-sayti *daryo.uz* saytiga qaraganda xavfsizlik siyosati anchagina yuqori. Buni skanerlash jarayonidagi skanerdan muvoffaqiyatli o‘tgan natijalar orqali ham bilib olsak bo‘ladi.

2.2. Veb saytlarni yaratishda xavfsiz kodlash usullari va algoritmlari

Ushbu bo‘limda Djnago frameworki yordamida xavfsiz veb-sayt tuzishni ko‘rib chiqamiz. Django python dasturlash tilinig web-saytlar qurish uchun yozilgan framework hisoblanadi. Python sintaksis jihatdan ancha tushunarliligi bilan boshqa dasturlash tillaridan ajralib turadi. Qo‘llanishlishida ham ancha qulay shu sababli loyihani tezro tugallash uchun Django tanlandi. Django 2003 yilning kuzida, Lawrence Journal-World gazetasining veb-dasturchilari Adrian Xolovati va Saymon Uillison ilovalar yaratish uchun Python-dan foydalanishni boshlaganlarida yaratilgan.

MVT – bu Django veb-ramka tomonidan qo‘llaniladigan dizayn namunasi. Model ilovaning ma’lumotlari va biznes mantiqini ifodalaydi. View MVC naqshidagi “controller”ga o‘xshash ma’lumotlarning foydalanuvchiga qanday taqdim etilishini aniqlash uchun javobgardir. Template ko‘rinish qatlamini ifodalaydi va MVC naqshidagi “View”ga o‘xshash ma’lumotlar qanday ko‘rsatilishi kerakligini aniqlash uchun javobgardir. Ma’lumotlar bazasi sifatida PostgreSql ma’lumotlar bazasini

tanlandi.



2.2.1-rasm. MVT arxitekturasi.

Ma’lumotlar bazasi asosiy turga bo‘linadi bular:

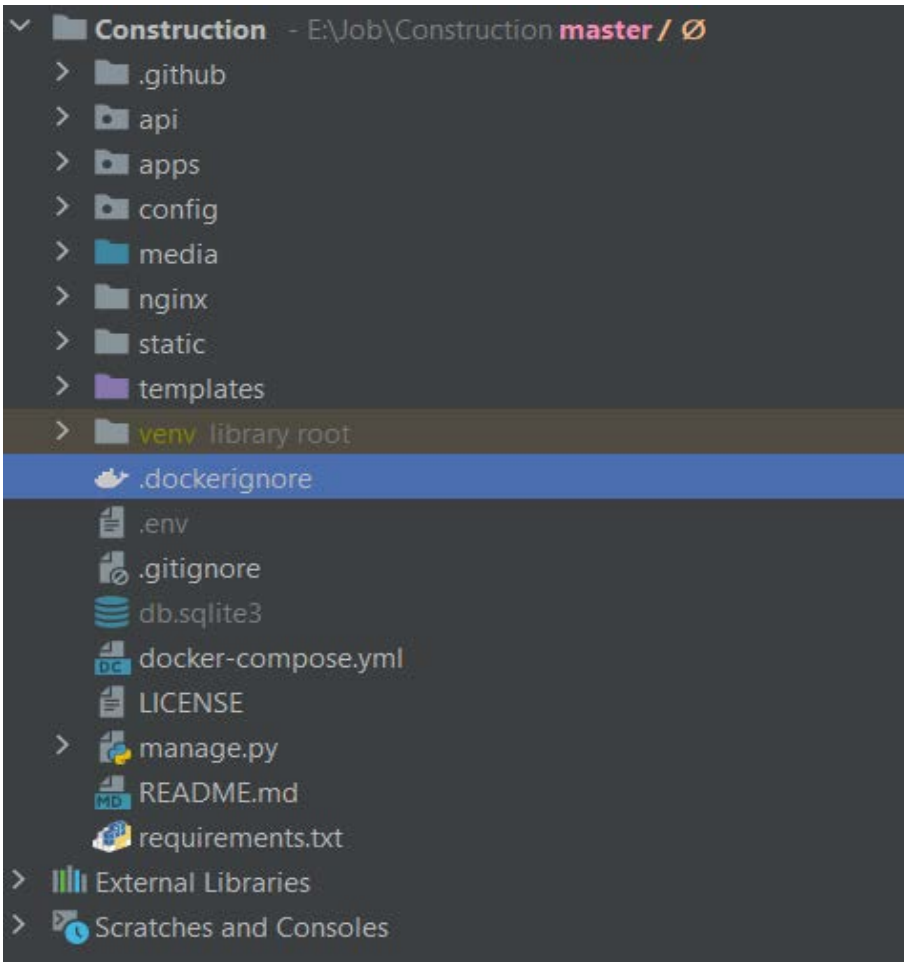
– Relations databases;

– NoSQL databases.

Relyatsion ma’lumotlar bazalari oldindan belgilangan ustunlar va ular orasidagi munosabatlarga ega jadvallardan foydalangan holda ma’lumotlarni tizimlashtirilgan tarzda saqlaydi. Ushbu ma’lumotlar bazalari sxema bo‘yicha ishlaydi va ma’lumotlarni so‘rash uchun SQL (Tuzilgan so‘rovlar tili) dan foydalanadi. Ular murakkab so‘rovlarni bajarish qobiliyati, yuqori tranzaksiya hajmlari va ma’lumotlarning izchilligi bilan mashhur.

Veb server uchun Nginx datur ilovasi tanlab olindi NGINX va Apache ikkalasi ham veb-kontentga xizmat ko‘rsatish uchun ishlatilishi mumkin bo‘lgan mashhur veb-serverlardir, lekin ular dizayni, xususiyatlari va foydalanish holatlarida farqlanadi.

NGINX hodisaga asoslangan arxitekturadan foydalanadi, bu unga xotiradan kam foydalanish va yuqori samaradorlik bilan ko‘p sonli so‘rovlarni bajarish imkonini beradi. U HTTP, HTTPS, SMTP, POP3 va IMAP kabi keng ko‘lamli protokollarni qo‘llab-quvvatlaydi va HTML fayllari, tasvirlar va veb-ilovalar kabi 46tatic va dinamik tarkibga xizmat ko‘rsatish uchun ishlatilishi mumkin.



2.2.2-rasm. Dastur kodining tuzilish strukturasi.

Dastur ishlashi uchun kerak bo‘lgan lekin maxfiyligi saqlanishi shart bo‘lgan parametrlar .env faylida saqlandi. Env fayli odatda API kalitlari, ma’lumotlar bazasi hisob ma’lumotlari va ishlab chiqish, sinovdan o‘tkazish yoki ishlab chiqarish kabimuayyan muhitga xos bo‘lgan boshqa konfiguratsiya sozlamalari kabi nozik ma’lumotlarni o‘z ichiga oladi. Umuman olganda, .env fayli dasturiy ta’minotni ishlab chiqish loyihalarida atrof-muhitga xos konfiguratsiya ma’lumotlarini boshqarish uchun foydali vosita bo‘lib, turli dasturlash tillari va ramkalarida keng qo‘llaniladi.

2.2.3-rasm. Maxfiy ma’lumotlarni saqlash usuli.

Dasturiy taminotimizga xavfsizlik choralarini o‘rnatib boramiz. Eng birinchi navbatda DDOS hujumlarga bardoshlilik siyosatini o‘rnatamiz.

1-qadam. Bunda daqiqasiga jo‘natuvchi so‘rovlar sonini 100 ta qilib belgilaymiz. Hamda bloklangan foydalanuvchilarning saytga qayta murojaat qilish (blokdan ochilishi) muddatini 5 soat qilib belgilaymiz.

```
1 usage  rustamovjavohir *
class DDOSMiddleware:
    rustamovjavohir
    def __init__(self, get_response):
        self.get_response = get_response

    rustamovjavohir *
    def __call__(self, request):
        # So'rov jo'natish tezligini o'rnating (daqiqasiga so'rovlar soni)
        rate_limit_threshold = 100

        # So'rovning IP manzilini oling
        ip_address = request.META.get('REMOTE_ADDR')

        # IP-manzil gora ro'yxatga kiritilganligini tekshiring
        if cache.get(f'ddos:blacklist:{ip_address}') or BlackIps.objects.filter(ip=ip_address,
                                                                                is_active=True)
            return HttpResponseForbidden("IP manzilingiz shubhali faoliyat tufayli bloklandi.")
```

2.2.4-rasm. DDOS hujumlarni oldini oluvchi dastur kodi

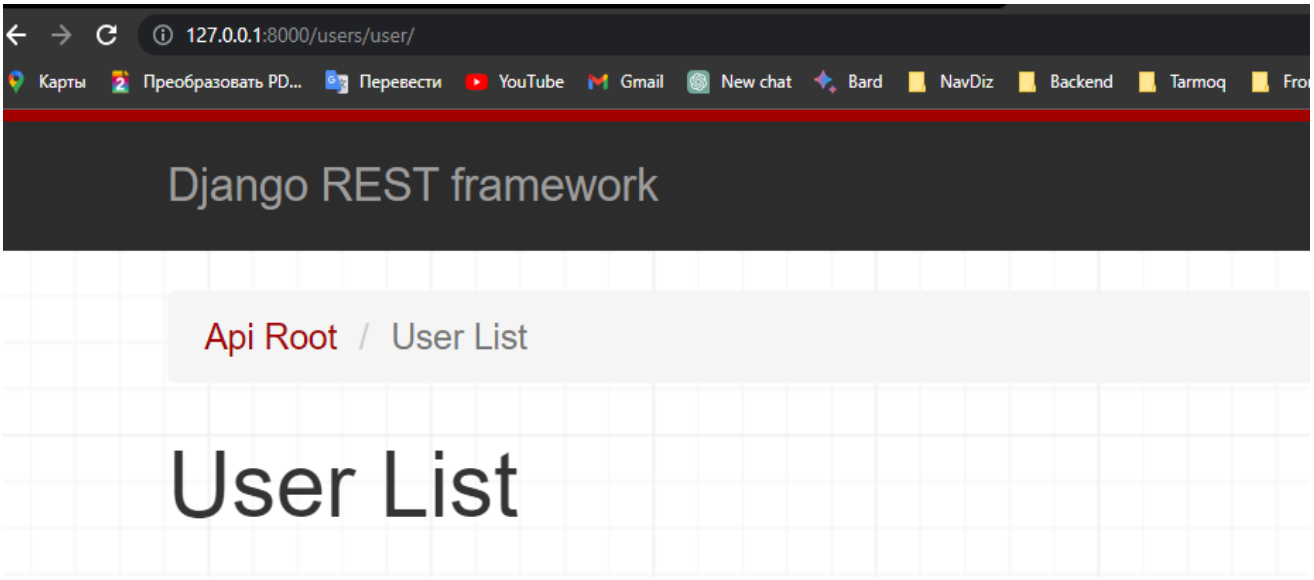
2-qadam. Yaratgan middlewareni ishlashi uchun settings qismiga qoshib

qo‘yamiz. Izoh uchun middlewarelar joylashish tartibi bo‘yicha ishlaydi.

```
MIDDLEWARE = [
    'django.middleware.security.SecurityMiddleware',
    'django.contrib.sessions.middleware.SessionMiddleware',
    'corsheaders.middleware.CorsMiddleware',
    'django.middleware.common.CommonMiddleware',    Rustamovjavohir, 1
    'django.middleware.csrf.CsrfViewMiddleware',
    'django.contrib.auth.middleware.AuthenticationMiddleware',
    'django.contrib.messages.middleware.MessageMiddleware',
    'django.middleware.clickjacking.XFrameOptionsMiddleware',
    'config.middlewares.TelegramErrorMiddleware',
    'ddos.middleware.DDOSMiddleware',
]
```

2.2.5-rasm. Dastur kodini o‘rta dasturlar qatoriga qo‘shish

3-qadam. Saytni ishlatib yuboramiz hamda ishlayotganini sinab ko‘ramiz

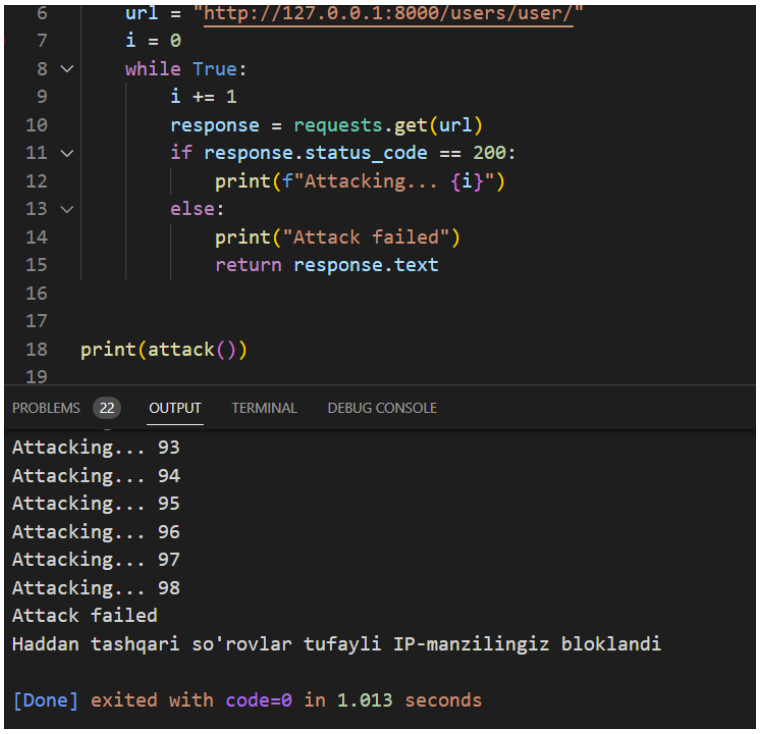


2.2.6-rasm. Sayt ishlayotganini tekshirish.

4-qadam. Sun’iy DDOS hujum qiluvchi dastur kodi yozamiz va uni ishlatib yuboramiz

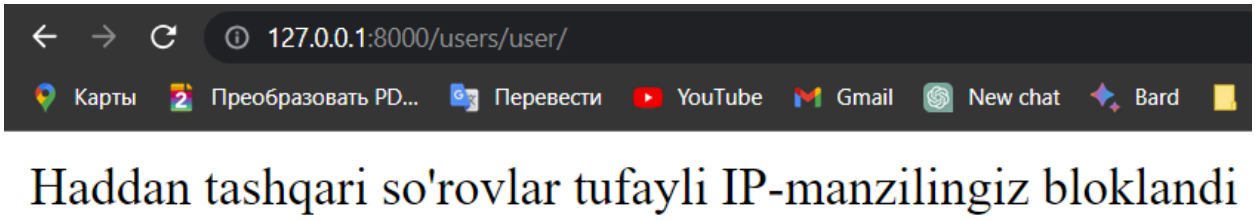
2.2.7-rasm. Sun’iy DDOS hujumini amalga oshiruvchi dastur kodi.

5-qadam. DDOS hujum jarayonini kuzatib boramiz.



2.2.8-rasm. Sun’iy DDOS hujumini amalga oshish jarayoni.

6-qadam Natijani ko‘rish uchun sahifamizga qaytamiz. Ko‘rib turganingizdek IP addresimiz 5 soat davomida blokka tushdi. Blokka tushgan IP addreslarimizni saytimiz foydalanishga ruhsat bermaydi.



2.2.9-rasm. Dastur tomonidan foydalanuvchi bloklangan holat.

Hozir biz dasturiy taminotimiz yordamida DDOS hujumlardan himoyalanishni ko‘rib chiqdik. Himoyalanishni yanada mukkamallashtirish uchun himoya siyosatini server qismida ham qo‘llashimiz kerak. Bu veb-saytimizni yanada ishonchli bo‘lishini ta’minlaydi. Bunga misol qilib quyidagilarni keltirish mumkin.

Ulanishlarni cheklash: Alohida IP manzillar yoki diapazonlardan ulanishlar sonini cheklash uchun limit_conn va direktivalaridan foydalaning . limit_conn_zone

bu bitta manbadan ortiqcha ulanishlarning oldini olishga yordam beradi, bu DdoS hujumlarining umumiy xususiyati. Masalan:

NGINX_conf_for_DDOS

```
http {
  ...
  limit_conn_zone $binary_remote_addr zone=conn_limit_per_ip:5m;
  ...
  server {
    ...
    limit_conn conn_limit_per_ip 5;
    ...
  }
}
```

2.2.10-rasm. DDOS hujumlardan NGINX orqali himoyalanish.

2.3. OWASP standartlaridan foydalanib veb saytlarni kiberhujumlardan himoya qilish choralarini samaradorligini oshirish algoritmlari

OWASP (Open Web Application Security Project) - ochiq veb-ilovalar xavfsizligi loyihasini anglatadi. Bu notijorat tashkilot bo‘lib, veb-ilovalar va dasturiy ta’minot xavfsizligini yaxshilashga qaratilgan. OWASP ishlab chiquvchilar, xavfsizlik bo‘yicha mutaxassislar va tashkilotlarga veb-ilovalar xavfsizligining eng muhim xavflarini tushunish va hal qilishda yordam berish uchun qimmatli manbalar, ko‘rsatmalar va vositalarni taqdim etadi.

OWASP ning asosiy maqsadi umumiy xavfsizlik zaifliklari haqida xabardorlikni oshirish va ularni oldini olish yoki yumshatish bo‘yicha ko‘rsatmalar berish orqali veb-ilovalarni yanada xavfsizroq qilishdir. Tashkilot dastur xavfsizligini yaxshilash bo‘yicha hamkorlik qilish uchun butun dunyo bo‘ylab ekspertlarni birlashtirgan jamiyat tomonidan boshqariladigan loyiha sifatida ishlaydi.

OWASPning eng muhim hissalaridan biri OWASP Top Ten Project loyihasidir. Ushbu loyiha veb-ilovalarda tez-tez uchraydigan o‘nta eng muhim xavfsizlik xavfini

aniqlaydi va ta’kidlaydi. OWASP Top Ten ro‘yxati paydo bo‘layotgan tahdidlar va zaifliklarni aks ettirish uchun vaqti-vaqti bilan yangilanadi. U ishlab chiquvchilar, xavfsizlikni sinovdan o‘tkazuvchilar va auditorlar uchun xavfsizlik bo‘yicha sa’y-harakatlarini birinchi o‘ringa qo‘yish va eng muhim xavflarni bartaraf etishga e’tibor berish uchun mos yozuvlar nuqtasi bo‘lib xizmat qiladi[6].

Mana 2022 yil holatiga ko‘ra joriy OWASP eng yaxshi o‘nta xavfsizlik xavflari:

1. *Buzilgan kirish nazorati* (Broken access control) - Kirish nazorati foydalanuvchilar kirishi mumkin bo‘lgan narsalarni cheklaydi va ularni o‘zlariga tayinlangan ruxsatlar doirasidagi resurslar bilan cheklaydi. Kirish nazorati buzilishi odatda foydalanuvchilarning boshqa faoliyatlar qatorida ularga tayinlanganidan farqli ruxsatlarni talab qiluvchi biznes funktsiyalarini bajarishiga olib keladi. Muvaffaqiyatsizlik shuningdek, ma’lumotlarning ruxsatsiz oshkor etilishi, o‘zgartirilishi yoki ma’lumotlarning yo‘q qilinishiga olib keladi;

Kirish nazorati zaifliklariga quyidagilar kiradi:

– barcha foydalanuvchilarga ma’lum rollar, foydalanuvchilar yoki ruxsat guruhlari uchun mo‘ljallangan resurslarga kirish huquqini berish orqali eng kam imtiyoz tamoyilini buzish;

– URL o‘zgartirish, ichki ilova holatini o‘zgartirish yoki HTML sahifani o‘zgartirish orqali kirish nazoratini chetlab o‘tish. API hujum vositasidan kirishni boshqarish tekshiruvlarini chetlab o‘tish uchun ham foydalanish mumkin;

– PUT, POST yoki DELETE uchun API kirish boshqaruvlari etishmayapti;

– imtiyozni oshirish, bunda tajovuzkor tizimga kirmasdan foydalanuvchi sifatida kirish huquqiga ega yoki pastroq imtiyozli foydalanuvchi hisobidan administrator darajasidagi funktsiyalarni bajarishi mumkin;

Oldini olish:

Kirish nazorati bilan resurslarga kirishni cheklang. Kirish nazorati ishonchli server muhitida ishlaydi, bu erda ma’lumotlar tajovuzkor tomonidan o‘zgartirilmaydi. Himoya usullari quyidagilarni o‘z ichiga oladi:

– agar ob’ekt ommaviy resurs bo‘lmasa, kirish sukut bo‘yicha rad etilishi kerak;

– ilova davomida kirishni boshqarish mexanizmlaridan qayta foydalaning;

– ilova biznes cheklovlarini qo‘llash;

– veb-serverlar uchun katalog ro‘yxatini o‘chirib qo‘ying;

– rate-limit API va kontrollerga kirish.

2. Kriptografik xatolar (Cryptographic failures) - Kriptografik nosozliklar kriptografiyadagi buzilish yoki etishmovchilikning keng belgisi bo‘lib, tizimning buzilishiga yoki nozik ma’lumotlarning ta’siriga olib kelishi mumkin. Shaxsiy identifikatsiya qilinadigan ma’lumotlar va kredit karta raqamlari qo‘shimcha himoyani talab qiladigan ma’lumotlar turlaridan biridir. Ma’lumotlarni himoya qilish usullari ma’lumotlar turiga va uning ma’lumotlarning maxfiyligi to‘risidagi qonunlarga bo‘ysunishi yoki yo‘qligiga qarab belgilanadi, masalan, Evropa Ittifoqining ma’lumotlarni himoya qilish bo‘yicha umumiy reglamenti (GDPR);

Kriptografik xatolarga quyidagilar kiradi:

− HTTP xavfsizlik sarlavhalari mavjud bo‘lmagan brauzerda majburiy shifrlash;

− yaroqsiz server sertifikati ishonch zanjiri;

− FTP, SMTP va HTTP kabi brauzer protokollari orqali aniq matnda uzatiladigan ma’lumotlar;

− zaif kriptografik algoritmlar va protokollarni qo‘llash, eski koddagi zaif kriptografik algoritmlarga e’tibor bermaslik.

Oldini olish:

Kriptografik nosozlikning oldini olish dasturning funksionalligi va foydalanilgan ma’lumotlar turiga bog‘liq. Ma’lumotni to‘gri himoya qilishning ko‘plab jihatlari mavjud. Oldini olish quyidagilarni o‘z ichiga oladi (va OWASP ma’lumotnomalarida keltirilgan yana ko‘p narsalar):

- ilova tomonidan qayta ishlangan, uzatilgan yoki saqlangan ma’lumotlarni tasniflang. Maxfiy ma’lumotlarni amaldagi maxfiylik qonunlari, biznes talablari yoki me’yoriy majburiyatlarga muvofiq tasniflang;
- faqat kerakli ma’lumotlarni saqlang, so‘ngra amal tugagandan so‘ng o‘chirib tashlang;
- uzatish paytida va dam olishda barcha ma’lumotlarni shifrlash;
- nozik ma’lumotlarni tashish uchun eski protokollardan foydalanmang.

3. *In’ektsiya* (Injection) - Injection zaifliklar manba kodini tekshirish orqali aniqlanishi mumkin. Ushbu turkumga saytlararo skript, SQL in’ektsiyasi va XML in’ektsiyasi kiradi. Avtomatlashtirish bu yerda zaifliklarni aniqlash uchun barcha parametrlar va ma’lumotlar kiritishlari sinovdan o‘tganiga ishonch hosil qilish orqali yordam berishi mumkin;

Ilovalar quyidagi hollarda in’ektsiyaga zaifdir:

- foydalanuvchi tomonidan kiritilgan ma’lumotlar tekshirish, tozalash yoki filtrlashsiz qabul qilinadi;
- dushman ma’lumotlar nozik ma’lumotlarni olish uchun ishlatiladi;
- oldini olish:
- so‘rovlar va buyruqlarni ma’lumotlardan alohida saqlash in’ektsiyaga urinishlarning oldini olish uchun juda muhimdir:
- escape sintaksisida foydalanilayotgan tarjimon uchun maxsus belgilar mavjudligiga ishonch hosil qiling;
- kutilmagan kiritish ruxsatsiz harakatlarni amalga oshirishning oldini

olish uchun so‘rovlarni boshqarish vositalaridan foydalaning;

− tarjimondan alohida xavfsiz API orqali parametrlangan so‘rovlar bilan tayyorlangan bayonotlardan foydalaning.

4. *Xavfsiz bo‘lmagan loyihalash* (Insecure design) - Ishonchsiz dizayn ishonchsiz amalga oshirishdan farq qiladi. Xavfsiz dizayn zaifliklarga olib keladigan nomukammal tarzda amalga oshirilishi mumkin. Ishonchsiz dizaynni amalga oshirish orqali tuzatib bo‘lmaydi, chunki dizaynning o‘zida tegishli xavfsizlik nazorati mavjud emas. Dasturiy ta’minot yoki ishlab chiqilayotgan tizim bilan bog‘liq biznes xavfini to‘gri baholamaslik xavfsizlik darajasining etarli emasligiga olib keladi;

− *xavfsizlik noto‘gri sozlanishi* (Security misconfiguration) - Xavfsizlikning noto‘gri konfiguratsiyasiga noto‘gri sozlangan boshqaruv elementlari qatori hamda ilova zaifligiga hissa qo‘shadigan boshqa omillar sabab bo‘lishi mumkin. Ushbu turkumga ko‘plab keng tarqalgan noto‘gri konfiguratsiyalar kiradi:

− bulutli xizmatlar uchun noto‘gri sozlangan ruxsatlar;

− keraksiz ochilgan portlar, xizmatlar yoki noto‘gri ko‘tarilgan imtiyozlarga olib keladigan keraksiz xususiyatlarni yoqish;

− standart hisob qaydnomasiga kirish ma’lumotlari o‘zgartirilmagan.

5. *Zaif va muddati o‘tgan komponentlar* (Vulnerable and outdated components) - Zaifliklar aniqlangan va oshkor qilinganidan keyin ishlab chiqarishda qoladigan yamalmagan va eski komponentlar katta xavf bo‘lishi mumkin. Ilovalar dasturiy ta’minotning so‘nggi versiyasini ishga tushirmasa, himoyasiz bo‘lishi mumkin. Qaysi kutubxona yoki komponent versiyasidan foydalanilayotgani noma’lum bo‘lsa, ilova zaif bo‘lishi mumkin. Zaifliklar tekshirilmagan komponentlar ham xavf ostida bo‘lishi mumkin;

6. *Identifikatsiya va autentifikatsiyadagi xatolar* (Identification and authentication failures) - Autentifikatsiya va identifikatsiya qilishda xatoliklar

foydalanuvchi identifikatori, autentifikatsiya va seans ma’lumotlari foydalanuvchi tizimlar va ma’lumotlarga kirishiga ruxsat berilmaganda sodir bo‘ladi. Ushbu zaif parollarga ruxsat berishni o‘z ichiga oladi, zaif xeshlangan, oddiy matnli parol ma’lumotlari do‘konlaridan foydalanish, va qo‘pol kuch va hisob ma’lumotlarini to‘ldirish kabi avtomatlashtirilgan hujumlarni amalga oshirishi mumkin bo‘lgan botlarga ruxsat berish;

7. *Dasturiy ta’minot va ma’lumotlar yaxlitligini buzish* (Software and data integrity failures) - OWASP ro‘yxatida yangi dasturiy ta’minot va ma’lumotlar yaxlitligidagi nosozliklar CWE hisoblanadi. Ko‘pgina tizimlar yangilanishlarning yaxlitligini tasdiqlamaydigan avtomatlashtirilgan dasturiy ta’minotni yangilash xususiyatlaridan foydalanadi;

8. *Xavfsizlik jurnali va monitoringdagi nosozliklar* (Security logging and monitoring failures) - Xavfsizlik jurnali va monitoring xatoliklari toifasi hujum paytida audit jurnallari va monitoring bilan bog‘liq muammolarga qaratilgan. Xavfsizlik monitoringi va jurnallar faol buzilishni aniqlash va yumshatish uchun zarurdir. Muvaffaqiyatsizliklar quyidagi hollarda yuz beradi:

– tizimga kirish yuqori qiymatga ega bo‘lgan tranzaktsiyalarni, tizimga kirishga urinishlarni va muvaffaqiyatsiz kirish urinishlarini hisobga olmaydi;

– xatolar va ogohlantirishlar noaniq, noto‘gri yoki jurnal yozuvlarini keltirib chiqaradi;

– API va ilovalarda shubhali harakatlar kuzatilmaydi;

– xavfsizlik jurnallari faqat mahalliy sifatida mavjud;

– davom etayotgan hujumlarni aniqlay olmaydigan yoki o‘z vaqtida ogohlantira olmaydigan ilovalar.

9. *Server tomonidagi so‘rovlarni qalbakilashtirish (SSRF)* - Server tomonidagi so‘rovlarni soxtalashtirish toifasi foydalanuvchi uchun qulaylik

xususiyatlarining zaif tomonlariga qaratilgan. Muayyan so‘rovlar kutilmagan manba orqali ilovaga yuborilishi mumkin.

Misol uchun: Siz bankdagi hisob raqamingizg kirmoqdasiz. Bank saytida SSL sertifikat yo‘q yani malumotlarni to‘gridan-to‘gri qabul qiladi va jo‘natadi. Shunda Garazli inson siz va bank o‘rtasidagi paketlar almashinuvini bemalol ko‘roladi. Sizning bank hisobingiz, login parolingiz uning uchun ma’lum bo‘ladi.

2.3.1 jadval

SSL sertifikati bor sayt va sertifikati yo‘q sayt farqlari

№	Xususiyat	SSL sertifikatiga ega veb-sayt (HTTPS)	SSL sertifikati bo‘lmagan veb-sayt (HTTP)
1	Ma'lumotlarni shifrlash	Foydalanuvchi va sayt o‘rtasida almashinadigan ma'lumotlarni xavfsiz shifrlaydi	Ma'lumotlar tinglash uchun sezgir bo‘lgan oddiy matnda uzatilad
2	Xavfsizlik	Xavfsiz ulanishni ta'minlaydi, ma'lumotlarning yaxlitligi va haqiqiyligini ta'minlaydi	Hech qanday shifrlash yoki xavfsizlik choralari mavjud emas
3	Ishonch va foydalanuvchi ishonchi	Xavfsiz ulanishni ko‘rsatuvchi qulf belgisi yoki yashil manzil satrini ko‘rsatadi	Xavfsizlikning vizual belgisi yo‘q, bu foydalanuvchi ishonchsizligiga olib kelishi mumkin
4	SEO ta'siri	Qidiruv tizimlari tomonidan tanlanadi va qidiruv reytinglariga ijobiy ta'sir ko‘rsatishi mumkin	Muayyan SEO afzalligi yo‘q, lekin ma'lum turdagi kontent uchun potentsial reyting jazosi
5	Maxfiylik	Maxfiy ma'lumotlarni himoya qilish orqali foydalanuvchi maxfiyligini yaxshilaydi	Maxfiylikni himoya qilishning yo‘qligi, foydalanuvchi ma'lumotlarini ushlash uchun himoyasiz qoldiradi

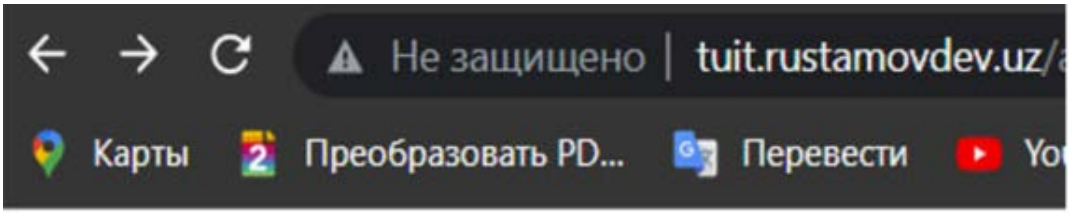
Ilovalar odatda oxirgi foydalanuvchilar uchun vazifalarni almashtirishni

osonlashtirish uchun URL manzillarini olib keladi, ko‘pincha ularni ilovada saqlaydi va olingan URL orqali boshqa xususiyatga kirishni ta’minlaydi. Bulut arxitekturasining tobora ortib borayotgan murakkabligi SSRF yuqori chastotada sodir bo‘lishini anglatadi.

Keling saytlarda uchraydigan xatolarni qanday qilib OWASP da keltirib o‘tilgan eng xavfsizlikka tahdid standartlari yordamida bartaraf etib ko‘ramiz. Bunda Hozirgi kunda dolzarb bo‘gan muammo klient-server o‘rtasidani aloqa kanallarini ochiqli va undan o‘tayotgan ma’lumotlar shifrlanmasdan ketayotganiga e’tibor qaratamiz. Hop shunda oddiy savol tugulishi mumkin nega ma’lumotlarimizni shifrlab jo‘natishimiz kerak, ochiq holatda ketsa nima bo‘ldi? Bu savolga oddiygina qilib javob bersak.

Hop buni qanday oldini olsak bo‘ladi. Avvalambor oldindan tayyorlab qo‘ygan saytimizni serverga joylab olamiz. (Server xizmatini *DigitalOcean* dan olindi)

Sayt domeni *tuit.rustamovdev.uz*



2.3.1- rasm. Veb-brouser tomonidan ogohlantirish.

Ko‘rib turganingizdek veb-brovzer ham sayt xavfsiz emasligini ogohlantirib turibdi. Bunda ma’lumotlarimiz ochiq holda ketmoqda. Bu juda yomon shaxsiy ma’lumotlar ochiq holda ketishi katta xavf olib kelishi mumkin. Buni oldini olish uchun sayt uchun SSL sertifikatini olamiz bizga bunda Let’s Encrypt tekin xizmati yordam beradi. SSL sertifikati asosan pullik hisoblangani sababli shu xizmatni tanladik.

2.3.2-rasm. Ishonchli SSL sertifikat taqdim etuvchi xizmat.

Serverimizga ushbu xizmatdan foydalanishimiz uchun *certbot* kutubxonasini yuklab olamiz. Buyruqlarni kiritib SSL sertifikatini domenimizga biriktirib olamiz

```
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
Congratulations! You have successfully enabled https://tuit.rustamovdev.uz

You should test your configuration at:
https://www.ssllabs.com/ssltest/analyze.html?d=tuit.rustamovdev.uz
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

IMPORTANT NOTES:
 - Congratulations! Your certificate and chain have been saved at:
   /etc/letsencrypt/live/tuit.rustamovdev.uz/fullchain.pem
   Your key file has been saved at:
   /etc/letsencrypt/live/tuit.rustamovdev.uz/privkey.pem
   Your cert will expire on 2023-08-12. To obtain a new or tweaked
   version of this certificate in the future, simply run certbot again
   with the "certonly" option. To non-interactively renew *all* of
   your certificates, run "certbot renew"
 - If you like Certbot, please consider supporting our work by:

   Donating to ISRG / Let's Encrypt:   https://letsencrypt.org/donate
   Donating to EFF:                    https://eff.org/donate-le

root@ubuntu-s-1vcpu-512mb-10gb-fra1-01-4-dollor:~#
```

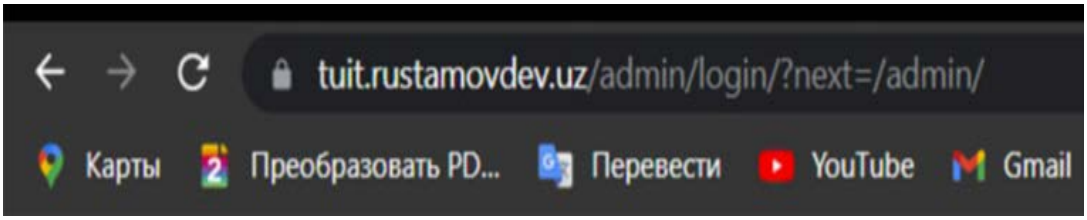
2.3.3-rasm. Let’s Encrypt yordamida SSL sertifikatini olish jarayoni.

Natijada quyidagi matni olamiz. SLL sertifikatni nginx konfiguratsiya faylimizdan tekshirib olsak bo‘ladi.

```
server {
    server_name tuit.rustamovdev.uz www.tuit.rustamovdev.uz;

    location = /favicon.ico { access_log off; log_not_found off; }
    location /static/ {
        root /home/tuit/Construction;
    }

    location / {
        include proxy_params;
        proxy_pass http://unix:/run/gunicorn.sock;
    }

    listen 443 ssl; # managed by Certbot
    ssl_certificate /etc/letsencrypt/live/tuit.rustamovdev.uz/fullchain.pem; # managed by Certbot
    ssl_certificate_key /etc/letsencrypt/live/tuit.rustamovdev.uz/privkey.pem; # managed by Certbot
    include /etc/letsencrypt/options-ssl-nginx.conf; # managed by Certbot
    ssl_dhparam /etc/letsencrypt/ssl-dhparams.pem; # managed by Certbot

}
server {
    if ($host = tuit.rustamovdev.uz) {
        return 301 https://$host$request_uri;
    } # managed by Certbot


    listen 80;
    server_name tuit.rustamovdev.uz www.tuit.rustamovdev.uz;
    return 404; # managed by Certbot
```

2.3.4-rasm. SSL sertifikatini serverga o‘rnatish jarayoni.

Veb-saytimizga qaytamiz ko‘rib turganingizdek natija ajoyib. Saytimiz SSL sertifikati orqali malumotlarni shifrlab jo‘natib qabul qilmoqda.



2.3.5-rasm. SSL sertifikati o‘rnatilgan sayt.

Xulosa o‘rnida shuni aytish mumkinki OWASP da keltirib o‘tilgan kiberxavfsizlik kamchiliklaridan birini hal qildik.

II bob bo‘yicha xulosalar

Dissertatsiya ishining ikkinchi bobining 1-bo‘limida Internet makonida Veb saytlarga uyushtiriladigan hujum turlari tahlili amalga oshirildi. ko‘plab davlat idoralari, korparatsiyalar va jismoniy shaxslarning ma’lumotlariga nisbatan amalga oshirilgan hujumlar tarixi tahlil qilindi. Saytlararo skript yaratish, SQL Injection, Saytlararo so‘rovlarni qalbakilashtirish, Taqsimlangan xizmat ko‘rsatishni rad etish, Shafqatsiz kuch hujumlari, Fayllarni qo‘shish ekspluatatsiyasi, Server tomonidagi so‘rovlarni soxtalashtirish, Clickjacking, Remote Code Execution, Fishing hujumlari kabi usullar haqidagi ma’lumotlar tahlil qilindi.

2-bo‘limda Veb saytlarni yaratishda xavfsiz kodlash usullari va algoritmlari o‘rganildi. Maxfiy ma’lumotlarni saqlash usuli, DDOS hujumlarni oldini oluvchi dastur kodi keltirilgan. Sun’iy DDOS hujumini amalga oshiruvchi dastur kodi yordamida Sun’iy DDOS hujumini amalga oshish jarayoni amalga oshirildi va dastur tomonidan foydalanuvchi bloklangan holat keltirib o‘tildi. DDOS hujumlardan NGINX orqali himoyalanish usuli keltirildi.

3-bo‘limda OWASP standartlaridan foydalanib veb saytlarni kiberhujumlardan himoya qilish choralarini samaradorligini oshirish algoritmlari keltirildi. Kirish nazorati zaifliklari, Kriptografik xatolar, Kriptografik nosozlikning oldini olish choralari, ilovalarning qaysi hollarda in’eksiyalarga zaifligi keltirildi. Saytlarda uchraydigan xatolarni qanday qilib OWASP da keltirib o‘tilgan eng xavfsizlikka tahdid standartlari yordamida bartaraf etish usullari tahlil qilindi.

III BOB. KOMMUNAL TO‘LOVLAR XIZMATI VEB-DASTURIY TA’MINOTI MISOLIDA INTERNET TARMOG‘IDA MA’LUMOTLAR XAVFSIZLIGINI TA’MINLASH

3.1. Kommunal to‘lov tizimlari ma’lumotlarini bazada himoyalash modellari va usullari

Tahdidlarni quyidagicha umumiy yechimlar orqali xal qilish mumkin:

Ma'lumotlar bazalarini himoyalash yechimlarini tasniflash - Ma'lumotlar bazasi xavfsizligi uchun dasturiy yechimlarning oltita asosiy toifasi mavjud. Ularning barchasi muayyan vazifalarni bajarishga mo‘ljallangan.

*O‘lchash va baholash vositalari***.** Ma'lumotlar bazasi zaifliklarini va muhim ma'lumotlarni joylashishni aniqlash.

Kirish huquqlarini boshqarish vositasi. Maxfiy ma'lumotlarga ortiqcha kirish huquqlarini aniqlash.

Monitoring va blokirovkalash vositalari. Ma'lumotlar bazalarini buzish, ruxsatsiz kirish va ma'lumotlar o‘g‘irlashdan himoya qilish.

Audit vositalari. Ma'lumot tizimining sanoat xavfsizligi standartlariga muvofiqligini tasdiqlashga yordam bering.

Ma'lumotlarni himoya qilish vositalari. Ma'lumotlarning yaxlitligi va maxfiyligini ta'minlash.

Texnikaviy bo‘lmagan xavfsizlik choralari. . Internet makonida veb-saytlar xavfsizligi muammolari, internet tarmog‘i veb-saytlari uchun xavfsizlik choralarini ishlab chiqish, ma’lumotlarni himoya qilishning model, usul va algoritmlari.

Nega aynan Oracle Ma’lumotlar bazasini tanladim - Oracle ma'lumotlar bazasining ustunliklari va foydalari:

MBBTlar birinchi prototiplari 70-yillarda paydo bo‘lgan. Biroq, bunday tizimlarning samaradorligi past edi va ularning rivojlanishi e'tiborga olinmadi.

Biroq, 90-yillarning boshida MBBT jahon bozorida yetakchi o‘rinlardan birini egalladi. Dastlabki ma'lumot tizimlaridan biri "System R" edi. Oracle dasturchilarini ilhomlantirgan edi. Qizig‘i shundaki, System R o‘zini o‘zi rivojlantira olmadi va kelajakda o‘zini yoqotib qo‘ydi.

Larry Ellinson tomonidan yaratilgan ushbu MBBTni yaratish g‘oyasi. 77-yilda Yale universitetining yosh talabasi tashqariga chiqib, o‘z biznesini tuzishga qaror qildi. O‘sha paytda cho‘ntagida 1200 dollardan ortiq pul bo‘lmagan edi. U Bob va Edning do‘stlariga loyihaga sarmoya kiritishni so‘radi, lekin faqatgina 500 dollar oldi. O‘shandan beri eng ommalashgan ma'lumotlar bazasi tarixi boshlandi.

Oracle va MySql ma'lumotlar bazalarida muayyan masalalarni muhokama qilishingiz kerak bo‘lsa, u holda Oraclening ixtisoslashgan ijtimoiy tarmog‘i mutaxassislarini jalb qilishning eng oson yo‘li.Oracle boshqa MBBT lar orasida yetakchi bo‘lganligi, hech bo‘lmaganda, 2007 yilda olingan ma'lumotlar bo‘yicha, MBBT ushbu sanoatning jahon bozorining 47% ni qoplaganligi haqida gapiradi.

Ushbu MBBTning afzalliklari juda ko‘p. Oracle eng katta ma'lumotlar bazalarini qo‘llab-quvvatlaydi. Ushbu tizim uchun foydalanuvchilarning ko‘pligi ham to‘siq emas. MBBT har qanday miqdordagi har qanday foydalanuvchilarni bir vaqtning o‘zida turli vazifalarni bajarishga ko‘maklasha oladi. Oracleda turli xil ma’lumotlar turlari o‘rtasida raqobat yo‘q. Oracle MBBT operatsiyalarni(tranzaksiyalarni) yaxshi boshqaradi. Tizim yuqori ish faoliyatini ta’minlaydi, natijada foydalanuvchilar past ishlov berish tezligidan aziyat chekmaydi.

Tizim yuqori darajadagi tayyorgarlikka ega. Turli tizimlarda Oracle ish vaqti individualdir. Misol uchun, ba'zi bir tizimda soatlab ishlash mumkin. Bunday holda, ma'lumotlar bazasini qaytarib olish yoki har qanday tizim xatosi bazaning ishlashini to‘xtatib qo‘ymaydi.Ushbu tizim mahalliy(lokal) boshqaruvga ega. Masalan, muayyan ilovadan ma'lumotni qayta tiklash uchun butun tizimni o‘chirib qo‘yish

kerak emas. Istalgan dasturga kirishni o‘chirish uchun administrator bo‘lish kifoya qiladi va u bilan kerakli manipulyatsiya amalga oshiradi.

Ushbu MBBTni osongina bitta OT dan boshqasiga o‘tkazish mumkin. Oracle uchun maxsus yaratilgan dasturlar osongina minimal o‘zgarishlarga ega bo‘lgan har qanday operatsion tizimga, ba'zan esa ularsiz ko‘chiriladi.

Oracle Ma’lumotlar bazasining o‘z avtomatik xavfsizlik konfiguratsiyalari.

Ma'lumotlar bazasi administratorlarining keng tarqalgan noto‘g‘ri tushunchasi. Oracle ma'lumotlar bazasini xavfsizlik devorlari bilan himoya qilish, hujumlarga mutlaqo to‘siq bo‘lishini ta'minlaydi. Tabiiyki, xavfsizlikka tahdidlar har doim tashqarida, lekin haqiqiy statistik ma'lumotlarga qaraganda, xavfsizlikka oid ko‘plab qonunbuzilishlar ichidan kelib chiqqan. Shu sababli «dadil-himoya» autentifikatsiya qilish siyosati va ma'lumotlar uzatishning oqilona siyosati kuzatilishi kerak.

Oracle ma'lumotlar bazasining xavfsizligini mustahkamlash uchun siz bir necha asosiy qadamlarni qabul qilishingiz mumkin. Ularning ko‘pchiligi aql-idrokka asoslangan va ma'lumotlar bazasiga ma'lum bo‘lgan bo‘shliqlar orqali xakerlik va kirishni oldini oladi. Ushbu xavfsizlik ko‘rsatmalariga birin ketin nazar solamiz.

Avtomatik xavfsiz konfiguratsiya - Yangi ma'lumotlar bazasida yangi Oracle Database 12c xavfsiz konfiguratsiya parametrlarini tanlaganingizda, quyidagi xavfsizlik xususiyatlari yoqilgan:

*Parolga bog‘liq xavfsizlik sozlamalari.*Ma'lumotlar bazasi parolni foydalanuvchilarga tayinlangan standart parol profilida o‘rnatilgan murakkablikni tekshirishni faollashtiradigan boshqa parol bilan bog‘liq siyosatlarni to‘ldirish va amalga oshirishga majbur qiladi.

*Audit.*Standart ma'lumotlar bazasi muayyan ruxsatlarni auditini faollashtiradi. Ma'lumotlar bazasi bilan bog‘lanish vakolati kabi ushbu ruxsatnomalar ma'lumotlar bazasi xavfsizligi uchun muhim hisoblanadi. Odatiy bo‘lib, ma'lumotlar bazasi AUD

$ jadvalidagi audit yozuvlarini saqlaydi va boshlash parametrini audit_trail-ni jb-ga o‘rnatadi.Avtomatik ravishda xavfsiz konfiguratsiya hizmatidan foydalanish ma'lumotlar bazasini Internet Security (MDH) markazi benchmark testlari ko‘rsatkichlari bo‘yicha tavsiya etilgan xavfsizlik tavsiflariga mos kelishini ta'minlaydi.

g
‘
ө
ȯ
ȯ
ȯ
ǵ
ǵ
ȯ‘proq vakolatga ega foydalanuvchilar tomonidan buzish mumkin emas.
ȯ‘q qilish huquqi. Bu muammoni ob'ektdan foydalanuvchilarga to‘g‘ridan-to‘g‘ri
o
‘
ǵ
‘ *P*
U

```
SQL> SELECT COUNT(*) FROM dba_tab_privs
2 WHERE grantee='PUBLIC';
COUNT(*)
-------------
12814
SQL>
```

ȯ PUBLIC vazifasiga berilgan 12000 dan ortiq ob'ekt ruxsatnomalarining 100 ȯ dan oshiq qismi DBMS_JOB, DBMS_METADATA, DBMS_SNAPSHOT,
‘
o
‘
g

DBMS_DDL, DBMS_SPACE va DBMS_OBFUSCATION_TOOLKIT kabi DBMS paketlarini bajarish vakolatiga ega. Barcha muhim ijro etuvchi hokimiyatni PUBLIC vazifasidan chiqarib tashlang. Foydalanuvchilarga rollarni diqqat bilan ishlatish orqali muhim vakolat bering.

SYSDBA imtiyozlari foydalanuvchini juda katta imkoniyatlarga ega, jumladan, ma'lumotlar bazasi ob'ektlarini yo‘q qilish va ma'lumotlar lug‘ati jadvallarini o‘zgartirish. Albatta, SYSDBA vakolatlari juda ehtiyotkorlik bilan berilishi kerak.

*Ko‘p Ma’lumotlar Bazalari Administratorlari muhitlari***.** Tashkilotingizdagi yagona Oracle ma'lumotlar bazasi ma'muri(administrator) bo‘lsangiz, ma'lumotlar bazasini boshqarish uchun barcha tizim imtiyozlariga ega bo‘lishingiz kerak. Biroq, siz bir necha ma'lumotlar bazalarini boshqaradigan Oracle DBA(Database Administrator) guruhiga ega bo‘lsangiz, ularning har biriga bir xil turdagi vakolatlar (masalan, SYSDBA) va bir xil turdagi rol (DBA kabi) berilmasligi kerak. O‘zingizning maxsus rollaringizni yaratishingiz kerak, ularning har biri muayyan ma'lumotlar bazasini boshqarish vazifalarini hal qilish uchun muayyan vakolatlar to‘plamini o‘z ichiga olishi kerak. Natijada, ishlab chiquvchilarga yangi ob'ektlarni yaratishga yordam berish uchun mas'ul bo‘lgan DBA, muayyan zamonaviy ishlarni bajarishga qodir emas va aksincha. Ushbu rollar DBAsga topshirilishi mumkin, shu bilan birga vazifalarni aniq ajratish imkonini beradi.

Ma'lumotlar lug‘atini muhofaza qilish - Har qanday tizimga ruxsat berilgan foydalanuvchilar ma'lumotlar lug‘ati jadvallarini o‘chirib tashlashi mumkin. Ma'lumotlar lug‘atini himoya qilish uchun parametr faylidagi 07_DICTIONARY_ACCESSIBILITY konfiguratsiya parametri FALSE ga o‘rnatilgan bo‘lishi kerak. Bu har qanday vakolatni faqat SYSDBA vakolatiga ega bo‘lgan foydalanuvchilarga berishni cheklaydi.

Ruxsatlarni o‘rnatish - Operatsion tizim darajasida tegishli fayl ruxsatini belgilang, chunki xavfsizlik kamchiliklari ko‘pincha ushbu darajada mavjud bo‘lishi mumkin. UNIX tizimlarida ko‘pincha yaratilgan faylga rw-rw-rw ruxsati beriladi. Bu shuni bildiradiki, UNIX serveriga avtorizatsiya qilingan har qanday foydalanuvchilar ma'lumotlar bazasi fayllari, shu jumladan barcha fayllarni o‘qishi yoki nusxalashi mumkin. UMASK o‘zgaruvchisi 022 ga o‘rnatilishi kerak, shunda faqat Oracle foydalanuvchi nomi ma'lumotlar bazasi fayllarini o‘qish va yozish huquqiga ega.Barcha Oracle fayllaridan darhol SETUIDni olib tashlang. UNIX tizimidagi ba'zi SETUID fayllari buyruq fayllarini imtiyozli foydalanuvchi nomidan ishlashga ruxsat berishi mumkin.

UTL_FILE to‘plami PL / SQL dasturidan operatsion tizim fayllariga yozish imkonini beradi. UTL_FILE_DIR parametrlaridan foydalanganda hech qachon * belgini qiymati sifatida foydalanmang, ya'ni fayllar OS fayl tizimining istalgan katalogiga chiqishi mumkin. UTL_FILE chiqdi fayllaridan butunlay ajralib turadigan ba'zi mashhur joylarga bunday kataloglarning chegarasini cheklash.

Agar zarurat bo‘lmasa, PL / SQL da EXTPROC funksiyasini o‘chirib tashlang. Birinchidan, serverdagi listener.ora faylida va mijozdagi tnsnames.ora faylida EXTPROC murojaatlarni o‘chirib tashlang. Shundan so‘ng, EXTPROC executable fayllari $ ORACLE_HOME / bin katalogidan o‘chirilishi mumkin.

Odatda, tizimda ikkita executable fayl mavjud - extproc va extproc0. EXTPROC funktsiyasi xakerlarni operatsion tizimga autentifikatsiya qilinmasdan kirib borish qobiliyatiga ega. Agar EXTRROC funksiyasi hali ham talab etilsa, Oracle ning MetaLink veb-saytida (http://metalink.oracle.com) 175429.1 Eslatmasini ko‘rib chiqing.

Odatdagi foydalanuvchilarning fayllar menejeriga eksport va import uchun kirishiga yo‘l qo‘ymaslik uchun ishonch hosil qiling, chunki bu fayllar ishlatiladigan parollardan iborat bo‘lishi mumkin.

Eslatma! Piter Finneganning Oracle Security veb-sayti (http://www.petefinnigan.com) ba'zi bir qiziq va foydali Oracle xavfsizlik maqolalari va skriptlarni taqdim etadi, jumladan SQL in'yeksiyani aniqlash va koʻp boshqa Oracle xavfsizlik masalalari. Finnegan veb-saytida joylashgan keng qamrovli Oracle ma'lumotlar bazasi roʻyxati (Oracle Database ma'lumotlar bazasi) Oracle oʻrnatilishini tekshirish uchun moʻljallangan va Oracle ma'lumotlar bazasi xavfsizligining barcha jihatlarini aks ettiradi.

Tarmoq va tinglovchilar hizmati. Tarmoq va tinglovchi hizmati (TNS Listener) - Oracle xavfsizlik kamchiliklari. Ma'lumotlar bazasiga hujum qilish uchun notoʻgʻri yoʻllarni ochiq qoldirish uchun koʻp imkoniyatlar mavjud. Birinchidan, tinglovchilarning hizmatini mustahkamlash yoʻllarini koʻrib chiqish kerak.

Tinglovchi himoyasi - tinglovchilar ruxsatsiz foydalanuvchilarning Oracle ma'lumotlar bazasiga ulanishiga yoʻl qoʻymaslik uchun doimo paroldan foydalanishi kerak. Esingizda boʻlsa, himoyasiz TNS Listener hizmatiga hujum qilish juda oson. Tinglovchi uchun parol oʻrnatilgach, tinglovchini toʻxtatish yoki ishga tushirish kabi imtiyozli harakatlar tegishli parolni kiritmasdan amalga oshirilmaydi.Siz shuningdek, tinglovchilarning funktsiyalariga aralashish uchun foydalanuvchi SET buyrugʻidan foydalanishga yoʻl qoʻymasligingiz mumkin. Buning uchun listener.ora konfiguratsiya fayliga quyidagi qatorni qoʻshing:

ADMIN_RESTRICTIONS = ON

Odatiy boʻlib, ushbu parametr FALSE ga sozlangan. Tarmoq orqali uzatilganda uning parollari shifrlanmaganligi sababli, tinglovchilarning masofadan boshqarish hizmatidan qochish kerak. Tinglovchi paroli listener.ora faylida saqlanadi, shuning uchun siz ushbu faylni himoya qilishingiz kerak.

Tarmoq xavfsizligi - Zamonaviy internetga asoslangan ma'lumotlar bazasi dasturlarining asosiy xavfsizlik talablaridan biri tizimni xavfsizlik devorlari bilan tashqi dunyodan himoya qilishdir. Xavfsizlik devori oʻrnatilganda, biron-bir sabab

tufayli bo‘shliqlarni bartaraf qilish orqali ishonchliligini saqlab qoling (masalan, tinglovchilar tomonidan internetga ulanish uchun foydalaniladigan portlardan foydalaning).

Odatiy xavfsizlik devoriga qo‘shimcha ravishda, Oracle Net hizmatidan serverga kirishni boshqarish deb ataladigan qo‘shimcha himoya darajasini yaratish uchun foydalanishingiz mumkin. Serverga kirishni boshqarish vositalari manzilni tinglovchilar hizmati orqali ma'lumotlar bazasiga ulanish imkoniyatini cheklaydi. Ulanishlarni amalga oshirish mumkin bo‘lgan manzillarni cheklashning ikki yo‘li mavjud. Siz taklif qilingan (qabul qilingan) manzillarni sqlnet.ora faylida, shuningdek chiqarib yuborilgan manzillarni ro‘yxatlashingiz mumkin. Takliflar ro‘yxatida ro`yxatdan o`tgan barcha tarmoq manzili ulanishga ruxsat beriladi va chiqarilgan tugunlar ro‘yxatidagi barcha manzillar rad etilishi mumkin.

Ishga tushganda, tinglovchilar hizmati sqlnet.ora faylini o‘qiydi va belgilangan kirish nazorati bo‘yicha ruxsat beradi. Server tomonidagi kirishni boshqarishni faollashtirish uchun taklif qilingan manzillarni belgilayotganda quyidagi satrlarni sqlnet.ora fayliga qo‘shing:

```
tcp.validnode_checking = yes
tcp.invited_nodes = (server1.us.wowcompany.com, 172.14.16.152)
```

Manzillarni chiqarib tashlash uchun quyidagi qatorni qo‘shing:

```
tcp.excluded_nodes = (server1.us.wowcompany.com, 172.14.16.152)
```

Eslatma! Odatda, ma'lumotlar bazasiga ulanadigan manzillar ma'lum bo‘lishi sababli, TCP_INVITED_NODES parametrlaridan foydalanib, tizimga kirishni cheklashning eng samarali usuli hisoblanadi.

Masofadan mijoz tomonidan autentifikatsiya qilishni rad etish - uzoqdan turib mijozlarga hisobga olish ma'lumotlari autentifikatsiyasini amalga oshirish qobiliyatiga ega emas va ma'lumotlar bazasiga ulangan mijozlarning

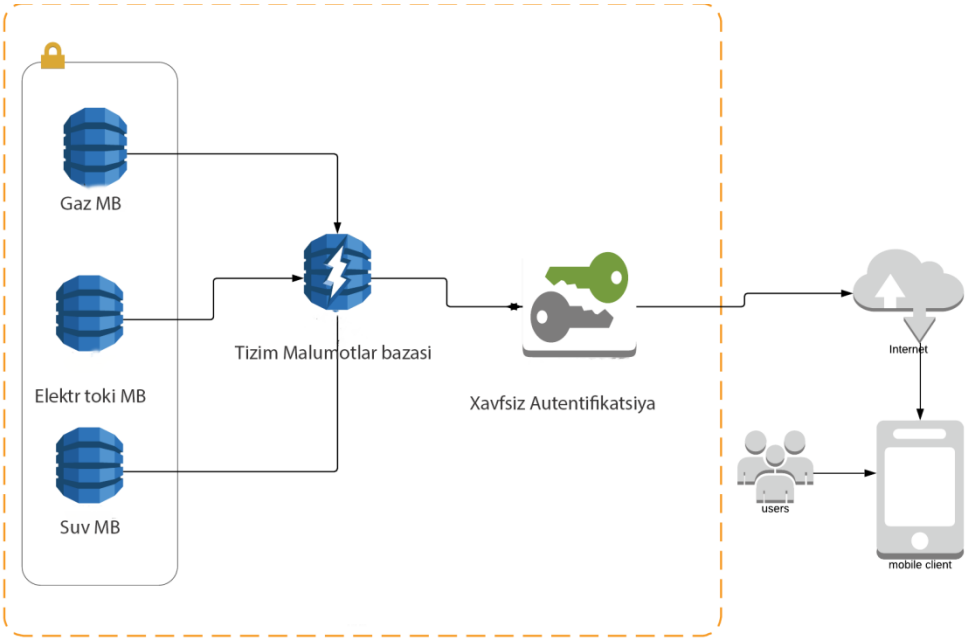
autentifikatsiyasi har doim serverga ishonib topshirilishi kerak. Mijozning operatsion tizimini autentifikatsiya qilish init.ora faylida quyidagi parametrni o‘rnatish orqali o‘chirib qo‘yilishi mumkin:

```
REMOTE_OS_AUTHENT=FALSE
```

Ushbu sozlama mijozlar serverining autentifikatsiyasini rag‘batlantiradi, bu esa mijozning operatsion tizimiga ushbu vazifaga ishonishdan ko‘ra xavfsizroq bo‘ladi.

3.2. Kommunal to‘lov tizimi dasturiy taminotini yaratish bosqichlari

o‘o‘o‘o‘o‘o‘o‘o‘o‘g‘‘(



3.2

o‘lov tizimi dasturiy taminotini server qismini yaratish

1

-

r

a

s

3.2.2 - rasm. IntelliJ IDEA bosh oynasi

a

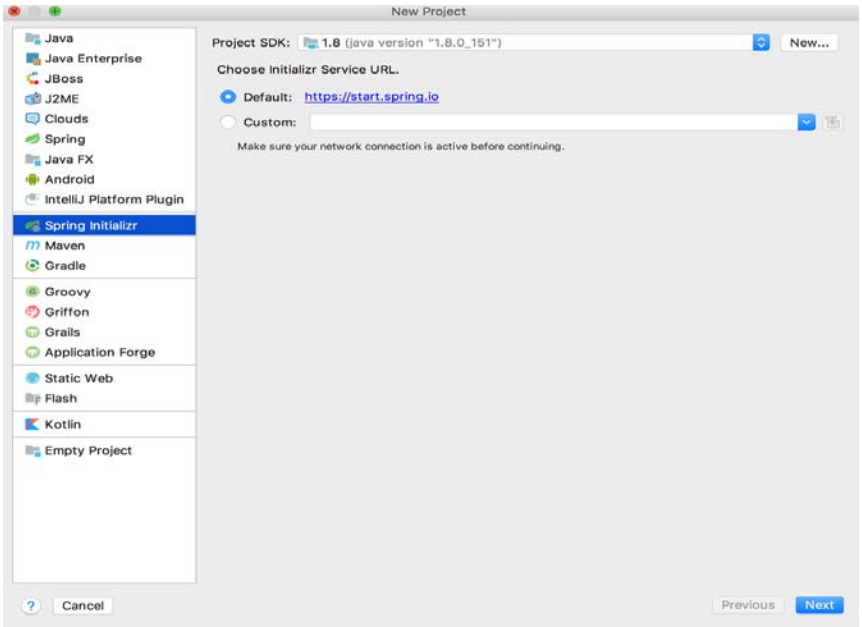
y

o

‘

g

‘



3

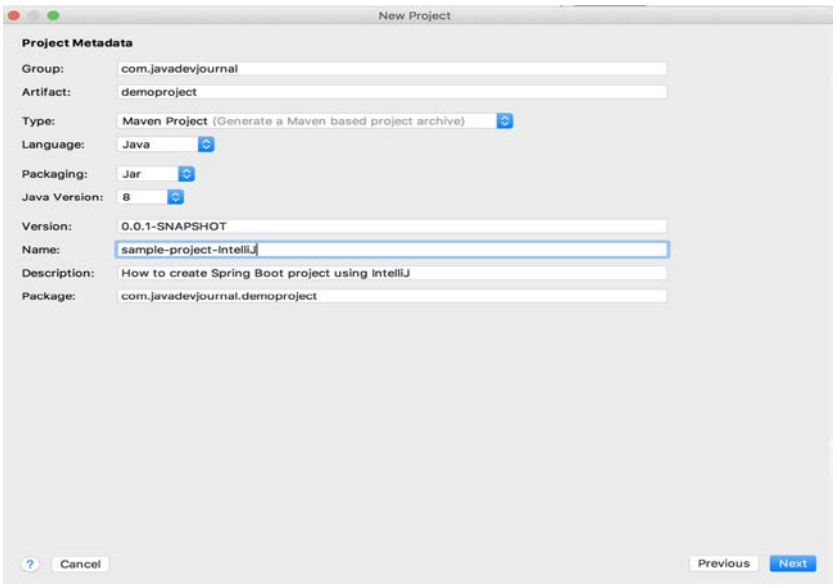
o‘tamiz.

.

2

.

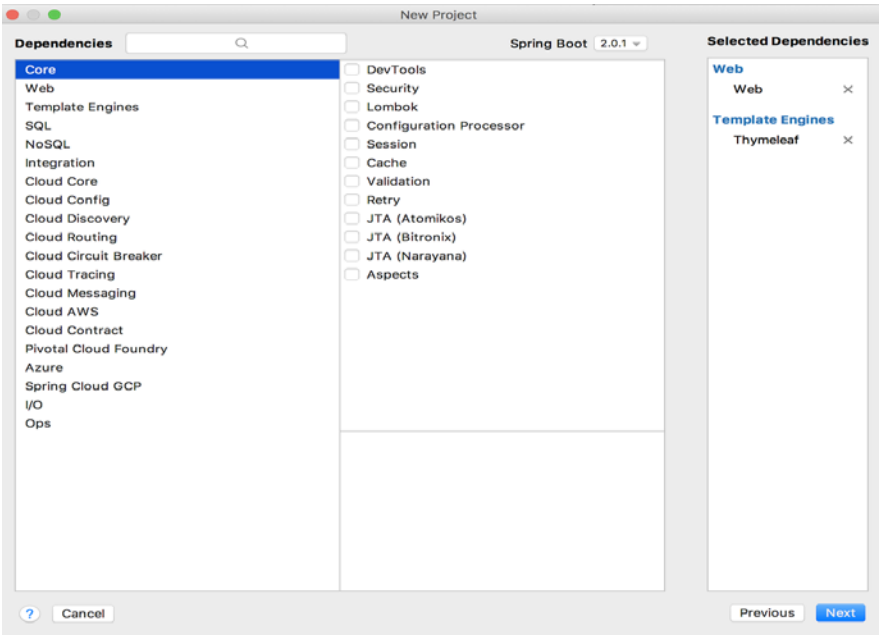
3



3

o .

g‘liqliklar asosida, pom.xml fayliga texnologiyalar kutubxonalarini qo‘shib

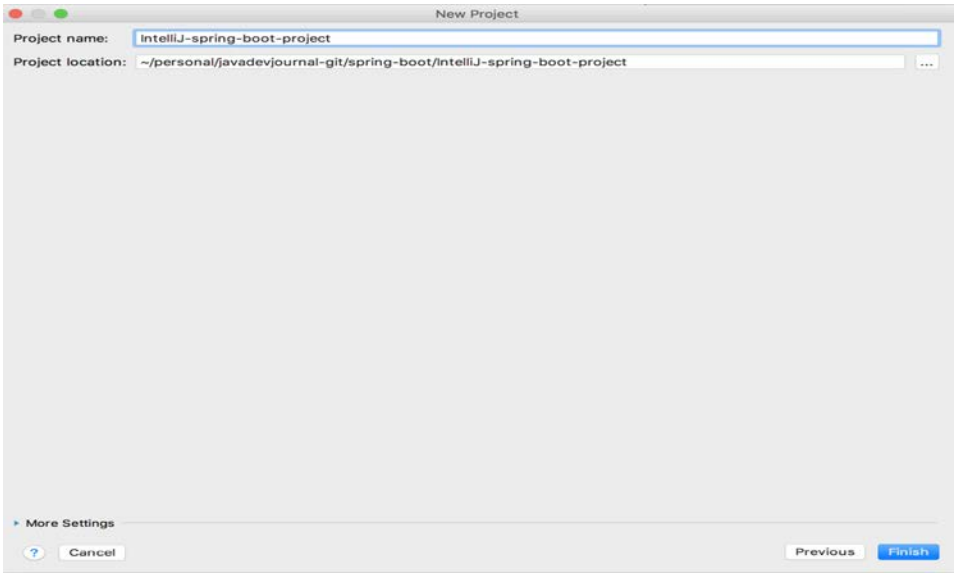


3.2.5 - rasm. Loyihaga texnalogiyalar qo‘shish oynasi

o‘nggi qismda biz loyiha nomi va loyiha o‘rnini tanlaymiz. Tanlanganidan so‘ng,

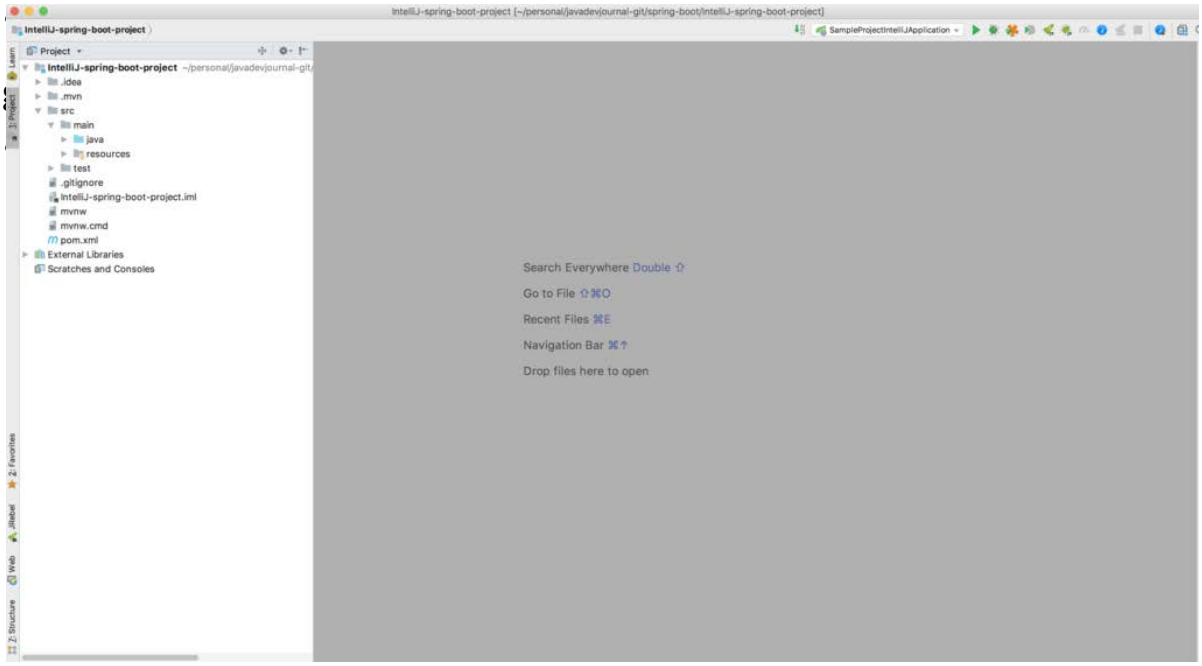
“

”



3.2.6 - rasm. Loyiha nomi va saqlash manzilini ko‘rsatish oynasi

o



3

g .

ɓ 2

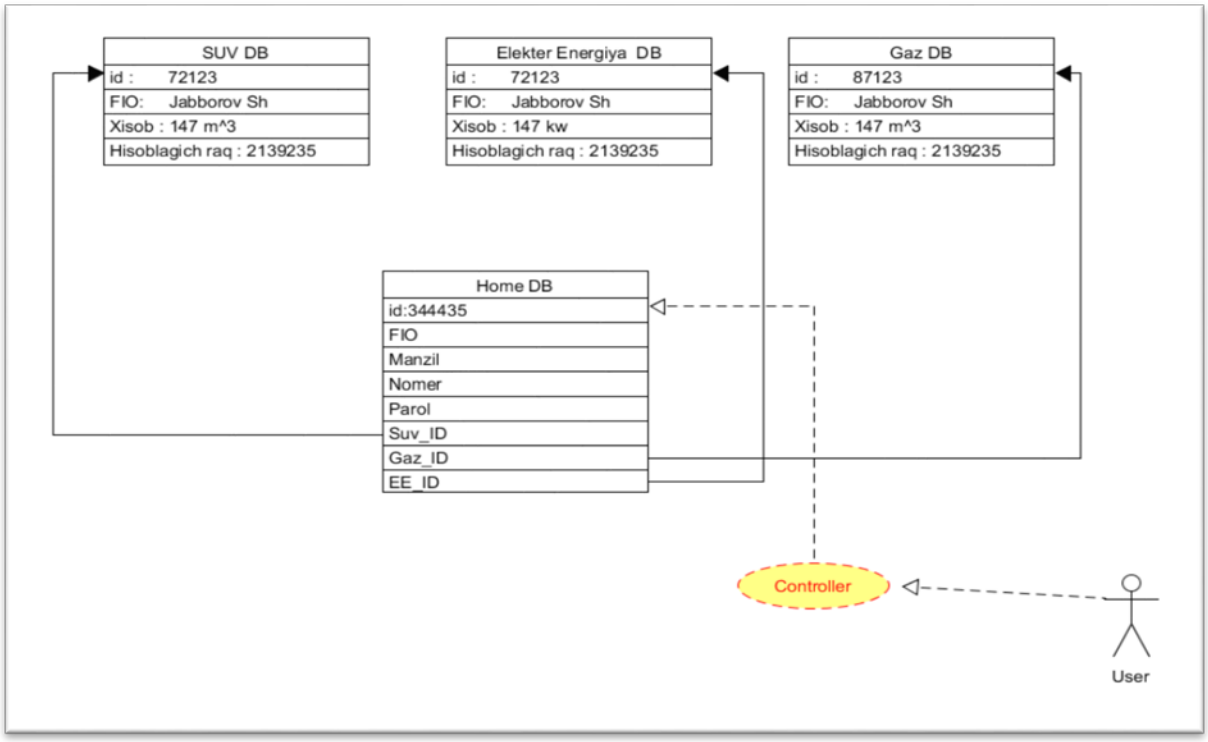
ɓ‘lib, to‘liq boshqarish imkonini beradi. .

ɓ 7

ɓ

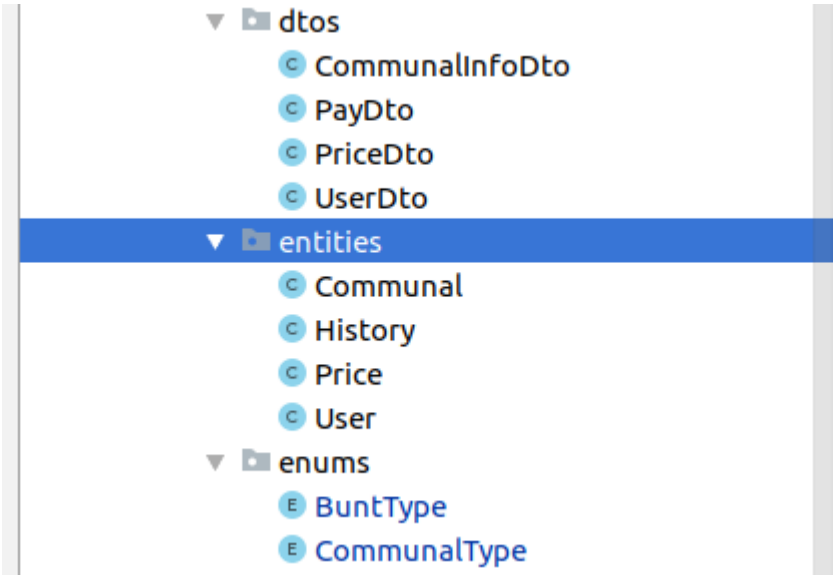
ɓ

g


SUV DB
id : 72123
FIO: Jabborov Sh
Xisob : 147 m^3
Hisoblagich raq : 2139235
Elekter Energiya DB
id : 72123
FIO: Jabborov Sh
Xisob : 147 kw
Hisoblagich raq : 2139235
Gaz DB
id : 87123
FIO: Jabborov Sh
Xisob : 147 m^3
Hisoblagich raq : 2139235
Home DB
id:344435
FIO
Manzil
Nomer
Parol
Suv_ID
Gaz_ID
EE_ID
Controller
User

3.2

3.2.8 da


dtos
CommunalInfoDto
PayDto
PriceDto
UserDto
entities
Communal
History
Price
User
enums
BuntType
CommunalType

3.2.9

```
17
18  server:
19    port: 8888
20  hashids:
21    salt: "eYKM+jt6N!^LZ#au*?Eav$@YJdhSx=kt"
22    minLength: 5
23    alphabet: abcdef1234567890
```

3

o .

‘

```
1   package communal.pay.configs
2
3
4   import ...
8
9   @Configuration
10  open class GlobalConfigs {
11
12    @Bean
13    open fun hashids(
14        @Value( value: "\${hashids.salt}") salt: String,
15        @Value( value: "\${hashids.minLength}") minLength: Int,
16        @Value( value: "\${hashids.alphabet}") alphabet: String
17    ): Hashids = Hashids(salt, minLength, alphabet)
18
19  }
20
```

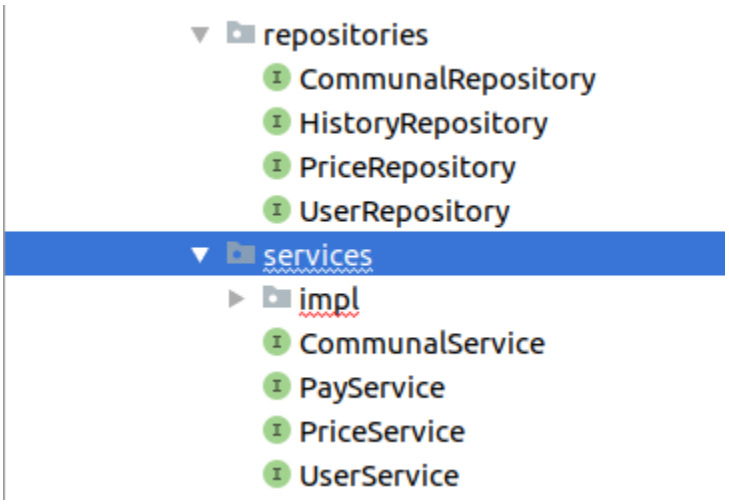
o

‘

3

g .

‘



o

‘

3

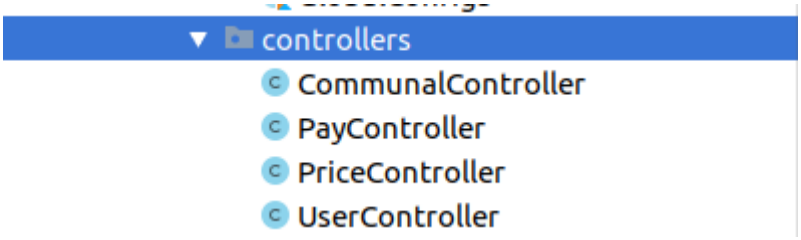
. P

2 A

. G

E

1



3

o‘lov tizimi dasturiy taminotini client qismini yaratish bosqichlari

Kommunal to‘lov tizimi dasturiy taminotini client qismini yani CSS (Communal System Securty) dasturini Android platformalari uchun ishlab chiqiladi.

1. ***“Welcome to Android Studio”*** oynasidan, **“Start a new Android Studio project”** tugmasini bosiladi.

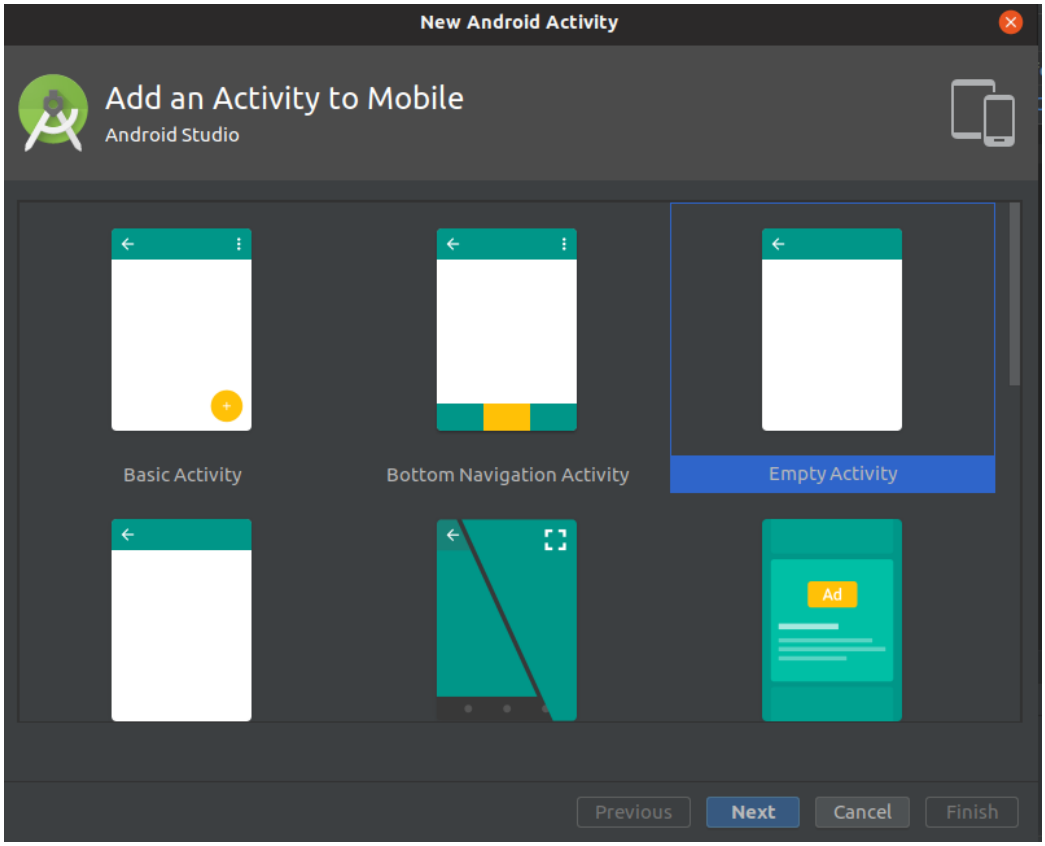
3.2.14 - rasm. Android studio dasturini bosh oynasi

2. “Choose your project” oynasidan (2.28 - rasm) “Empty Activity” tanlanadi.
3. “Next” tugmasi bosiladi.
4. Loyihani sozlash oynasiga quyidagi qiymatlarni kiritiladi:

- Name: “Communal System Securty”;
- Package name: “com.example.communal”;
- Loyiha manzilini o‘zgartirishni hohlasangiz;

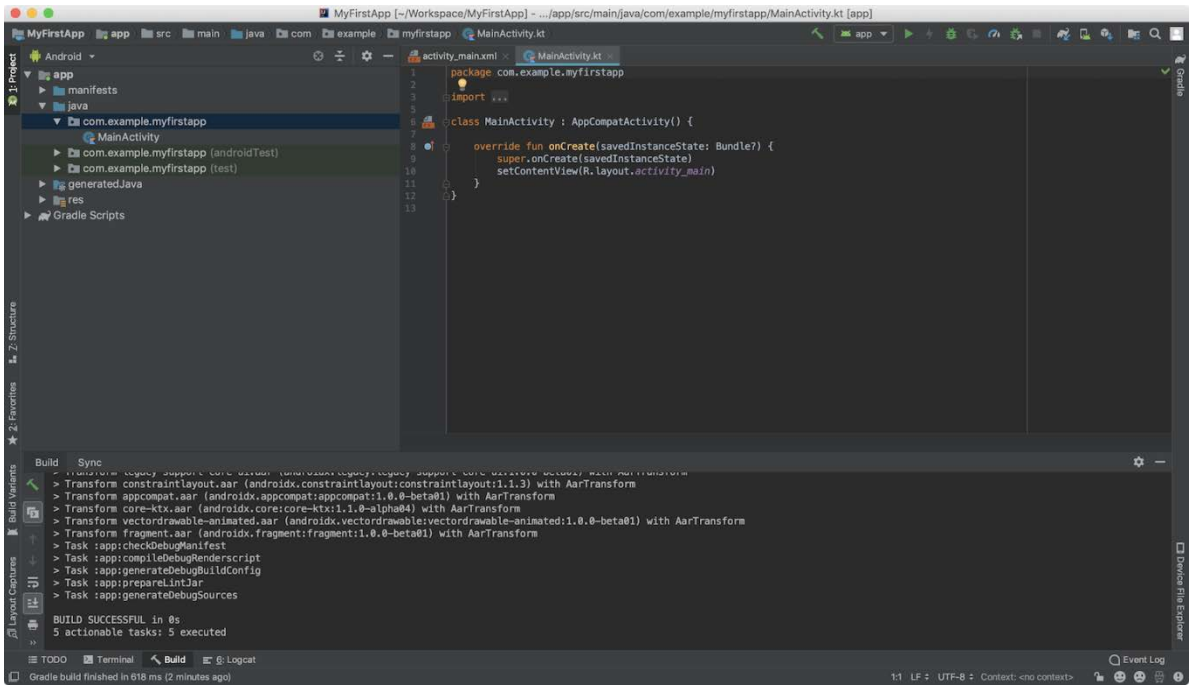
Dasturni Javada yozganimiz uchun, Javani dasturlash tili tanlanadi.

5. “Finish” tugmasi bosiladi.



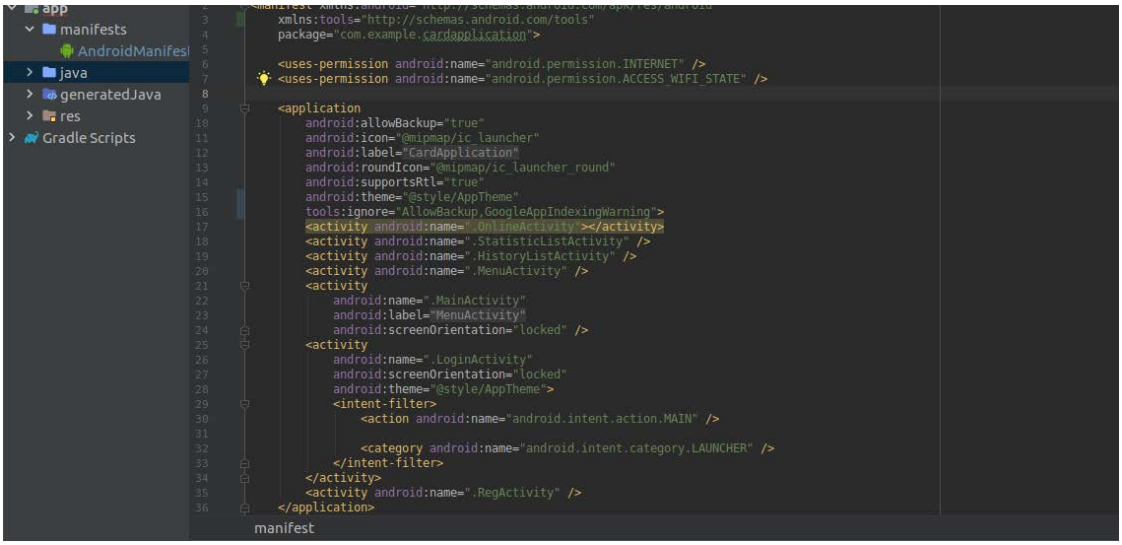
3.2.15 – rasm. Activity qo‘shish oynasi

Bazi sozlamalarni yuklab olgandan so‘ng Android Studio IDEni ochadi.



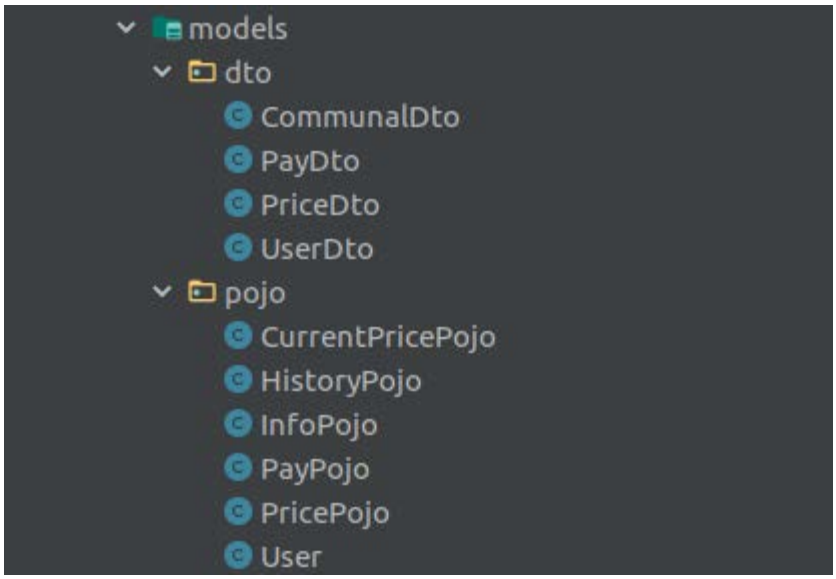
3.2.16 – rasm. Android Studio IDEni barcha sozlamalari yuklanish jarayoni

CSS (Communal System Securty) dasturini manifest sozlamalarini quyidagi oynada ko‘rishingiz mumkun bo‘ladi. Ushbu sozlamalar dasturni eng asosiy qismlaridan biri hisoblanadi.



3.2.17 - rasm. Sozlamalar oynasi

CSS dasturida foydalanilgan barcha modellarni 2.31 – rasmda ko‘rishingiz mumkun bo‘ladi bunda dto va pojolarga bo‘lingan bo‘lib network va user modellar hisoblanadi.

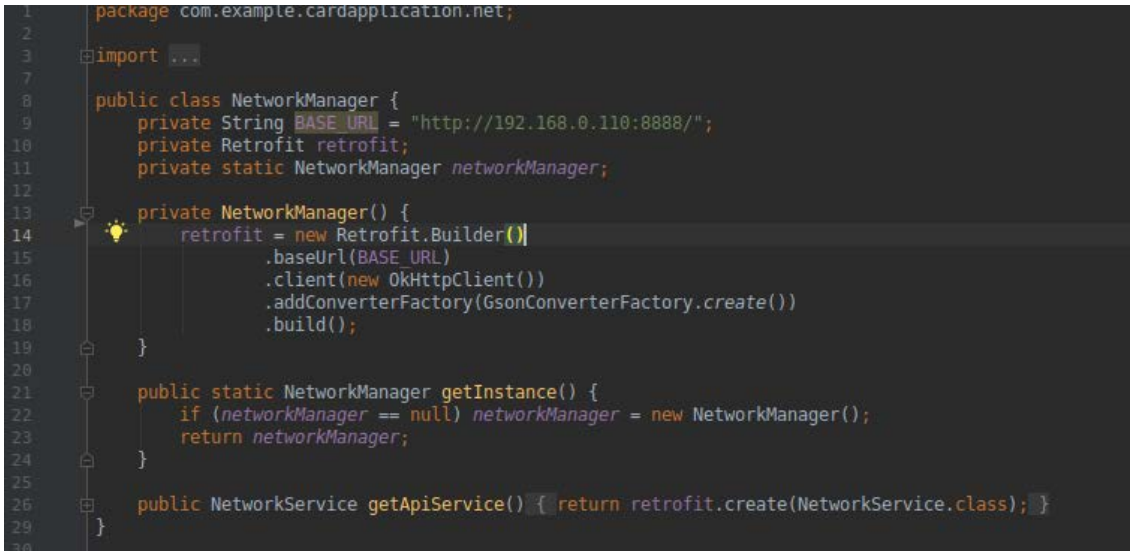


3.2.18 - rasm. Dasturdagi modellar

Android platformalari uchun qilingan dasturlar Activitylardan tashkil topgan bo‘ladi. Activitylar ko‘rinuvchi oynalar hisoblanadi.

3.2.19 – rasm. Dasturdagi barcha Activitylar

CSS dasturini server qisim bilan bog‘lab beruvchi NetworkManager 2.33 – rasmdagi kabi yoziladi unda serverni manzili beriladi va bu manzilda kelga responselar GsonConverter orqali pojolarga o‘giriladi.



3.2.20 - rasm. CSS dasturidagi NetworkManager

CSS dasturida barcha kerakli adapter, model, interface va servicelar server qonuniyatlariga asoslangan holda java dasturlash tilida yozib chiqiladi. Dasturni yaratishni oxirgi bosqichida Keycloak servisi bilan foydalanuvchilarni autentifikatsiyadan o‘tkazish tizimi qoshiladi va server bilan test qilinadi.

3.3. Kommunal to‘lov tizimi dasturiy taminotidan foydalanish yo‘riqnomasi

o

ó

Ć

ś

S

ǵ

Ć

o

ḿ

m

ú

n

á

l

3

.

o

‘

o

‘

o

‘

o

‘

o

‘

o

‘

o

‘

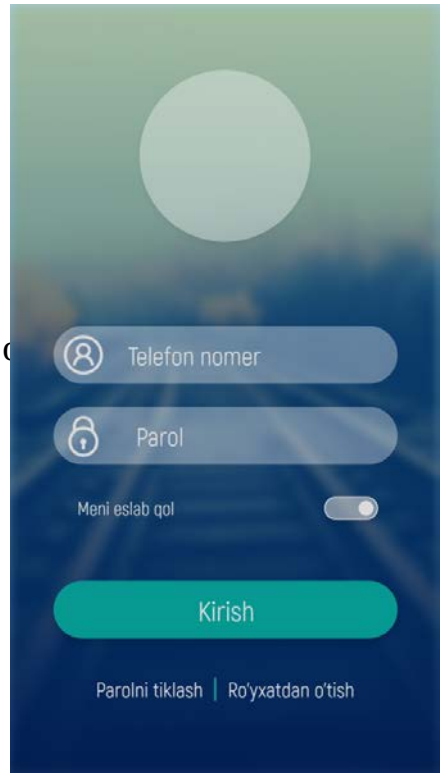
o

‘

o

‘

o

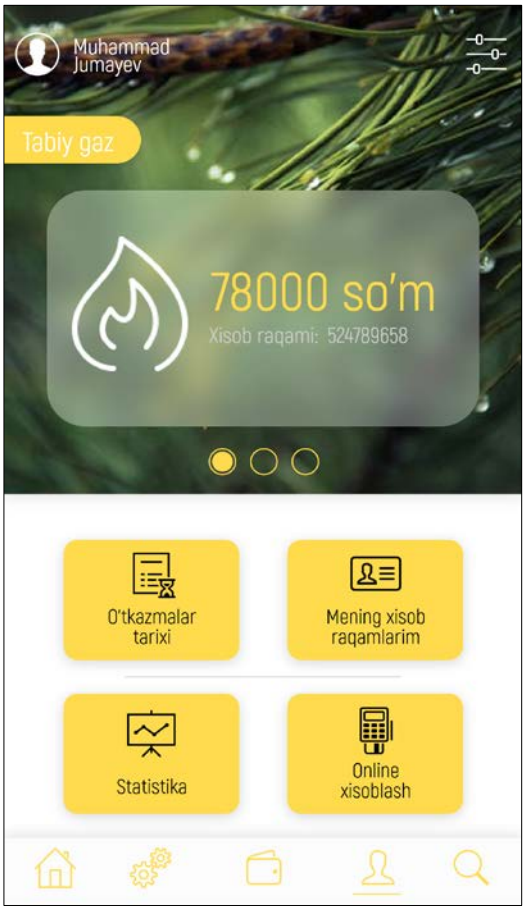


3

.

3

.



3

o .

‘ 3

g .

‘ 3

o

‘

o ma’lumotlar oynasi

‘

“

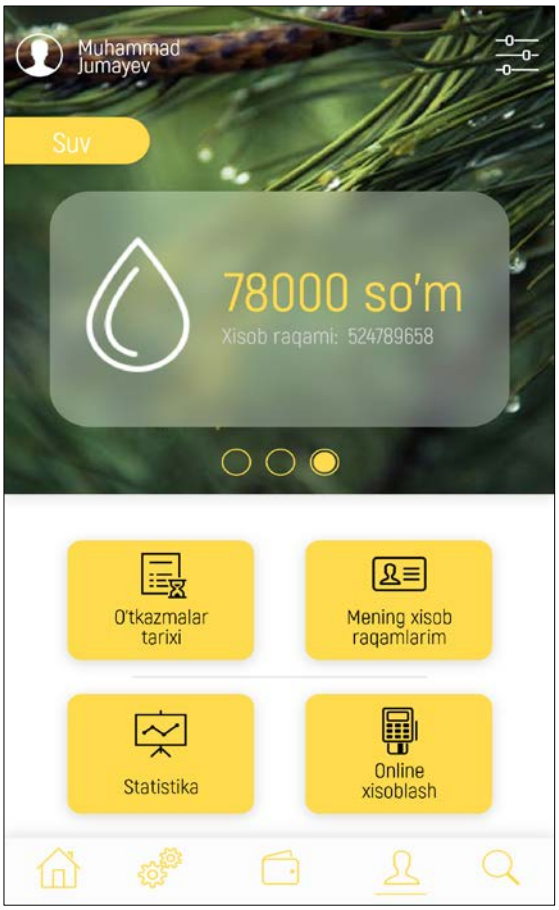
O

‘

”

o

‘



3

3 .

. 3

3 .

. 4

4

o

‘ ma’lumotlar oynasi

o

‘

g

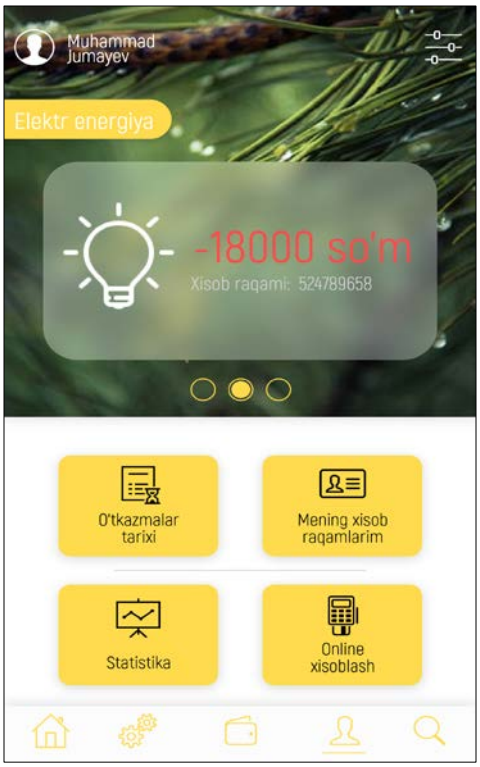
‘

o

‘

o

‘



3

.

o 3

‘ .

o 5

‘

o

‘

o

‘

o

‘

o

‘

o

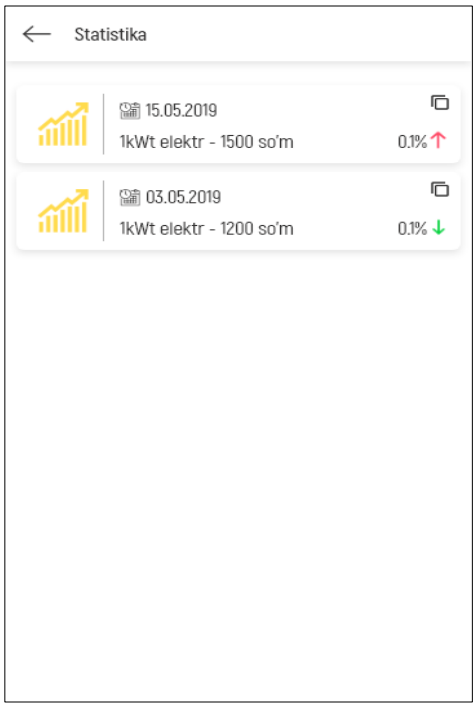
‘

o

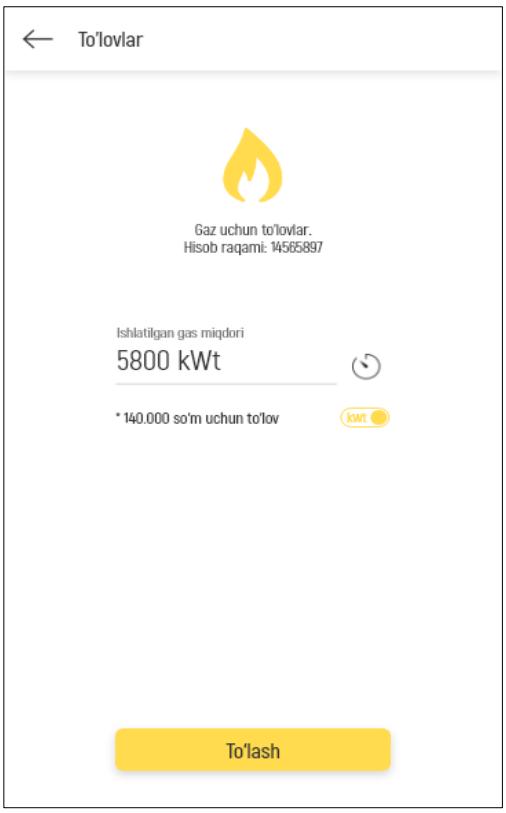
‘

o

‘


Statistika
15.05.2019
1kWt elektr - 1500 so'm
0.1%
03.05.2019
1kWt elektr - 1200 so'm
0.1%

3


To'lovlar
Gaz uchun to'lovlar.
Hisob raqami: 14565897
Ishlatilgan gas miqdori
5800 kWt
* 140.000 so'm uchun to'lov
kWt
To'lash

3

.

3

.

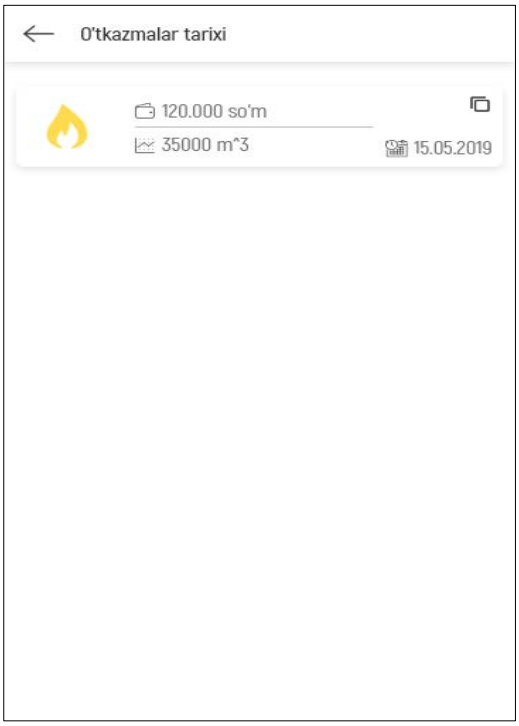
7

o

‘

s

PAGE

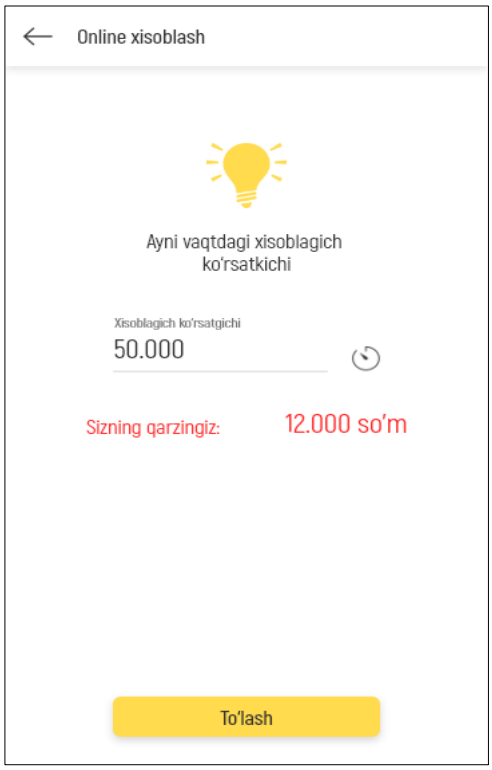


3

O‘tkazmalar tarixi oynasi orqali foydalanuvchini barcha to‘lovlarini o‘rishi mumkun bo‘ladi(2.42 – rasm). O‘tkazma summasini, sanasini va hajmini o‘rish mumkun boladi. Bu orqali foydalanuvchi o‘z tolovlarini tarixini va qancha o‘langanini ko‘rish imkoniga ega bo‘ladi.

o‘ o‘ o‘ “ o‘ ” o‘

O‘tkazmalar tarixi oynasi



3

III bob bo‘yicha xulosalar

3

Dissertatsiya ishining uchinchi bobining 1-bo‘limida Internet makonida Kommunal to‘lov tizimlari ma’lumotlarini bazada himoyalash modellari va usullari keltirildi. Tahdidlarni hal qilish usullari keltirildi. Oracle Ma’lumotlar bazasining o‘z avtomatik xavfsizlik konfiguratsiyalarini sozlash algoritmlari keltirildi.

2-bo‘limda Kommunal to‘lov tizimi dasturiy taminotini yaratish bosqichlari keltirib o‘tildi. Kommunal to‘lov tizimi dasturiy taminotini server qismini yaratish boshqichlari va Kommunal to‘lov tizimi dasturiy taminotini client qismini yaratish bosqichlari yaratish texnologiyalari misolida keltirildi.

3-bo‘limda Kommunal to‘lov tizimi dasturiy taminotidan foydalanish yo‘riqnomasi keltirildi.

XULOSA

Magistrlik dissertatsiyasi ishining birinchi bobida Xavfsiz veb-saytlarning tuzilishi arxitekturalari, xavfsizlikni ta’minlashdagi muammolar va ularni hal qilish usullari tahlil qilindi. Tahlil natijasida Malumotlarni shifrlash (Encrypt data) - ma’lumotlar xavfsizligi veb-sayt xavfsizligining eng muhim jihati hisoblanishi aniqlandi. HTTPS (Hypertext Transfer Protocol Secure) yordamida veb-saytlarni shifrlash onlayn aloqani ta’minlash va maxfiy ma’lumotlarni himoya qilishda muhim qadamdir. Internet makoni Veb-saytlarida kiberxavfsizlikni ta’minlash muammolari ham tahlil qilindi. Tahlil natijasida saytlararo so‘rovlarni qalbakilashtirish, Zararli dastur veb-saytga zarar etkazishi va tashrif buyuruvchilar qurilmalariga tarqalishi, Veb-saytlarda kiberxavfsizlikni ta’minlashning bir usuli bu veb-ilovalar xavfsizlik devorini (WAF) amalga oshirish ekanligi aniqlandi. Uchinchi tomon provayderlarini tekshirish va ularning tegishli xavfsizlik choralari mavjudligini ta’minlash muhimligi ta’kidlandi. Internet makonidagi Veb-saytlarning zaifliklarini aniqlash usullari tahlili amalga oshirildi. Zaiflikni uzluksiz skanerlash va sinovdan o‘tkazish pentesterchilar va ishlab chiquvchilarga bosimni engillashtirib, xavfsizlik operatsiyalarini soddalashtirishi aniqlandi. Veb-skanerlar tomonidan qo‘llaniladigan usullarga ilovalarni skanerlash, standart tarkibni tekshirish, shuningdek, umumiy tarkibni topish va umumiy zaifliklar uchun veb-ilovalarni tekshirish kiradi. Web sayt xavfsizligini tekshirishning tasdiqlangan usullari ko‘rib chiqildi.

Dissertatsiya ishining ikkinchi bobida Internet makonida Veb saytlarga uyushtiriladigan hujum turlari tahlili amalga oshirildi. ko‘plab davlat idoralari, korparatsiyalar va jismoniy shaxslarning ma’lumotlariga nisbatan amalga oshirilgan hujumlar tarixi tahlil qilindi. Saytlararo skript yaratish, SQL Injection, Saytlararo so‘rovlarni qalbakilashtirish, Taqsimlangan xizmat ko‘rsatishni rad etish, Shafqatsiz kuch hujumlari, Fayllarni qo‘shish ekspluatatsiyasi, Server tomonidagi so‘rovlarni

soxtalashtirish, Clickjacking, Remote Code Execution, Fishing hujumlari kabi usullar haqidagi ma’lumotlar tahlil qilindi. Veb saytlarni yaratishda xavfsiz kodlash usullari va algoritmlari o‘rganildi. Maxfiy ma’lumotlarni saqlash usuli, DDOS hujumlarni oldini oluvchi dastur kodi keltirilgan. Sun’iy DDOS hujumini amalga oshiruvchi dastur kodi yordamida Sun’iy DDOS hujumini amalga oshish jarayoni amalga oshirildi va dastur tomonidan foydalanuvchi bloklangan holat keltirib o‘tildi. DDOS hujumlardan NGINX orqali himoyalanish usuli keltirildi. OWASP standartlaridan foydalanib veb saytlarni kiberhujumlardan himoya qilish choralarini samaradorligini oshirish algoritmlari keltirildi. Kirish nazorati zaifliklari, Kriptografik xatolar, Kriptografik nosozlikning oldini olish choralari, ilovalarning qaysi hollarda in’eksiyalarga zaifligi keltirildi. Saytlarda uchraydigan xatolarni qanday qilib OWASP da keltirib o‘tilgan eng xavfsizlikka tahdid standartlari yordamida bartaraf etish usullari tahlil qilindi.

Dissertatsiya ishining uchinchi bobida Internet makonida Kommunal to‘lov tizimlari ma’lumotlarini bazada himoyalash modellari va usullari keltirildi. Tahdidlarni hal qilish usullari keltirildi. Oracle Ma’lumotlar bazasining o‘z avtomatik xavfsizlik konfiguratsiyalarini sozlash algoritmlari keltirildi. Kommunal to‘lov tizimi dasturiy taminotini yaratish bosqichlari keltirib o‘tildi. Kommunal to‘lov tizimi dasturiy taminotini server qismini yaratish boshqichlari va Kommunal to‘lov tizimi dasturiy taminotini client qismini yaratish bosqichlari yaratish texnologiyalari misolida keltirildi. Shuningdek, Kommunal to‘lov tizimi dasturiy taminotidan foydalanish yo‘riqnomasi keltirildi.

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 Farmonidan.
2. O‘zbekiston Respublikasi Prezidentining 2017 yil 7 fevraldagi “O‘zbekistonni Respublikasini yanada rivojlantirish bo‘yicha Harakatlar strategiyasi to‘grisida” gi Farmoni.
3. S. K. G‘aniev, M. M. Karimov. K. A. Tashev “Axborot Xavfsizligi”. “Fan va texnologiyalar” nashryoti. Toshkent 2016.
4. S. K. G‘aniev, A.A G‘aniyev, Z.T. Xudoyqulov “Kiberxavfsizlik asoslari”. O‘quv qo‘llanma. Toshkent-“Aloqachi” - 2020.
5. Bryan Sullivan, Vincent Liu, Michael Cross “Web Application Security: A Beginner's Guide” 2011 year.
6. Бабин С.А. Инструментарий хакера – СПб.: БХВ-Петербург, 2014. – 240 с
7. Malcolm McDonald and James Richardson “Web Security for Developers: Real Threats, Practical Defense” 2015 year.
8. Ivan Ristic «Bulletproof SSL and TLS: Understanding and Deploying SSL/TLS and PKI to Secure Servers and Web Applications» 2014 year.

“Hayot faoliyati xavfsizligi va ekologiya”. O‘quv qo‘llanma, Toshkent-“Aloqachi”-2019, 276 b.

“Hayot faoliyati xavfsizligi”. Toshkent-2012

Phil Hughes, Ed Ferrett “Introduction to Health and Safety at Work”. The Boulevard, Langford Lane, Kidlington, Oxford OX5 1GB, UK. ISBN: 978-0-08-097070-7. 2011/ p-636/

“

13. Ehttps://fvv.uz/ . Favqulotda vaziyatlar vazirligi veb-sayti.
14. uhttps://lex.uz/ . O‘zbekiston qonun hujjatlari ma’lumotlari milliy bazasi.

n

d

a