OLIY TA’LIM, FAN VA INNOVATSIYALAR VAZIRLIGI

RAQAMLI TEXNOLOGIYALAR VAZIRLIGI

MUHAMMAD AL-XORAZMIY NOMIDAGI TOSHKENT AXBOROT TEXNOLOGIYALARI UNIVERSITETI HUZURIDAGI PEDAGOG KADRLARNI QAYTA TAYYORLASH VA ULARNING MALAKASINI OSHIRISH TARMOQ MARKAZI

“KRIPTOGRAFIK PROTOKOLLAR”

MODULI BO‘YICHA

O‘ Q U V –U S L U B I Y M A J M U A

Toshkent – 2025

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

OLIY TA’LIM TIZIMI PEDAGOG VA RAHBAR KADRLARINI QAYTA TAYYORLASH VA ULARNING MALAKASINI OSHIRISHNI TASHKIL ETISH BOSH ILMIY - METODIK MARKAZI

MUHAMMAD AL-XORAZMIY NOMIDAGI TOSHKENT AXBOROT TEXNOLOGIYALARI UNIVERSITETI HUZURIDAGI PEDAGOG KADRLARNI QAYTA TAYYORLASH VA ULARNING MALAKASINI OSHIRISH TARMOQ MARKAZI

“Kriptologiya” yo‘nalishi

“KRIPTOGRAFIK PROTOKOLLAR”

MODULI BO‘YICHA

O‘ Q U V –U S L U B I Y M A J M U A

Toshkent – 2025

Modulning o‘quv-uslubiy majmuasi Oliy ta’lim, fan va innovatsiyalar vazirligining 2024 yil 27 dekabrdagi №485-sonli buyrug‘i bilan tasdiqlangan o‘quv dasturi va o‘quv rejasiga muvofiq ishlab chiqilgan.

Tuzuvchilar: **O.Allanov-** texnika fanlari bo‘yicha falsafa doktori, dotsent.

.....

Taqrizchilar: **B.F. Abdurahimov** – fizika-matematika fanlari doktori, professor.

O.P. Axmedova - texnika fanlari nomzodi, dotsent.

O‘quv-uslubiy majmua O‘quv dasturi Muhammad al-Xorazmiy nomidagi Toshkent axborot texnologiyalari universiteti Kengashining qarori bilan tasdiqqa tavsiya qilingan (2024-yil 27-noyabrdagi 3/4 (745/746)- sonli bayonnoma).

MUNDARIJA

I-BO‘LIM
ISHCHI DASTUR

I. ISHCHI DASTUR

KIRISH

Dastur O‘zbekiston Respublikasining 2020 yil 23 sentabrda tasdiqlangan “Ta’lim to‘g‘risida”gi Qonuni, O‘zbekiston Respublikasi Prezidentining 2017 yil 7 fevraldagi “O‘zbekiston Respublikasini yanada rivojlantirish bo‘yicha Harakatlar strategiyasi to‘g‘risida”gi PF-4947-son, 2019 yil 27 avgustdagi “Oliy ta’lim muassasalari rahbar va pedagog kadrlarining uzluksiz malakasini oshirish tizimini joriy etish to‘g‘risida”gi PF-5789-son, 2019 yil 8 oktabrdagi “O‘zbekiston Respublikasi oliy ta’lim tizimini 2030 yilgacha rivojlantirish konsepsiyasini tasdiqlash to‘g‘risida”gi PF-5847-son va 2020 yil 29 oktabrdagi “Ilm-fanni 2030 yilgacha rivojlantirish konsepsiyasini tasdiqlash to‘g‘risida”gi PF-6097-sonli Farmonlari hamda O‘zbekiston Respublikasi Vazirlar Mahkamasining 2019 yil 23 sentabrdagi “Oliy ta’lim muassasalari rahbar va pedagog kadrlarining malakasini oshirish tizimini yanada takomillashtirish bo‘yicha qo‘shimcha chora-tadbirlar to‘g‘risida”gi 797 sonli Qarorlarida belgilangan ustuvor vazifalar mazmunidan kelib chiqqan holda tuzilgan bo‘lib, u oliy ta’lim muassasalari pedagog kadrlarining kasb mahorati hamda innovatsion kompetentligini rivojlantirish, sohaga oid ilg‘or xorijiy tajribalar, yangi bilim va malakalarni o‘zlashtirish, shuningdek amaliyotga joriy etish ko‘nikmalarini takomillashtirishni maqsad qiladi.

Qayta tayyorlash va malaka oshirish yo‘nalishining o‘ziga xos xususiyatlari hamda dolzarb masalalaridan kelib chiqqan holda dasturda tinglovchilarning ushbu fan doirasidagi bilim, ko‘nikma, malaka hamda kompetensiyalariga qo‘yiladigan talablar takomillashtirilishi mumkin.

Modulning maqsadi va vazifalari

Modulining maqsadi: oliy ta‘lim muassasalari pedagog kadrlarining avtomatlashtirilgan axborot tizimlarida axborot himoyasini xavfsizlik protokollari yordamida ta’minlash bilan bog‘liq nazariy va amaliy bilimlar, ko‘nikmalar va malakalarini oshirishdan iborat.

Modulning vazifalari:

Kompyuter tarmoqlarida axborotlarni xavfsiz uzatishda qo‘llaniladigan protokollar haqida nazariy bilimlar va amaliy ko‘nikmalalarini orttirish, kriptografik protokollarni tadqiq qilish va ishlatish bo‘yicha malakalarini oshirishdan iborat.

Modul bo‘yicha tinglovchilarning bilim, ko‘nikma, malaka va kompetensiyalariga qo‘yiladigan talablar

“Kriptografik protokollar” modulini o‘zlashtirish jarayonida amalga oshiriladigan masalalar doirasida:

Tinglovchi:

– protokollar, kompyuter tarmoqlari, kompyuter tizimlari ***haqida tasavvurga ega bo‘lishi;***

– kompyuter tarmoqlarida autentifikatsiyalash va identifikatsiyalash protokollarini;

– parol tizimlari orqali ruxsat etilganlikni chegaralashni;

– protokollarni buzishga urinishlarni;

– bir tomonlama funksiyalarini;

– xabarlarni haqiqiyligini tekshirish kodlarini;

– aralash kriptotizimlarini;

– kalitlarni generatsiyalash protokollarini ***bilishi va ulardan foydalana olishi;***

– protokollarning turlari va ularning vaziflari ularni ishlatish, autentifikatsiya va identifikatsiya protokollarini qo‘llash, Xavfsiz aloqa protokollarining ishlatish xususiyatlari,

Xavfsiz aloqa protokollarini ishlab chiqish va qo‘llash, ularni samaradorligini oshirish *ko‘nikmalariga ega bo‘lishi kerak*.

Modulni tashkil etish va o‘tkazish bo‘yicha tavsiyalar

“Kriptografik protokollar” moduli ma’ruza va amaliy mashg‘ulotlar shaklida olib boriladi.

Modulni o‘qitish jarayonida ta’limning zamonaviy metodlari, pedagogik texnologiyalar va axborot-kommunikatsiya texnologiyalari qo‘llanilishi nazarda tutilgan:

- ma’ruza darslarida zamonaviy kompyuter texnologiyalari yordamida prezentatsion va elektron-didaktik texnologiyalardan;
- o‘tkaziladigan amaliy mashg‘ulotlarda texnik vositalardan, ekspress-so‘rovlar, test so‘rovlari, aqliy hujum, guruhli fikrlash, kichik guruhlar bilan ishlash, va boshqa interaktiv ta’lim usullarini qo‘llash nazarda tutiladi.

Modulning o‘quv rejadagi boshqa modullar bilan bog‘liqligi va uzviyligi

“Kriptografik protokollar” moduli mazmuni o‘quv rejadagi “Kriptografiyaning matematik asoslari” o‘quv moduli bilan uzviy bog‘langan holda pedagoglarning ta’lim jarayonida kriptografik protokollar, ularning axborot xavfsizligini ta’minlashdagi ro‘li, ularni sozlash bo‘yicha kasbiy pedagogik tayyorgarlik darajasini oshirishga xizmat qiladi.

Modulning oliy ta’limdagi o‘rni

Modulni o‘zlashtirish orqali tinglovchilar ta’lim jarayonida kriptografik protokollar, ularning axborot xavfsizligini ta’minlashdagi ro‘li, ulardan foydalanish va amalda qo‘llashga doir kasbiy kompetentlikka ega bo‘ladilar.

MODUL BO‘YICHA SOATLAR TAQSIMOTI

№	Modul mavzulari	Auditoriya uquv yuklamasi				
		Jami	jumladan			
			Nazariy	Amaiy mashg'ulot	Ko'chma mashg'uloti	
1.	Kriptografik protokollar faniga kirish	2	2			
2.	Kriptografik kalit almashish protokollari	2	2			
3.	Ochiq kalitlar infratuzilmasi	2	2			
4.	Xavfsiz aloqa protokollari	2	2			
5	Xavfsiz pochta protokollari	2	2			
6	Xavfsiz tranzaksiya protokollari	2	2			

7	E’lon qilinganligi nolga teng protokollar	2	2			
8	HLPSL tilida sodda protokollarni ifodalash va SPAN+AVISPA vositasida yuklash	2		2		
9	HLPSL tilida kalitlarni almashinish protokollarini ifodalash va SPAN+AVISPA vositasida yuklash	4		4		
10	HLPSL tilida turli zamonaviy protokollarni ifodalash va SPAN+AVISPA vositasida yuklash.	4		4		
	Jami:	**24**	**14**	**10**		

NAZARIY MASHG‘ULOTLAR MAZMUNI

1-MAVZU: KRIPTOGRAFIK PROTOKOLLAR FANIGA KIRISH (2 soat)

Protokol va uning vazifalari. Protokol ishtirokchilari. Vositachi yordamida protokol. O‘ziga yetarli protokol. Arbitir ishtirokidagi protokol. Protokollarning xossalari. Protokollarni sinflanishi. Protokollarga qo‘yilgan talablar. Sodda xavfsizlik protokollari. Autentifikatsiyalash protokollari. Autentifikatsiya va TCP. Nollik bilimga asoslangan protokollar. Sessiya kalitlarni uzatish. Simmetrik kalitlar yordamida autentifikatsiyalash. Ochiq kalitlar yordamida autentifikatsiyalash.

2-MAVZU: KRIPTOGRAFIK KALIT ALMASHISH PROTOKOLLARI (2 SOAT)

Simmetrik shifrlash algoritmidan foydalanib kalit uzatish protokollari. Asimmetrik kalitli algoritmlar foydalanib kalit uzatish protokollari.

3-MAVZU: OCHIQ KALITLAR INFRATUZILMASI (2 SOAT)

X.509 sertifikati. Sertifikatlarni boshqarish infratuzilmasi. PKCS standartlari. DER kodirovkasi. Ochiq kalitlar infratuzilmasidan foydalanuvchi xizmatlar.

4-MAVZU: XAVFSIZ ALOQA PROTOKOLLARI (2 SOAT)

SSH protokoli. SFTP va FTPS protokollari. SSL protokoli. IPSec protokoli. WEP va WPA protokollari. GSM standarti.

5-MAVZU: XAVFSIZ POCHTA PROTOKOLLARI (2 SOAT)

PGP protokoli. S/MIME protokoli. POP protokoli. POP3 protokoli.

6-MAVZU: XAVFSIZ TRANZAKSIYA PROTOKOLLARI (2 SOAT)

Kerberos protokoli. Kerberos protokoli ishtirokchilari. Kerberos yorlig‘i. Kerberos xavfsizligi. SET (Secure Electronic Transactions) protokoli. SET protokoli tashkil etuvchilari. Ikki tomonlama imzolash.

7-MAVZU:E’LON QILINGANLIGI NOLGA TENG PROTOKOLLAR (2 SOAT)

Nollik bilimga asoslangan tasdiq. Fiat-Shamir protokoli. Afzalliklari va kamchiliklari. Nollik bilimga asoslangan vebda foydalanishli protokol.

AMALIY MASHG‘ULOTLAR MAZMUNI

1-MAVZU: HLPSL TILIDA SODDA PROTOKOLLARNI IFODALASH VA SPAN+AVISPA VOSITASIDA YUKLASH (2 SOAT)

HLPSL tilida, SPAN+AVISPA vositasi, secret(information,identifier,agents-set), witness(X,Y,identifier,information) va request(Y,X,identifier,information) buyruqlari.

2-MAVZU: HLPSL TİLİDA KALİTLARNİ ALMASHİNİSH PROTOKOLLARİNİ İFODALASH VA SPAN+AVISPA VOSİTASİDA YUKLASH (SOAT)

Kalit almashish protokoli, kalit almashish protokolini HLPSL tilida SPAN+AVISPA vositasiga yuklash, kalit almashish protokollarini xavfsizlik talablariga tekshirish.

3-MAVZU: HLPSL TILIDA TURLI ZAMONAVIY PROTOKOLLARNI IFODALASH VA SPAN+AVISPA VOSITASIDA YUKLASH (4 SOAT)

Endryu Secure RPC protokolining vazifasi va ishlash prinsipi, Endryu Secure RPC protokolini HLPSL tilida ifodalash, SPAN+AVISPA vositasida yuklagan protokolni kriptografik hujum usullariga tekshirish va zaifliklarni bartarat etish.

O‘QITISH SHAKLLARI

Mazkur modul bo‘yicha quyidagi o‘qitish shakllaridan foydalaniladi:

- ma’ruzalar, amaliy mashg‘ulotlar (ma’lumotlar va texnologiyalarni anglab olish, motivatsiyani rivojlantirish, nazariy bilimlarni mustahkamlash);
- davra suhbatlari (ko‘rilayotgan loyiha yechimlari bo‘yicha taklif berish qobiliyatini rivojlantirish, eshitish, idrok qilish va mantiqiy xulosalar chiqarish);
- bahs va munozaralar (loyihalar yechimi bo‘yicha dalillar va asosli argumentlarni taqdim qilish, eshitish va muammolar yechimini topish qobiliyatini rivojlantirish).

II-BO‘LIM

MODULNI O‘QITISHDA FOYDALANILADIGAN INTERFAOL TA’LIM METODLARI

II. MODULNI O‘QITISHDA FOYDALANILADIGAN INTERFAOL TA’LIM METODLARI

“Blum kubigi” metodi

Metodning maqsadi: Mazkur metod tinglovchilarda yangi axborotlar tizimini qabul qilish va bilimlarni o‘zlashtirilishini yengillashtirish maqsadida qo‘llaniladi, shuningdek, bu metod tinglovchilar uchun “Ochiq” savollar tuzish va ularga javob topish mashqi vazifasini belgilaydi.

Metodni amalga oshirish tartibi:

1. Ushbu metodni ko‘llash uchun, oddiy kub kerak bo‘ladi. Kubning har bir tomonida ko‘yidagi so‘zlar yoziladi:
   - **Sanab bering, ta’rif bering (oddiy savol)**
   - **Nima uchun (sabab-oqibatni aniqlashtiruvchi savol)**
   - **Tushintirib bering (muammoni har tomonlama qarash savoli)**
   - **Taklif bering (amaliyot bilan bog‘liq savol)**
   - **Misol keltiring (ijodkorlikni rivojlantirovchi savol)**
   - **Fikr bering (tahlil kilish va baxolash savoli)**

2. O‘qituvchi mavzuni belgilab beradi.

3. O‘qituvchi kubikni stolga tashlaydi. Qaysi so‘z chiqsa, unga tegishli savolni beradi.

“KWHL” metodi

Metodning maqsadi: Mazkur metod tinglovchilarda yangi axborotlar tizimini qabul qilish va bilimlarni tizimlashtirish maqsadida qo‘llaniladi, shuningdek, bu metod tinglovchilar uchun mavzu bo‘yicha quyidagi jadvalda berilgan savollarga javob topish mashqi vazifasini belgilaydi.

Izoh. *KWHL:*
Know – nimalarni bilaman?
Want – nimani bilishni xohlayman?
How - qanday bilib olsam bo‘ladi?
Learn - nimani o‘rganib oldim?.

“KWHL” metodi	
1. Nimalarni bilaman: -	*2. Nimalarni bilishni xohlayman, nimalarni bilishim kerak:* -
3. Qanday qilib bilib va topib olaman: -	*4. Nimalarni bilib oldim:* -

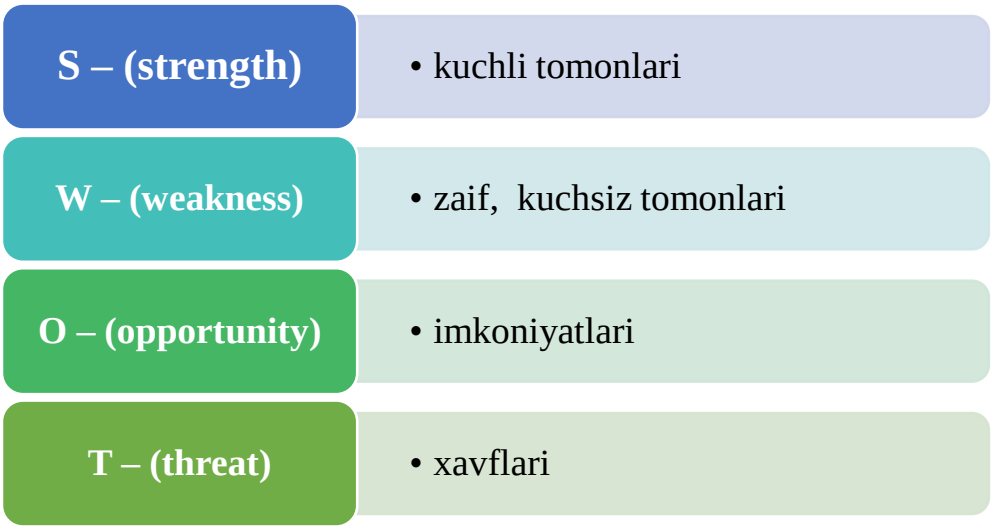
“5W1H” metodi

Metodning maqsadi: Mazkur metod tinglovchilarda yangi axborotlar tizimini qabul qilish va bilimlarni tizimlashtirish maqsadida qo‘llaniladi, shuningdek, bu metod tinglovchilar uchun mavzu bo‘yicha qo‘yidagi jadvalda berilgan oltita savollarga javob topish mashqi vazifasini belgilaydi.

What?	Nima? (ta’rifi, mazmuni, nima uchun ishlatiladi)	
Where?	Qayerda (joylashgan, qayerdan olish mukin)?	
What kind?	Qanday? (parametrlari, turlari mavjud)	
When?	Qachon? (ishlatiladi)	
Why?	Nima uchun? (ishlatiladi)	
How?	Qanday qilib? (yaratiladi, saqlanadi, to‘ldiriladi, tahrirlash mumkin)	

“SWOT-tahlil” metodi.

Metodning maqsadi: mavjud nazariy bilimlar va amaliy tajribalarni tahlil qilish, taqqoslash orqali muammoni hal etish yo‘llarini topishga, bilimlarni mustahkamlash, takrorlash, baholashga, mustaqil, tanqidiy fikrlashni, nostandart tafakkurni shakllantirishga xizmat qiladi.



“VEYER” metodi

Metodning maqsadi: Bu metod murakkab, ko‘ptarmoqli, mumkin qadar, muammoli xarakteridagi mavzularni o‘rganishga qaratilgan. Metodning mohiyati shundan iboratki, bunda mavzuning turli tarmoqlari bo‘yicha bir xil axborot beriladi va ayni paytda, ularning har biri alohida aspektlarda muhokama etiladi. Masalan, muammo ijobiy va salbiy tomonlari, afzallik, fazilat va kamchiliklari, foyda va zararlari bo‘yicha o‘rganiladi. Bu interfaol metod tanqidiy, tahliliy, aniq mantiqiy fikrlashni muvaffaqiyatli rivojlantirishga hamda o‘quvchilarning mustaqil g‘oyalari, fikrlarini yozma va og‘zaki shaklda tizimli bayon etish, himoya qilishga imkoniyat yaratadi. “Veyer” metodidan ma’ruza mashg‘ulotlarida individual va juftliklardagi ish shaklida, amaliy va seminar mashg‘ulotlarida kichik guruhlardagi ish shaklida mavzu yuzasidan bilimlarni mustahkamlash, tahlil qilish va taqqoslash maqsadida foydalanish mumkin.

Metodni amalga oshirish tartibi:

trener-o‘qituvchi ishtirokchilarni 5-6 kishidan iborat kichik guruhlarga ajratadi;

trening maqsadi, shartlari va tartibi bilan ishtirokchilarni tanishtirgach, har bir guruhga umumiy muammoni tahlil qilinishi zarur bo‘lgan qismlari tushirilgan tarqatma materiallarni tarqatadi;

har bir guruh o‘ziga berilgan muammoni atroflicha tahlil qilib, o‘z mulohazalarini tavsiya etilayotgan sxema bo‘yicha tarqatmaga yozma bayon qiladi;

navbatdagi bosqichda barcha guruhlar o‘z taqdimotlarini o‘tkazadilar. Shundan so‘ng, trener tomonidan tahlillar umumlashtiriladi, zaruriy axborotlr bilan to‘ldiriladi va mavzu yakunlanadi.

Muammoli savol					
1-usul		**2-usul**		**3-usul**	
afzalligi	kamchiligi	afzalligi	kamchiligi	afzalligi	kamchiligi
Xulosa:					

Muammoli savol					
1-usul		**2-usul**		**3-usul**	
afzalligi	kamchiligi	afzalligi	kamchiligi	afzalligi	kamchiligi
Xulosa:					

“Keys-stadi” metodi

«Keys-stadi» - inglizcha so‘z bo‘lib, («case» – aniq vaziyat, hodisa, «stady» – o‘rganmoq, tahlil qilmoq) aniq vaziyatlarni o‘rganish, tahlil qilish asosida o‘qitishni amalga oshirishga qaratilgan metod hisoblanadi. Mazkur metod dastlab 1921 yil Garvard universitetida amaliy vaziyatlardan iqtisodiy boshqaruv fanlarini o‘rganishda foydalanish tartibida qo‘llanilgan. Keysda ochiq axborotlardan yoki aniq voqea-hodisadan vaziyat sifatida tahlil uchun foydalanish mumkin.

“Keys metodi” ni amalga oshirish bosqichlari

Ish bosqichlari	Faoliyat shakli va mazmuni
1-bosqich: Keys va uning axborot ta’minoti bilan tanishtirish	✓ yakka tartibdagi audio-vizual ish; ✓ keys bilan tanishish(matnli, audio yoki media shaklda); ✓ axborotni umumlashtirish; ✓ axborot tahlili; ✓ muammolarni aniqlash
2-bosqich: Keysni aniqlashtirish va o‘quv topshirig‘ni belgilash	✓ individual va guruhda ishlash; ✓ muammolarni dolzarblik iyerarxiyasini aniqlash; ✓ asosiy muammoli vaziyatni belgilash
3-bosqich: Keysdagi asosiy muammoni tahlil etish orqali o‘quv topshirig‘ining yechimini izlash, hal etish yo‘llarini ishlab chiqish	✓ individual va guruhda ishlash; ✓ muqobil yechim yo‘llarini ishlab chiqish; ✓ har bir yechimning imkoniyatlari va to‘siqlarni tahlil qilish; ✓ muqobil yechimlarni tanlash
4-bosqich: Keys yechimini yechimini shakllantirish va asoslash, taqdimot.	✓ yakka va guruhda ishlash; ✓ muqobil variantlarni amalda qo‘llash imkoniyatlarini asoslash; ✓ ijodiy-loyiha taqdimotini tayyorlash; ✓ yakuniy xulosa va vaziyat yechimining amaliy aspektlarini yoritish

“Assesment” metodi

Metodning maqsadi: mazkur metod ta’lim oluvchilarning bilim darajasini baholash, nazorat qilish, o‘zlashtirish ko‘rsatkichi va amaliy ko‘nikmalarini tekshirishga yo‘naltirilgan. Mazkur texnika orqali ta’lim oluvchilarning bilish faoliyati turli yo‘nalishlar (test, amaliy ko‘nikmalar, muammoli vaziyatlar mashqi, qiyosiy tahlil, simptomlarni aniqlash) bo‘yicha tashhis qilinadi va baholanadi.

Metodni amalga oshirish tartibi:

“Assesment”lardan ma’ruza mashg‘ulotlarida talabalarning yoki qatnashchilarning mavjud bilim darajasini o‘rganishda, yangi ma’lumotlarni bayon qilishda, seminar, amaliy mashg‘ulotlarda esa mavzu yoki ma’lumotlarni o‘zlashtirish darajasini baholash, shuningdek, o‘z-o‘zini baholash maqsadida individual shaklda foydalanish tavsiya etiladi. Shuningdek, o‘qituvchining ijodiy yondashuvi hamda o‘quv maqsadlaridan kelib chiqib, assesmentga qo‘shimcha topshiriqlarni kiritish mumkin.

Har bir katakdagi to‘g‘ri javob 5 ball yoki 1-5 balgacha baholanishi mumkin.

Test

Muammoli vaziyat

Tushuncha tahlili (simptom)

Amaliy vazifa

“Insert” metodi

Metodni amalga oshirish tartibi:

➢ o‘qituvchi mashg‘ulotga qadar mavzuning asosiy tushunchalari mazmuni yoritilgan matnni tarqatma yoki taqdimot ko‘rinishida tayyorlaydi;

➢ yangi mavzu mohiyatini yorituvchi matn ta’lim oluvchilarga tarqatiladi yoki taqdimot ko‘rinishida namoyish etiladi;

➢ ta’lim oluvchilar individual tarzda matn bilan tanishib chiqib, o‘z shaxsiy qarashlarini maxsus belgilar orqali ifodalaydilar. Matn bilan ishlashda talabalar yoki qatnashchilarga quyidagi maxsus belgilardan foydalanish tavsiya etiladi:

Belgilar	Matn
“V” – tanish ma’lumot.	
“?” – mazkur ma’lumotni tushunmadim, izoh kerak.	
“+” bu ma’lumot men uchun yangilik.	
“– ” bu fikr yoki mazkur ma’lumotga qarshiman?	

Belgilangan vaqt yakunlangach, ta’lim oluvchilar uchun notanish va tushunarsiz bo‘lgan ma’lumotlar o‘qituvchi tomonidan tahlil qilinib, izohlanadi, ularning mohiyati to‘liq yoritiladi. Savollarga javob beriladi va mashg‘ulot yakunlanadi.

III-BO‘LIM

NAZARIY MATERIALLAR

III. NAZARIY MATERIALLAR

1-ma’ruza. Kriptografik protokollar faniga kirish

Reja:

1. Kriptografik protokol va ularning vazifalari.
2. Sodda autentifikatsiyalash protokollari
3. Simmetrik va asimmetrik kalitlar yordamida autentifikatsiyalash

Kriptografik protokol va ularning vazifalari

Ikki yoki undan ortiq tomonlar bajaradigan, biror-bir masalani yechish uchun loyihalashtirilgan harakatlar ketma-ketligi protokol hisoblanib, “harakatlar ketma-ketligi” so‘zi protokol boshidan to oxiriga qadar ketma-ket bajarilishini bildiradi. Har bir harakat navbatma-navbat bajariladi, shuningdek keyingi harakatlar oldingi harakatlar tugagandan keyingina bajarilishni boshlaydi. “Ikki yoki undan ortiq tomonlar bajaradigan” so‘zi protokol bajarilishi uchun kamida ikki tomonning ishtiroki kerakligini bildiradi. Protokolni yakka tartibda bajarib bo‘lmaydi. Nihoyat “biror-bir masalani yechish uchun loyihalashtirilgan” so‘zi protokol qandaydir natijaga olib borishi kerakligini anglatadi.

Protokolga o‘xshash, ammo biror-bir natijaga olib bormaydigan harakatlar ketma-ketligi – bu protokol emas, aksincha bekorga ketkazilgan vaqt hisoblanadi.

Protokollar quyidagi hususiyatlarga ega bo‘lishi kerak:

- amallar boshidan oxirigacha tartibga ega, ya’ni hech bir amal undan oldingisi tugamaguncha boshlanmasligi kerak;
- protokolning har bir ishtirokchisi protokolga bo‘ysunishi shart;
- har bir amal aynan aniqlangan bo‘lib, ikki xil ma’no kasb etmasligi kerak, har bir vaziyatdan aniq chiqish yo‘li bo‘lishi kerak;
- protokol uchun bitta ishtirokchining bo‘lishi yetarli emas (ikki yoki undan ortiq bo‘lishi kerak);
- protokolning barcha ishtirokchilari avvaldan bajariladigan amallar ketma-ketligi bilan tanish va uni bajarishga rozi bo‘lishlari kerak;
- tomonlar biror bir aniq vazifani bajaradilar – bu maqsadsiz amallar bo‘lmasligi kerak.
- protokol to‘liq bo‘lishi lozim – unda aniq harakatlar keltirilishi kerak.

Har kunlik hayotimizda formal bo‘lmagan protokollar deyarli hamma joyda ishlatiladi: masalan, telefon orqali tort buyurish, saylovlarda ovoz berish va h.z. Odamlar bu protokollar haqida uncha o‘ylashmaydi. Ular uzoq vaqt mobaynida evolyutsiyalashgan, ulardan qanday foydalanishni hamma biladi va ular ishonchli ishlaydi.

Hozir ko‘pgina odamlar shaxsiy muloqot uchun kompyuter tarmog‘ini ma’qul ko‘radilar. Ammo odamlar ko‘p o‘ylamay qiladigan ishlarni qilishlari uchun kompyuterlar uchun formal protokollar kerak bo‘ladi. Masalan, odamlar bir joydan boshqa joyga ko‘chib ketsa, ular u yerdagi o‘zgarishlarga moslashadi, ammo kompyuterlar bunday moslashuvchanlik qobiliyatiga ega emaslar.

Kompyuter tarmog‘idan foydalanuvchilar va kompyuter tarmog‘i yaratuvchilarining rostgo‘yligiga har doim ham ishonib bo‘lmaydi. Albatta ularning ko‘pchiligi to‘g‘ri so‘z odamlar, ammo ularning orasida bir nechta buzg‘unchisi bo‘lsa ham katta talofot keltirishi mumkin. Protokollar formalizatsiyasi buzg‘unchilar tomonidan protokolni ochish uchun ishlatiladigan usullarni aniqlash imkonini beradi. Buning natijasida bardoshli protokollarni ishlab chiqish imkoniyati tug‘iladi.

Harakatlarni formalizatsiya qilishdan tashqari protokollar vazifani yechish jarayonini mexanizmni yechish jarayonidan ajratib olish imkonini beradi. Masalan, IBM shaxsiy kompyuteri va VAX meynfreymlarida bir xil aloqa protokollari ishlatiladi.

Protokollar ishlashini namoyish qilish uchun bir-nechta ishtirokchilar yordamidan foydalanamiz (1-jadval).

1.1-jadval

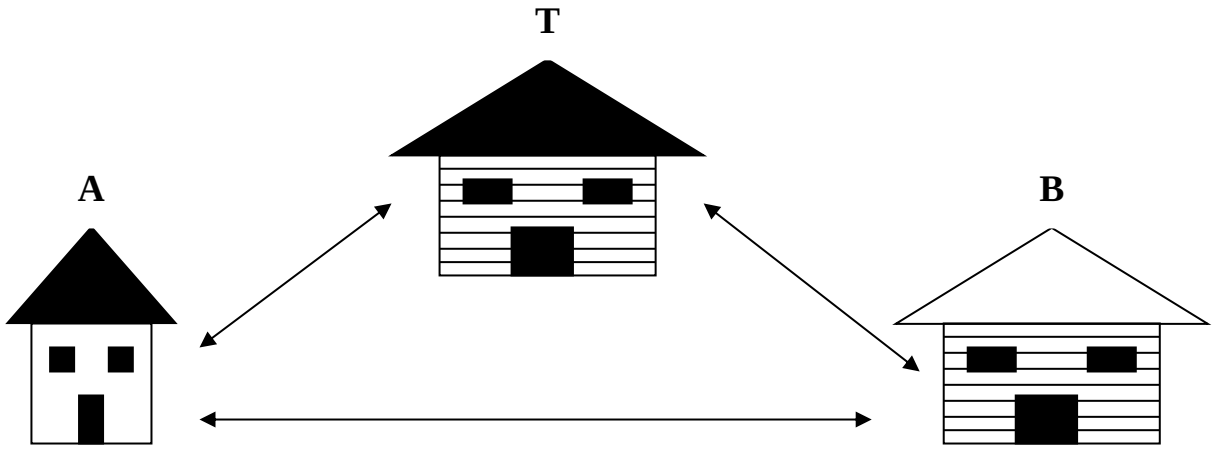
Protokol ishtirokchilari

Ishtirokchilar	Faoliyati	Belgilanishi
Alisa	Barcha protokollarning birinchi ishtirokchisi	**A**
Bob	Barcha protokollarning ikkinchi ishtirokchisi	**V**
Kerol	Uch va to‘rt tomonli protokollar ishtirokchisi	**K**
Deyv	To‘rt tomonli protokollar ishtirokchisi	**D**
Trent	Ishonchli vositachi	**T**
Yeva	Passiv buzg‘unchi	**E**
Mellori	Yomon niyatli aktiv buzg‘unchi	**M**

Jarayondagi asosiy ishtirokchilar – **A** va **V** bo‘lib, ular umumiy qabul qilingan barcha ikki tomonlama protokollarni bajaradilar. Qoida bo‘yicha barcha protokollarni **A** initsializasiya qiladi, **B** esa javob beradi. Agar protokol 3 va 4 tomonlar ishtirokini talab qilsa, o‘yinga **K** va **D** qo‘shiladilar. Boshqa ishtirokchilar maxsus yordamchi rolni bajarishadi.

Vositachi yordamidagi protokollar

Vositachi deb protokolni bajarilishini yakuniga yetkazishga ishonch bildirilgan manfaatdor bo‘lmagan uchinchi tomonga aytiladi (1.1-rasm). Vositachining “manfaatdor bo‘lmasligi” protokol bajarilishining natijasi hamda protokol ishtirokchisining hech biri u uchun ahamiyatga ega emasligini bildiradi. “Ishonch bildirish” so‘zi protokolning barcha ishtirokchilari vositachining so‘zlarini haqiqat deb qabul qilishini, uning hamma harakatlarini to‘g‘ri deb bilishligini, bundan tashqari vositachi protokoldagi o‘zining qismini bajarishiga ishonishligini bildiradi. Vositachilar bir-biriga ishonchi bo‘lmagan 2 tomonga protokolni bajarilishiga yordam beradilar.



1.1-rasm. Vositachi yordamidagi protokol

Real hayotda vositachi sifatida ko‘p hollarda advokatlarni tanlashadi.

Masalan, **A** unga begona bo‘lgan **B** ga avtomobil sotyapti. **B** chek yordamida pulini to‘lamoqchi, ammo **A** da chekning haqiqiyligini tekshirish imkoni yo‘q. Shuning uchun **B** ga mulkka egalik huquqini berishdan oldin **A** chek orqali pulini olmoqchi. **B** ham **A** ga o‘xshab egalik huquqini olmasdan turib chekni berishni xohlamaydi.

Bu ishda advokatning ishtiroki ikkala tomonni ham qoniqtiradi. U **A** va **B** ga bir-birini alday olmasligini kafolatlovchi quyidagi protokolni bajarilishiga yordam beradi:

1 **A** egalik xuquqini advokatga beradi.

2 **B** chekni **A** ga beradi.

3 **A** chekni depozitga qo‘yadi.

4 Chekni to‘lash uchun kerakli vaqt o‘tgandan keyin advokat **B** ga egalik huquqini beradi. Agar ma’lum vaqt mobaynida chek to‘lanmasa, **A** bu faktni advokatga isbotlab beradi va u egalik

huquqini **A** ga qaytarib beradi.

Bu protokolda **A** chek to‘langunga qadar advokat egalik huquqini **V** ga bermasligiga va agar chek bo‘yicha to‘lov amalga oshmasa egalik huquqini **A** ga qaytarib berishiga ishonadi. **V** chek to‘lanmasdan turib egalik huquqini advokatda turishiga ishonadi va u chek to‘langandan keyin shu zahoti egalik huquqini **V** ga beradi. Advokatni chekning to‘lovi qiziqtirmaydi. Ixtiyoriy holatda u protokoldagi o‘zining qismini bajaradi, chunki ishning qanday yakunlanishiga qaramay o‘zining xizmat haqini oladi.

Bu misolda advokat vositachi rolini o‘ynaydi. Advokatlar ko‘pincha shaxs rolida ishtirok etishadi, bunda ikki shaxs orasidagi munosabatlar tartibga solingunga qadar hisob ularning qo‘lida turadi. Bundan tashqari, advokatlar ko‘pincha vasiyatnoma yozilganda, ba’zan esa savdo-sotiq shartnomasi tuzilganda vositachi sifatida ishtirok etadilar. Sotuvchi va oluvchi orasida ba’zan vositachi sifatida turli birjalar ham ishtirok etadi.

Quyidagi misolda vositachi sifatida bank ham ishtirok etishi mumkin. **A** dan avtomobil sotib olish uchun **V** kafolatlangan chekni ishlatishi mumkin:

1 **V** chekni yozib bankka beradi.

2 **V** ning hisobida chekni to‘lash uchun yetarli miqdordagi pullarni rezervlab, bank chekni tasdiqlaydi va uni **V** ga qaytaradi.

3 **A** egalik huquqini **V** ga beradi, **V** esa **A** ga kafolatlangan chekni beradi.

4 **A** chekni depozitga o‘tkazadi.

A bankning kafolatlariga ishonganligi uchun bu protokol ishlaydi. **A** bank u uchun **V** ning pullarini ushlab qolishiga va ularni xavfli operatsiyalarni bajarishda mablag‘ bilan ta’minlashga ishlatmasligiga ishonadi.

Yana bir bo‘lishi mumkin bo‘lgan vositachi – notarius. Qachonki **V** **A** dan notariusda tasdiqlangan hujjatni olsa, u **A** hujjatga o‘z ixtiyori va o‘z qo‘li bilan imzo chekkaniga ishonadi. Kerak bo‘lganda notarius bu faktni sudda tasdiqlashi mumkin.

Arbitrli protokollar

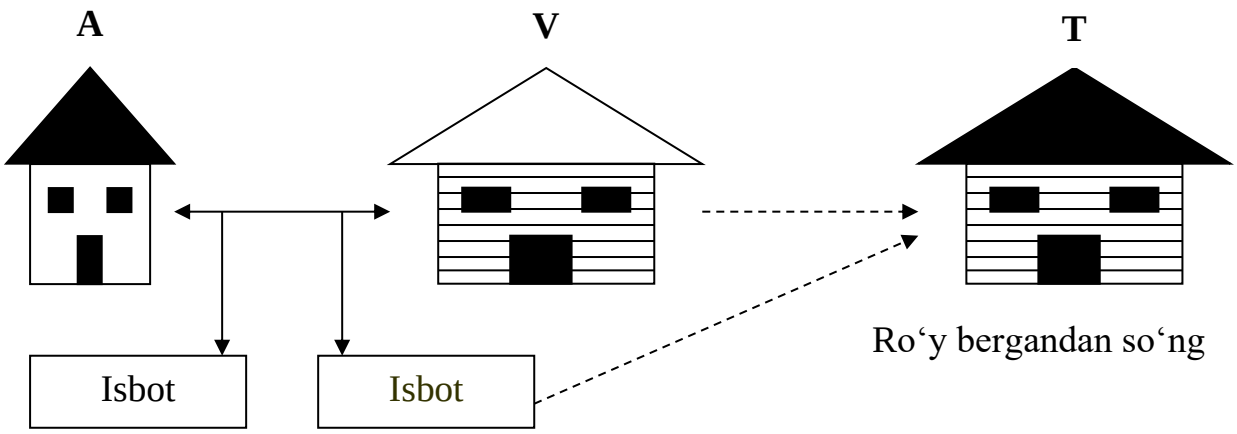
Vositachini yollash katta mablag‘ talab etganligi uchun, vositachi ishtirok etgan protokollarni hiyla pastroq darajali ikkita qism protokolga ajratish mumkin. Birinchisi vositachisiz protokol hisoblanadi, bunda tomonlar protokolni bajarish niyatida bo‘lgan hollardagina ishlaydi. Ikkinchisi, faqat ayrim hollarda ijro etiladigan – qachon tomonlar orasida kelishmovchilik kelib chiqsa, vositachi yordamidagi protokollar hisoblanadi. Bu protokolda maxsus turdagi vositachi ishtirok etadi – bu arbitr (1.2-rasm).

Arbitr xuddi vositachi kabi qiziqmaydigan va ishonchli protokolning uchinchi tomoni hisoblanadi. Vositachidan farqli ravishda u har bir protokolning bajarilishida ishtirok etishi shart emas. Arbitr faqat protokolning to‘g‘ri bajarilganligini tekshirish uchun taklif qilinadi.

Professional arbitrlarga misol qilib hakamlarni (sudya) keltirish mumkin. Notariuslardan farqli ravishda hakamlarga faqat kelishmovchilik kelib chiqqanda murojaat qilinadi. **A** va **V** hakam ishtirokisiz shartnoma tuzishlari mumkin, agar ixtiyoriy bir tomon ikkinchi tomonni sudga bermasa, hakam shartnoma haqida hech qachon bilmaydi.

Shartnoma imzolash protokolini quyidagicha ifodalash mumkin:

Vositachisiz protokol (har doim bajariladi):



1.2-rasm. Arbitrli protokol

1 **A** va **V** shartnoma shartlariga rozilik bildirishadi;
2 **A** shartnomani imzolaydi;
3 **V** shartnomani imzolaydi.
Arbitr ishtirokidagi protokol (kelishmovchilik kelib chiqqanda bajariladi):
1 **A** va **V** sudda ishtirok etishadi;
2 **A** o‘zining dalillarini keltiradi;
3 **V** o‘zining dalillarini keltiradi;
4 Dalillarga tayanib hakam o‘z qarorini chiqaradi.

Arbitrli kompyuter protokollari ham ma’lum. Bu protokollar tomonlarning rostgo‘yligi taxminiga tayanishadi. Biroq agar kimdir firibgarlikni sezsa, uchinchi ishonchli tomon **T** mavjud ma’lumotlar massivi asosida aldovni fosh qilishi mumkin. Bundan tashqari yaxshi arbitrajli protokol arbitrga firibgarning shaxsini xam aniqlash imkonini beradi. Shunday qilib arbitrajli protokollar firibgarlikni oldini olmaydi, balki uni aniqlaydi.

O‘ziga yetarli protokollar

O‘ziga yetarli protokollar - eng yaxshi protokol turi hisoblanadi (3-rasm). Tomonlar to‘g‘riligi protokollarning o‘zi bilan kafolatlanadi. Protokolning bajarilishi uchun vositachi kerak emas, kelishmovchiliklarni bartaraf etish uchun esa – arbitr (hakam). Kelishmovchiliklarning yo‘qligini (mavjud emasligini) protokol konstruksiyasining o‘zi ta’minlaydi. Agar tomonlarning biri g‘irromlik qilishga harakat qilsa, boshqa tomon shu zahoti aldovni aniqlaydi va protokol bajarilishi to‘xtatiladi.

1.3-rasm. O‘ziga yetarli protokol

Kriptografik protokol kriptoalgoritmdan va shifrlash kalitlaridan foydalanishni belgilab beradigan qoidalar va protseduralar to‘plamidir. Tomonlar bir-biriga ishonib do‘st bo‘lishi mumkin yoki aksincha bir-biriga ishonmasligi, ya’ni buzg‘unchi bo‘lishi mumkin. Kriptografik protokol tarkibiga ma’lum bir kriptografik algoritm kiradi, ammo protokollar faqatgina

maxfiylikni ta’minlash uchun mo‘ljallanmagan. Protokollarda kriptografiyani ishlatishdan maqsad firibgarlik va noqonuniy eshitishni aniqlash yoki unga yo‘l qo‘ymaslik.

Umumiy qoida shunday:

Protokolda keltirilgandan tashqari ko‘proq narsa bilish yoki o‘zgartirish mumkin emas.

Ba’zi protokollarda ishtirokchilardan biri ikkinchisini aldashi mumkin. Boshqa protokollarda esa buzg‘unchi protokolni buzishi yoki undagi maxfiy ma’lumotni bilib olishi mumkin.

Kriptografik protokollar (KP) quyidagi bir necha ishtirokchilardan tarkib topgan taqsimlangan algoritmdir:

- odamlar;
- kompyuter dasturlari;
- kompyuterlar va hisoblash komplekslari;
- ma’lumotlar bazasi;
- aloqa tarmoqlari;
- autentifikatsiya vositalari;
- va boshqalar.

KPning har bir ishtirokchisi ma’lum algoritmlar ketma-ketligiga mos ravishda ish bajaradi. Har bir ishtirokchi tomonidan bajariladigan amal quyidagicha bo‘lishi mumkin:

- boshqa ishtirokchiga (yoki ishtirokchilar guruhiga) *xabarni yuborish;*
- boshqa ishtirokchidan *xabar qabul qilish;*
- *ichki amal,* ya’ni ishtirokchilar amalga oshiradigan ba’zi hisoblash ishlari.

KP ishtirokchilari 3 sinfga bo‘linadi:

1. *Odatdagi (qonuniy) ishtirokchilar* (**A, V** va hakozo belgilar ko‘rinishida ifodalanadi, indekslar bilan ham kelishi mumkin).
2. *Ishonchli vositachi* (**T** belgisi ko‘rinishida ifodalanadi, indeks bilan ham kelishi mumkin).
3. Quyidagi ikki sinfga bo‘linuvchi *buzg‘unchilar:*

a) *Passiv buzg‘unchilar* (**E** belgisi ko‘rinishida ifodalanadi, indeks bilan ham kelishi mumkin).

Passiv buzg‘unchi boshqa ishtirokchilarga yuborgan xabarni ushlab olishi, o‘g‘irlashi va tahlil qilishi mumkin.

b) *Aktiv buzg‘unchilar* (**M** belgisi ko‘rinishida ifodalanadi, indeks bilan ham kelishi mumkin).

Aktiv buzg‘unchi quyidagi amallarni bajarishi mumkin:

- boshqa ishtirokchilarga yuborilgan xabarni ushlab olishi va tahlil qilishi;
- yuborilgan xabarni o‘zgartirishi yoki o‘chirishi;
- yangi xabarni hosil qilib, boshqa ishtirokchilarga yuborishi;
- o‘zini boshqa ishtirokchi qilib ko‘rsatishi (bunday aktiv buzg‘unchilarni *firibgar* deb nomlashadi).

Shunday qilib KP – bu shunday protokolki, unda kriptografik algoritmlar qo‘llaniladi, va u biror bir kriptografik masalani yechish uchun xizmat qiladi.

Nazariy kriptografiyada protokol m ta kirish va $n \le m$ ta chiqishga ega bo‘lgan “qora quti” sifatida qaraladi (1.4-rasm):

$$\bar{x} = \begin{pmatrix} x_1 \to \\ x_2 \to \\ \dots \to \\ x_m \to \end{pmatrix} \boxed{N} \begin{matrix} \to f_1(x_1,\dots,x_m) = f_1(\overline{x}) \\ \to f_2(x_1,\dots,x_m) = f_2(\overline{x}) \\ \to \quad \dots\dots\dots\dots \\ \to f_n(x_1,\dots,x_m) = f_n(\overline{x}) \end{matrix}$$

1.4-rasm. Nazariy kriptografiyada protokol sxemasi

Avval ta’kidlab o‘tilganidek, protokol – bu kriptografik algoritm va boshlang‘ich elementlariga (primitivlarga) nisbatan yuqori darajadagi struktura. Bitta protokolda turli elementlar va algoritmlar ishlatilishi mumkin. Ulardan bittasining yoki bir nechtasining noto‘g‘ri qo‘llanilishi butun protokol xavfsizligini yo‘qolishiga olib kelishi mumkin. Shunday vaziyatga oddiy misol keltirishimiz mumkin: protokolda uzatilayotgan ma’lumotlarni yopish uchun biror shifrlash algoritmidan foydalanilayotgan, ammo uning kaliti protokol bajarilish jarayonida ochiq holda uzatilayotgan bo‘lsin. Ma’lumki, bu yerda protokolning noto‘g‘ri tuzilishi shifr qanchalar turg‘un bo‘lmasin xavfsizlikning yo‘qolishiga olib keladi.

Kriptografik protokollarning berilgan talablarga ko‘ra to‘g‘ri tuzilishi odatda ikki maqsadni ko‘zlaydi: tashqaridagi buzg‘unchidan himoya qilishni va o‘zaro bir birini aldashdan himoya qilishni. Kriptografik protokollar – bu ishtirokchilar orasidagi shunday o‘zaro bog‘lanish protsedurasiki, uning natijasida qonuniy ishtirokchilar o‘z maqsadlariga erishadilar, buzg‘unchi esa maqsadiga yeta olmaydi.

Bardoshli protokol – uni “sindirish” uchun urinishlarga nisbatan “ichki” konstruktiv turg‘unlikka ega bo‘lgan protokol. Protokolni ko‘pchilik ishtirokchilar unga to‘g‘ri amal qilganda, ishonchli natija oladigan qilib tuzish mumkin.

Turg‘un protokol – buzg‘unchi ishtirokchilarning maxfiy ma’lumotining biror qismini bilganda ham xavfsizlikni saqlay oladigan protokol.

Bitta protokol aynan bitta ishtirok etuvchi shaxslar tomonidan biror vaqt oralig‘ida bir necha marta bajarilishi mumkin. Seans – bu protokolning bir marta bajarilishi. Protokol raundi – bu bir martali ikki tomonlama xabar yuborish. Raund kontekstiga bog‘liq holda ikki yoki undan ortiq xabarlarni jo‘natishni o‘z ichiga olishi mumkin. Ba’zan protokolning ichida siklik konstruksiyalar ham uchraydi: bunda bir martali siklning bajarilishi raund deb ataladi.

Kriptografik protokollarning tavsifi, odatda ishtirokchilarning hatti-harakati tavsifidan tashqari talab qilinayotgan algoritmlarning xarakteristikalarini, protokolning to‘g‘ri ishlashi uchun talab qilinadigan boshlang‘ich shartlarni o‘z ichiga oladi.

Umumiy holda protokol ishtirokchilari ikki guruhga bo‘linadi:

1. protokol masalasini bevosita yechuvchilar;
2. birinchi guruh ishtirokchilariga xizmat ko‘rsatuvchilar.

Birinchi guruhga kiruvchi ishtirokchilar soniga bog‘liq ravishda protokollar ikki tomonlama va ko‘p tomonlama bo‘ladi. O‘z navbatida bunday ishtirokchilar vijdonli va vijdonsizga bo‘linadi. Vijdonsizlarga protokol masalalarini yechishga atayin xalaqit beruvchilar kiradi, ya’ni bular dushmanlar, buzg‘unchilar, qasddan bo‘lmasada xato o‘tkazishga sabab bo‘lganlardir.

Amaliyotda odatda biror bir ishtirokchining protokolda keltirilgan amallardan chetlashishi atayinmi yoki tasodifiymi ekanligini aniqlash juda ham qiyin. Shuning uchun amaliyotda protokolda buzg‘unchi ishtirok etmoqda degan yetarlicha kuchli taxmin qabul qilinadi va bu xavfni amalga oshirilishini hisobga olgan holda protokol tuziladi. Buzg‘unchi turli masalalarda turli imkoniyatlarga ega bo‘lishi mumkin: abonentlar bilan boshqa ishtirokchilar nomidan bog‘lanish, axborot almashinuviga aralashish. Buzg‘unchi abonentlardan biri yoki u bilan til biriktirgan bir necha kishi bo‘lishi mumkin.

Yuqorida bayon qilinganlarga asoslanib *quyidagi xulosaga kelish* mumkin: Protokolga hujum qilish usuliga ko‘ra buzg‘unchi passiv yoki aktiv (faol) bo‘lishi mumkin. Passiv buzg‘unchi faqat aloqa kanallarini eshitishi va bu kanallar orqali yuborilayotgan axborotlarni to‘laligicha yoki saralab saqlashi mumkin. Faol buzg‘unchi aloqa kanalida xabarlarni qo‘shib qo‘yishi, o‘zgartirishi, olib tashlashi va xattoki protokol ishtirokchilarining maxfiy kalitining bir qismini qo‘lga kiritishi mumkin.

Kriptografik protokollar nazariyasining paydo bo‘lish tarixi kishilar amaliyotining tomonlar o‘rtasida ishonchsizlik mavjud bo‘lgan, qiziqishlar ustma-ust tushmay bir-birini aldash ehtimolligi mavjud bo‘lgan sohalarida yuzaga kelgan. Bunday sohalarga avvalam bor bank ishlari, notarial ishlar, savdo-sotiq, moliyaviy bitimlar, xizmat yozishmalari, hujjat aylanishi va hokazolar kiradi.

Kriptografik protokolning xossalari

KPning xossalari bir necha sinflarga bo‘linadi. Quyidagi sinfdagi xossalar eng dolzarb hisoblanadi.

1. *Aniqlik,* ya’ni:

- KP ishtirokchilari tomonidan amalga oshiriladigan hisoblashlarning to‘g‘riligi;
- ishtirokchilar tomonidan hisoblangan natijalarning berilgan o‘zaro nisbatga mos kelishi;
- va hokazo.

Aniqlik xossasi asosiy hisoblanadi, chunki ularning buzilishi natijasida, hattoki KP qolgan hamma xossalarga ega bo‘lsa ham, KPni ishlatib bo‘lmaydi.

2. *Xavfsizlik.*

Ushbu xossalar sinfi bir necha qismiy sinflarga bo‘linadi. Ulardan eng dolzarblari quyidagilar:

- *Yaxlitlik,* ya’ni qonuniy ishtirokchilar almashinadigan xabarni buzg‘unchi tomonidan o‘zgartirish harakatlari KPni bajarish jarayonida aniqlanishidan iborat;
- *Maxfiylik,* ya’ni KP ishi jarayonida axborotning mualliflashtirilmagan tarzda chiqib ketishining oldi olinganligidan iborat: KP ishlab turgan ixtiyoriy paytda buzg‘unchi shifrlangan xabarni tarkibi bilan tanishish imkoni bo‘lmasligi kerak.

3. *Turg‘unlik* **(**quyidagi hollarda):

- ma’lum hatti-harakatlardan ishtirokchining hatti-harakatlarini rad etishda;
- KP bajarilib turgan muhitda kutilmagan hatti-harakati holatida.

Shuningdek ushbu xossalar sinfiga KP ishlab turgan kompyuter tizimida nosozlikdan so‘ng normal ishlashini tezda ta’minlash qobiliyati kiradi.

4. *Samaradorlik* - KP ishlashi jarayonida xotira va vaqt resurslaridan samarali foydalanish. KPdagi amalga oshirilgan algoritmlarning optimalligi.

5. *Adaptatsiya* – ichki strukturasini o‘zgartirmasdan uni sozlagichi yordamida o‘zgartirish yo‘li bilan KP muhitining ozgina o‘zgarishiga moslashuvchanligi.

6. *Hujjatlashtirilganlik* - uni ishlatish shartida muhim o‘zgarish bo‘lganda KPga tezda o‘zgartirish kiritishga imkon beruvchi KP tavsifining tiniq va aniq hujjatlashtirilganligi (Masalan, juda ko‘p mumkin bo‘lgan kirish ma’lumotlarini kengaytirish yoki toraytirish holatida).

7. *Mobillilik va moslashuvchanlik,* ya’ni KPning turli konfiguratsiya va platformalarda yaxshi ishlash qobiliyati.

Kriptografik protokollarning sinflanishi

Kriptografik protokollarni sinflashda turli yondashuvlar mavjud. Quyida ulardan ba’zi birlari keltiriladi.

1. *Ishtirokchilar soniga ko‘ra sinflanish*:

– ikki tomonlama;
– uch tomonlama;
– ko‘p tomonlama.

2. *Yuboriladigan xabarlar soniga ko‘ra sinflanish:*

– interaktiv (o‘zaro xabarlar almashinuvi);
– nointeraktiv (faqat bir tomonlama yuborish). Nointeraktiv protokollar ko‘pincha sxema deb nomlanadi.

3. *Protokolning belgilangan maqsadiga ko‘ra sinflanishi:*

– xabar yaxlitligini ta’minlovchi protokol:
  - manbani autentifikatsiya qilib;
  - manbani autentifikatsiya qilmasdan.
– ERI protokoli (sxemasi):
  - jamoaviy/shaxsiy ERI protokoli;
  - xabarni qayta tiklashli/qayta tiklashsiz;

- ko‘r-ko‘rona ERI protokoli;
- maxfiy ERI protokoli;
- soxtalashtirilganligini isbotlovchi (yoki soxtalashtirilgan-ligini isbotlash sifatiga ega bo‘lgan) ERI protokoli.

– Identifikatsiyalash protokoli (ishtirokchilarni autentifikatsiya qilish):
  - bir tomonlama autentifikatsiya;
  - ikki tomonlama (o‘zaro) autentifikatsiya.
– Maxfiy yuborish:
  - xabarlarni odatiy almashinishi;
  - keng qamrovli/sirkulyar yuborish;
  - sirlarning haqqoniy almashinuvi;
  - unutiladigan yuborish;
  - bitga (satr) bog‘lanuvchi protokol.
– Kalitlarni taqsimlash protokoli:
  - kalitlarni avvaldan taqsimlash protokoli (sxemasi);
  - kalitni yuborish protokoli (kalitlar almashinish);
  - kalitni birgalikda ishlab chiqish protokoli;
  - juftli/jamoaviy protokol;
  - sirni bo‘lishish protokoli (sxemasi);
  - telekonferensiya protokoli;
  - va hokazolar.

Kriptografik protokollarning vazifalari

Xavfli ochiq kompyuter tarmoqlarida va taqsimlangan kompyuter tarmoqlarida o‘zaro xavfsiz ma’lumot almashinuvini tashkillashtirish muhim vazifalardan biri bo‘lib, uni hal qilish uchun KPdan foydalanish mumkin.

Kriptografik protokollar quyidagi asosiy vazifalarni bajaradi:
- Ma’lumotlar manbasining autentifikatsiyasi;
- Tomonlar autentifikatsiyasi;
- Ma’lumotlar maxfiyligi;
- Rad etishning mumkin bo‘lmasligi;
- Qabul qilganlikning isboti bilan rad etishning mumkin bo‘lmasligi;
- Manbaning isboti bilan rad etishning mumkin bo‘lmasligi;
- Ma’lumotlar yaxlitligi;
- Qayta tiklashsiz ulanishning yaxlitligini ta’minlash;
- Qayta tiklashli ulanishning yaxlitligini ta’minlash;
- Foydalanishni chegaralash.

1.2-paragrafda bayon etilganlarga asoslanib kriptografik protokollarni bajaradigan asosiy vazifalariga qarab umumiy holda quyidagicha sinflash mumkin:
- shifrlash/shifrni ochish protokollari;
- ERI protokoli;
- identifikatsiya/autentifikatsiya protokoli (AP);
- kalitlarni autentifikatsiya qilib tarqatish protokoli.

Shifrlash/shifrni ochish protokollari. Bu sinfdagi protokol asosini shifrlash/shifr ochishning simmetrik yoki nosimmetrik algoritmi tashkil etadi. Shifrlash algoritmi jo‘natuvchi xabarni yuborayotganda amalga oshiriladi, natijada xabar ochiq holatdan shifrlangan holatga almashtiriladi. Shifrni ochish algoritmi qabul qiluvchi xabarni olayotganda amalga oshiriladi, natijada xabar shifrlangan holatdan ochiq holatga almashtiriladi. Shu tarzda maxfiylik xususiyati ta’minlanadi.

Odatda simmetrik shifrlash/shifrni ochish algoritmlarida uzatilayotgan xabarlarning

yaxlitlik xossasini saqlash uchun, uzatishda va qabul qilishda shifrlash kaliti qo‘llaniladigan imitohimoya qo‘shimchalarini tekshirishni imitohimoya qo‘shimchalarini hisoblovchi algoritmlar bilan birga qo‘llaniladi. Nosimmetrik shifrlash/shifrni ochish algoritmlarini qo‘llaganda yaxlitlik xossasi alohida ERIni hisoblash orqali, uzatishda va qabul qilishda qabul qilingan xabarni rad eta olmaslik va haqiqiylikni ta’minlovchi ERIni tekshirish yordamida amalga oshiriladi.

ERI protokoli. Bu sinfdagi protokol asosini yuborishda yuboruvchining yopiq kaliti yordamida ERIni hisoblash, qabul qilishda ochiq ma’lumotnomadan olinadigan va o‘zgartirishdan himoyalangan yopiq kalitga mos ochiq kalit yordamida ERIni tekshirish algoritmlari tashkil etadi. Protokolni tekshirish natijasi ijobiy bo‘lganda, qabul qilingan xabar, uning ERIsi va mos ochiq kalitlarni arxivlash amali bilan tugallanadi. Agar ERI rad qila olmaslik xususiyati uchun emas, balki faqat yaxlitlik va qabul qilingan xabarning haqiqiyligini ta’minlash uchun qo‘llanilsa, arxivlash amali bajarilmasligi mumkin. Bu holda tekshiruvdan so‘ng ERI o‘sha zahotiyoq yoki kutish davri chegarasi tugashi bilan o‘chirib tashlanadi.

Identifikatsiya/autentifikatsiya protokoli. Bu sinfdagi protokol asosini identifikatorga ega bo‘lgan identifikatsiya qilinuvchi obyektning faqatgina qayd etilgan obyektga ma’lum bo‘lgan maxfiy axborotni bilishligini tekshiradigan ba’zi algoritm tashkil etadi. Bunda tekshirish usuli bilvosita hisoblanadi, ya’ni bu maxfiy axborotni taqdim qilmasdan amalga oshiriladi.

Odatda har bir obyektning nomi (identifikatori) himoyalangan ma’lumotlar bazasiga yozilgan tizimdagi huquq va vositalar ro‘yxati bilan bog‘lanadi. Bu holatda identifikatsiya protokoli identifikatsiya qilingan obyekt buyurtirgan xizmatning vakolatli ekanligini tekshiradigan APgacha kengaytirilishi mumkin.

Agar identifikatsiya protokolida ERI ishlatilsa, u holda maxfiy axborot sifatida ERIning maxfiy kaliti ishlatiladi. ERIni tekshirish yopiq kalitni aniqlashga yo‘l qo‘ymaydigan, lekin bu yopiq kalitni ERI muallifiga ma’lum bo‘lishiga ishonch hosil qiladigan ma’lumot bo‘lgan unga mos ochiq kalit yordamida amalga oshiriladi.

Autentifikatsiya qilingan kalitlarni taqsimlash protokoli. Bu sinfdagi protokol ishtirokchilarni autentifikatsiya qilishni generatsiyalash va kanal bo‘yicha kalitlarni taqsimlash protokoli bilan qo‘shilib ketadi. Protokol ikki yoki uch ishtirokchilardan iborat: uchinchi ishtirokchi bo‘lib, kalitlarni taqsimlash va generatsiyalash markazi xizmat qiladi (*s* server). Protokol 3 bosqichdan iborat: generatsiya, qayd etish (registratsiya) va kommunikatsiya. Generatsiya bosqichida *s* server tizimning parametrlari qiymatlarini, shuningdek o‘zining ochiq va yopiq kalitini ham generatsiyalaydi. Qayd qilish bosqichida *s* server hujjatlar bo‘yicha ishtirokchilarni (shaxsan o‘zining kelishi yoki vakolatli shaxslar orqali) identifikatsiya qiladi. Buning uchun *s* server har bir obyekt uchun kalit va/yoki identifikatsiyalovchi axborotni generatsiya qiladi va kerakli tizim konstantalari va *s* serverining ochiq kalitidan (zaruriy holatda) iborat bo‘lgan xavfsizlik markerini shakllantiradi. Kommunikatsiya bosqichida umumiy seans kalitini shakllantirish bilan yakunlanadigan o‘zining autentifikatsiya qilingan kalitlar almashinuvi protokolini amalga oshiradi.

Yuqorida keltirilgan funksiyalardan kriptografik protokollarning asosiy vazifalari kelib chiqadi. Bu vazifalar quyidagilardan iborat:

- Autentifikatsiyaning turli rejimlarini ta’minlash;
- Kriptografik kalitlarni hosil qilish, taqsimlash va o‘zaro moslash;
- Ishtirokchilarning o‘zaro aloqasini himoyalash;
- Ishtirokchilar orasidagi javobgarlikni taqsimlash.

Protokol xavfsizligiga oid talablar

Protokol xavfsizligi tushunchasi xavfsizlikni tavsiflaydigan foydalana olishlik, konfidensiallik, yaxlitlik va boshqa xususiyatlarni bajarish kafolatini ta’minlash bilan ifodalanadi. Kriptografik protokollar turli hujumlarga nisbatan bardoshli bo‘lsa, bunday protokollar xavfsiz deyiladi. Hujumlar protokollarda ishlatiladigan kriptografik algoritmlarga, algoritm va protokollarni tadqiq qilishda ishlatiladigan kriptografik usullarga yoki protokollarning o‘ziga yo‘naltirilgan bo‘lishi mumkin. Kriptografik protokollarga qilinadigan asosiy hujumlarning

tasnifini keltiramiz.

1. *Ma'lum kalitlar bo'yicha hujum* – bu hujumda buzg'unchi protokolning avvalgi seanslarida qo'llanilgan bir qancha kalitlarga ega bo'lib, bu ma'lumotdan keyingi seanslarda yangi kalitlarni aniqlash maqsadida foydalanadi, masalan kalitlarni o'zgarishi qonunini aniqlashi mumkin.
2. *Seansni takrorlash usuli bo'yicha hujum* – bu hujumda buzg'unchi protokol seansini qisman yoki to'laligicha yozib oladi va keyingi seansda takroran qo'llaydi, ya'ni seansni yoki uning qismini bir oz vaqtdan so'ng "qaytadan o'ynaydi".
3. *O'zini boshqa shaxs nomidan ko'rsatish usuli bo'yicha hujum* – buzg'unchi o'ziga protokolning qonuniy ishtirokchilaridan birining o'xshashligini oladi.
4. *Lug'at bo'yicha hujum* – protokolda ishlatilish ehtimolligi katta bo'lgan kattaliklar yoki xabarlarni tanlash orqali hujum (masalan, parollarni tanlash, chunki odatda ular uchun oson topiladigan ma'lumotlar: familiya, ism, otasining ismi, telefon raqami, manzili va hokazolar olinadi).
5. *Oldindan qidirish usuli bo'yicha hujum* – bajarilishiga ko'ra lug'at bo'yicha hujumga o'xshab ketadi, ammo bunda biror bir kattalikning mumkin bo'lgan barcha qiymatlarni to'la tanlash orqali amalga oshiriladi va odatda xabarlarni shifrini ochish uchun ishlatiladi. Masalan, bank tomonidan tarnzaksiya bajarilayotgan bo'lib, ochiq shifrlash sxemasiga ko'ra shifrlangan tranzaksiya kattaligi 32 bitli maydonda ko'rsatilgan bo'lsin. Buzg'unchi ochiq shifrlash xususiyatidan kelib chiqib 2^{32} ta ochiq matin olib ularni shifrlashi mumkin. So'ngra 2^{32} shifrmatinning har birini buzg'unchi tomonidan kuzatilayotgan tarnzaksiya kattaligi bilan solishtirib, unga mos keluvchi ochiq matinni aniqlashi mumkin.
6. *Kanalga suqilib kirish usuli bo'yicha hujum* – buzg'unchi **M** qonuniy **A** va **V** ishtirokchilar orasidagi aloqa kanaliga shunday "suqilib" kiradiki, u **A** ishtirokchiga **V** bilan bog'lanayotganlik illyuziyasini va aksincha **V** ishtirokchiga **A** bilan bog'lanayotganlik illyuziyasini hosil qiladi. Haqiqatda esa ularning har biri **M** bilan bog'lanayotgan bo'lib, **M** har bir xabarni "o'zi orqali o'tkazib", ularni o'zgartirishi, ushlab qolishi, o'rnini almashtirishi va hokazo qilishi mumkin. Ma'lumki, bu hujumda buzg'unchi faol bo'ladi.

Protokolning obro'sizlantirilishi (kompromentatsiya) – bu protokol oldiga qo'yilgan maqsadlarga erishishga qodir bo'lmay, buzg'unchi protokol asosida yotadigan kattaliklarni va algoritmlarni "ochmasdan" turib faqt protokolni boshqarish yo'li bilangina ustunlikka ega bo'ladigan vaziyat.

Masalan, oqimli shifrlash qo'llanilayotgan bo'lsin. Protokolda uzatilayotgan xabarlar maxsus ko'rinishga egaligi ma'lum bo'lsin, ya'ni: boshlang'ich 20 bitda bir hisob raqamidan boshqa hisob raqamiga o'tkazilayotgan pul miqdorini ifodalovchi ma'lumot shifrlangan holda uzatilayotgan bo'lsin. Faol buzg'unchi boshlang'ich 20 bitni biron bir kattalik bilan bitma bit qo'shib, pul miqdorini bilmasdan turib o'zgartirishi mumkin.

Kriptografik protokol xavfsizligini tavsiflaydigan xususiyatlar yetarli darajada ko'p bo'lib, odatda protokollarning turli hujumlarga bardoshligi xususiyati xavfsizlik talablarini shakllanishiga olib keladi. Quyida protokol xavfsizligiga oid asosiy talablar keltirildi.

1. Autentifikatsiya (noommaviy):
   - Subyekt autentifikatsiyasi;
   - Xabar autentifikatsiyasi;
   - Takrorlanishdan himoya.
2. Ko'p adreslarga jo'natish yoki yozilish/ma'lum qilish xizmatiga ulanish paytida autentifikatsiya:
   - Ishtirokchini oshkor bo'lmagan tarzda (maxfiy) autentifikatsiya qilish;
   - Manbani autentifikatsiya qilish.
3. Mualliflashtirish (ishonchli uchinchi tomon).
4. Kalitni hamkorlikda generatsiyalash xususiyati:
   - Kalitni autentifikatsiyalash;
   - Kalit haqiqiyligini tasdiqlash;

- Orqaga o‘qishdan himoyalash;
- Yangi kalitlarni shakllantirish;
- Xavfsizlik parametrlari haqida kelishishning himoyalangan imkoni.

5. Konfidensiallik.
6. Anonimlik:
   - Identifikatorlarni eshitishdan himoya qilish (bog‘liq bo‘lmaslik);
   - Identifikatorlarni boshqa ishtirokchilardan himoya qilish.
7. “Xizmat ko‘rsatishni rad etish” hujumidan chekli himoyalanish.
8. Yuboruvchining invariantligi.
9. Avvalgi qilingan amallarni rad eta olmaslik:
   - Hisob berishlik;
   - Manbaning isboti;
   - Qabul qiluvchining isboti.
10. Xavfsiz vaqtinchalik xususiyat.

Xavfsizlik talablariga javob beradigan protokollarni yaratish uchun quyidagi asosiy yondashuvlar ham mavjud:

Mavjud nosimmetrik kriptotizimlar bardoshliligini ta’minlashga asos bo‘lgan hisoblash murakkab bo‘lgan masalalar asosan quyidagicha tasniflanadi:

- faktorlash muammosining murakkabligiga asoslangan kriptotizimlar;
- diskret logarifmlash muammosining murakkabligiga asoslangan kriptotizimlar;
- EECHda diskret logarifmlash muammosining murakkabligiga asoslangan kriptotizimlar;
- boshqa muammolarga asoslangan kriptotizimlar.

Sodda xavfsizlik protokollari

Sodda xavfsizlik protokoli sifatida xavfsizlik tashkilotlari binosiga kirishda foydalanilgan protokolni qaraylik. Tashkilotning har bir xodimi maxsus “znachok”ga ega bo‘lib, uni o‘zi bilan har doim olib yuradi. Binoga kirishda maxsus karta o‘quvchi qurilmaga znachokni qo‘yadi va PIN kodni kiritadi. Ushbu holda to‘liq xavfsizlik protokoli quyidagicha ifodalanadi:

1. Karta o‘quvchiga znachokni qo‘yish.
2. PIN kodni kiritish.
3. PIN kod to‘g‘ri ?
   - Ha, Binoga kirish mumkin.
   - Yo‘q, Xavfsizlik xodimi tomonidan qo‘lga olinasiz.

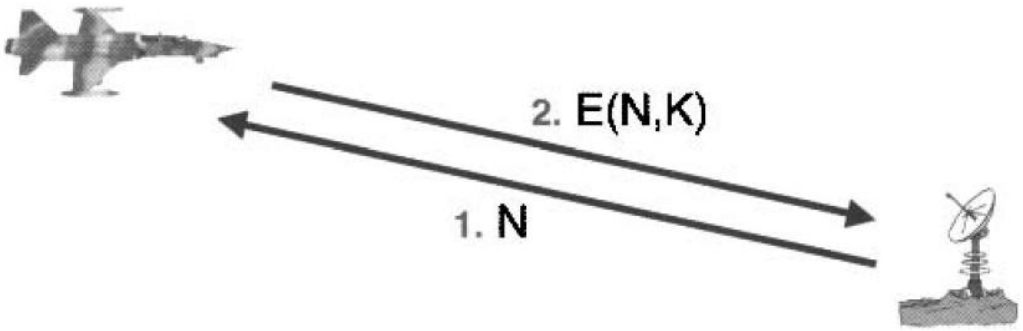
Xuddi shunga o‘xshash bo‘lgan xavfsizlik protokoli plastik kartadan bankomat orqali naqd pul olish jarayonida ham amalga oshiriladi.

Harbiy sohada ham qator xavfsizlik protokollarini qo‘llaniladi. Ulardan eng mashhuri bu – “Do‘st yoki dushman ekanlikni aniqlash (Identify friend or foe, IFF)” protokoli. Ushbu protokol askarlar o‘zlari tomondagi askarlarga hujum qilmasliklari uchun ishlab chiqilgan.

Sodda IFF protokoli quyidagi 1.5 – rasmda keltilirlgan. Ushbu protokol Janubiy Afrika havo kuchlari (South African Air Force, SAAF) tomonidan 1970-yillar o‘rtalarida Angolada bo‘lgan urishda foydalanilganligi qayd etilgan. Janubiy Afrika Angola davlati bilan Namibiya uchun urish qilgan. Angola tomoni Sovetlarga tegishli bo‘lgan MIG samalyotlaridan foydalangan bo‘lib, ular Kuba tomonidan uchirilgan (Bu Sovuq Urushning bir qismi bo‘lgan, Urushning dastlabki vaqtlarida Janubiy Afrika tomonini “Angola” uchuvchilari xayratda qoldirgan. Keyinchalik, sun’iy yo‘ldoshdan olingan tasvirlar ular aslida Kuba tomonidan amalga oshirilgani aniqlagan).

IFF protokolining ishlashi 1.5 – rasmda keltirilgan. SAAF tomoni o‘zining aloqa stansiyalari orqali samalyot harakatini aniqlasa, tasodifiy son, *N* ni samalyotgan yuborgan. Barcha SAAFga tegishla samalyotlar bir xil *K* kalit bilan ta’minlangan va ular qabul qilingan tasodifiy sonni shifrlab, *E(N,K),* stansiyaga uzatgan. Bunda vaqt sarfini kamaytirish uchun barcha harakatlar

inson aralashuvisiz avtomatik ravishda amalga oshirilgan. Dushman kuchlari *K* kalitni bilmasliklari sababli, tasodifiy sonni shifrlab yubora olmagan. Ushbu protokol radarni zonadagi samalyotni o‘ziniki yoki dushman ekanligini aniqlash uchun foydalanilgan.

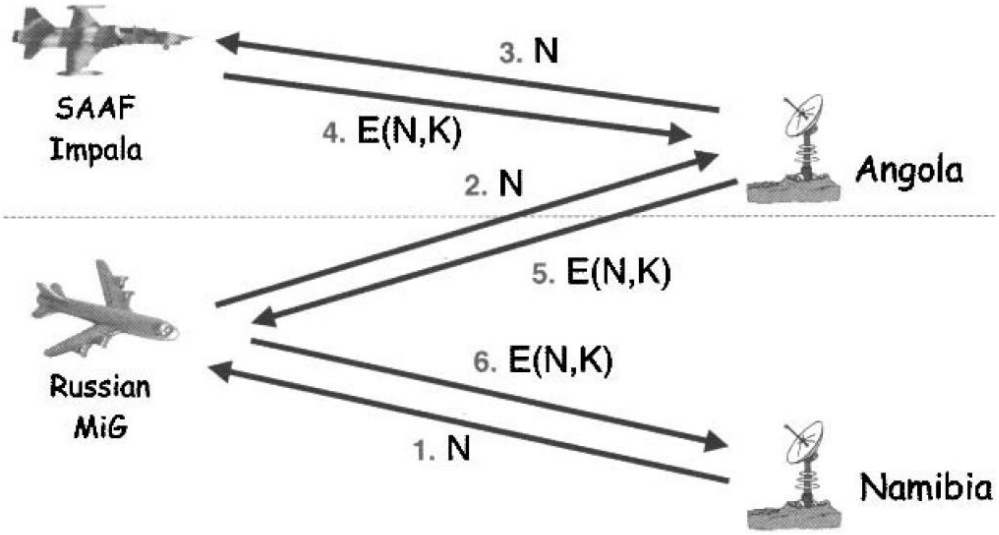


1.5-rasm. IFF protokoli

Ushbu radar stansiyalarining baxtiga qarshi, 1.5 – rasmda ifodalangan IFF tizimiga qarshi aqlli hujum mavjud. Ushbu hujumni ba’zi olimlar MIG-in-the-middle hujumi deb atagan. Ushbu hujumning senariysi 1.6 – rasmda aks ettirilgan. SAAF Impala qiruvchilari Angola missiyasini amalga oshirgan vaqtda, Kubalik uchuvchilar bo‘lgan MIG samalyotlari (SAAFning dushmani) SAAF radarlaridan tashqari bo‘lgan. Impala qiruvchilari Angolada Kuba radar stansiyalari zonasiga kirgan vaqtda, MIGga SAAF radar stansiyalari zonasiga kirish aytilgan. Protokolda keltirilgani kabi, SAAF radar stansiyalari N tasodifiy sonni MIGga yuborgan. MIG samalyotlari dushman hujumiga uchramaslik uchun *E(N,K)* shifrni amalga oshirishi shart bo‘lgan.

MIG buni amalga oshira olmasligi sababli, tasodifiy son *N* ni Angoladagi o‘zining stansiyasiga yuborgan. Bu stansiya esa uni SAAF Impala qiruvchilariga yuborgan. SAAF Impala qiruvchilari buni dushman tomonidan amalga oshirilganini bilmagan va *E(N,K)* amalga oshirgan. Shundan so‘ng, ushbu signalni Angola radarlari MIGga yuboradi va natijada SAAF radarlari javob bilan qanoatlantiriladi. Ushbu jarayonlar barchasi qisqa vaqtga amalga oshirilgani sababli, SAAF radarlari MIGni o‘z “do‘stlari” deb uylagan.

Ushbu protokolning ishlashi aniq keltirilgan bo‘lsada, MIG-in-the-middle hujumini aslida amalga oshganligi haqida ma’lumotlar mavjud emas. Boshqa tomondan ushbu keltirilgan senariy xavfsizlik protokolining buzulishini aniq ifodalab beradi.



1.6 – rasm. MIG-in-the-middle hujumi

Autentifikatsiya protokollari. Alisa va Bob tarmoq orqali bog‘langanda faraz qilaylik Alisa Bobga haqiqatan o‘zi ekanlagini isbotlashi kerak bo‘lsin. Bu o‘rinda Alisa va Bobni inson

yoki mashina bo‘lishini esdan chiqarmaslik zarur. Aniqki, tarmoq senariysi bo‘lganda Alisa va Bobni o‘zgarmas, mashina ekanligi ehtimoli yuqori.

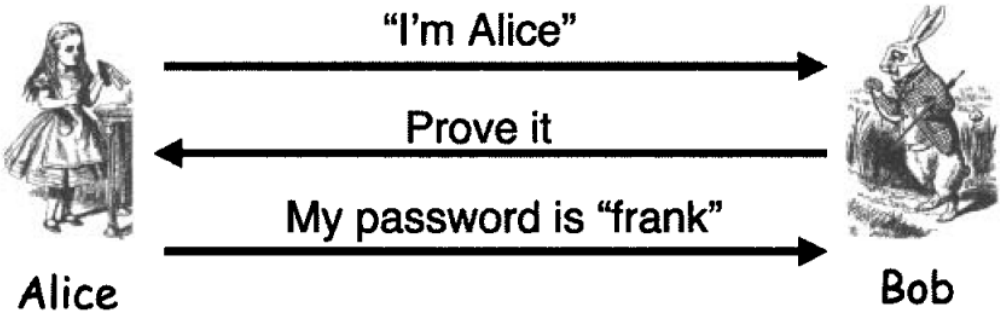
Aksariyat hollarda Alisani Bobga haqiqiyligini isbotlashining o‘zi etarli. Ammo ba’zida har ikkala tomon bir biriga o‘zlarini haqiqiyligini isbotlashi talab etiladi. Bunda odatda Alisa Bobga o‘zini haqiqiyligini tasdiqlashda va Bob Alisaga haqiqiyligini tasdiqlashda ham yagona protokoldan foydalaniladi. Ushbu bobda barcha xavfsizlik protokollari ham har doim xavfsiz emasligini misollar orqali ko‘rib o‘tiladi.

Autentifikatsiyadan tashqari ***sessiya kaliti*** ham ko‘p holda talab etiladi. Sessiya kaliti simmetrik kalit bo‘lib, autentifikatsiyani ta’minlagan holda joriy sessiyada uzatiladigan axborot maxfiyligi yoki butunligin ta’minlashda xizmat kilishi mumkin. Dastlab, faqat autentifikatsiyani amalga oshiruvchi protokollar bilan tanishib chiqiladi.

Xususiy hollarda, xavfsizlik protokollarida boshqa talablar ham bo‘lishi mumkin. Masalan, protokolda ***ochiq kalitli tizimlardan, simmetrik tizimlardan yoki xesh – funksiyalardan foydalanish*** so‘ralishi mumkin. Bundan tashqari, anonimlikni ta’minlovchi, haqiqatan rad etishni ta’minlovchi protol talab etilishi mumkin.

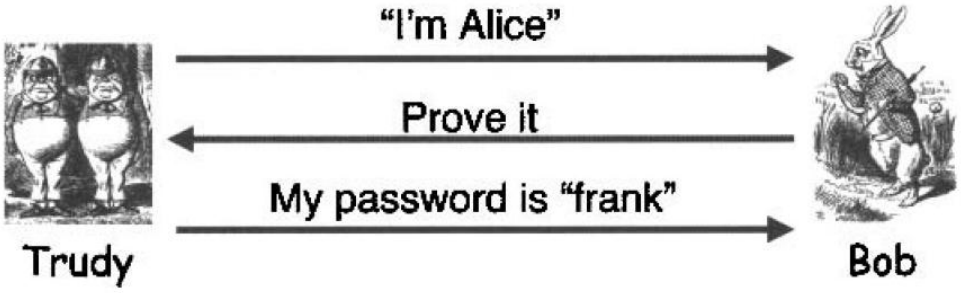
Quyida dastlab yolg‘iz kompyuter tizimida autentifikatsiyalashni ta’minlash bilan bog‘liq muammolar bilan tanishib chiqiladi. Bu o‘rinda ushbu protokollar o‘zida ***xeshlash yoki “salting***” kabi texnologiyalarini qamrab oladi. Biroq, tarmoq orqali autentifikatsiyalashni amalga oshirganda jiddiy muammolar vujudga kelishi mumkin. Tarmoq orqali autentifikatsiyalash amalga oshirilganda buzg‘unchi uchun imkoniyatlar ortadi. Tarmoq bo‘ylab autentifikatsiya xabarlari uzatilganda ***buzg‘unchi tomonidan uni qayta takrorlanishi, qo‘shilishi, o‘chirib tashlash yoki xabarni o‘zgartirish*** holatlari bo‘lishi mumkin.

Birinchi ko‘rinishdagi tarmoq orqali sodda autentifikatsiyalash imkonini beruvchi protokol 1.7-rasmda keltirilgan. Ushbu protokol uchta xabar uzatilishidan iborat bo‘lib, dastlab Alisa Bob bilan aloqani o‘rnatadi va unga o‘z identifikatorini yuboradi. Shundan so‘ng, Bob undan haqiqiyligini ta’minlashni talab etadi va Alisa unga o‘z paroli bilan javob beradi. Parolga asosan Bob Alisani autentifikatsiyadan o‘tkazadi.



1.7-rasm. Sodda autentifikatsiya protokoli

Ushbu protokol ko‘rinishidan sodda ko‘rinsada, unda jiddiy kamchilik mavjud. Agar Buzg‘unchi, Tridi, tarmoqni kuzatish imkoniga ega bo‘lsa, ushbu ma’lumotlardan qayta foydalanishi mumkin. Bunda Tridi ma’lumotlardan keyinchalik foydalanishi mumkin bo‘ladi (1.8-rasm). Ushbu ***takrorlash hujumi*** jiddiy muammo olib keladi.



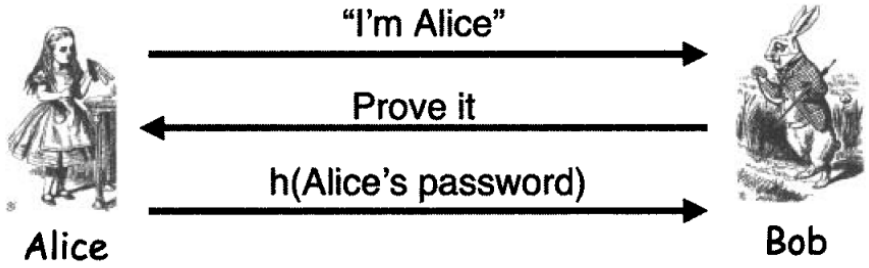
1.8-rasm. Takrorlash hujumi

1.7-rasmda keltirilgan protokol bilan bog‘liq bo‘lgan muammolardan yana biri bu

parolning ochiq yuborilishidir. Agar Tridi tomonidan Alisani parolini kuzatish imkoni bo‘lsa, u holda Tridi uning parolini bilishi mumkin. Ushbu muammo takrorlash hujumiga qaraganda ham jiddiy muammo bo‘lib, Alisa ushbu paroldan ***boshqa qaydlar uchun ham foydalangan taqdirda***, yanada jiddiy muammo kelib chiqadi. Bundan tashqari, yana bir muammo bu ***autentifikatsiyalash uchun Bobni ham Alisaga tegishli parolni bilishi*** talab etilishidir.

Bundan tashqari ushbu protokol ***samarasiz*** ham hisoblanadi. Ya’ni, barcha ma’lumotlarni yagona xabar orqali yuborish mumkin. Shundan qilib, ushbu protokol har tomonlama zaiflikga ega. Va nihoyat, 1.7 – rasmda keltirilgan protokol ***ikki tomonlama autentifikatsiyani*** ta’minlamaydi.

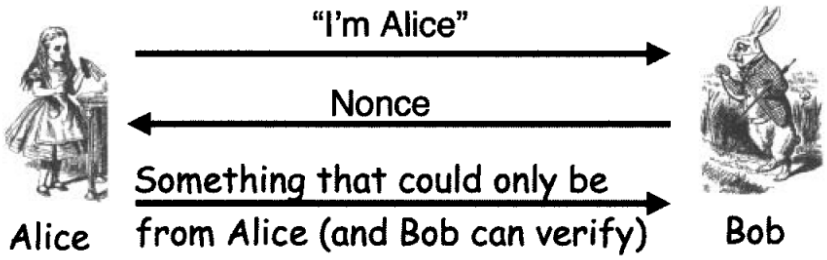
Keyingi ko‘rinishdagi protokol 1.9-rasmda keltirilgan bo‘lib, u yuqorida keltirilgan protokoldagi ***muammolarni bartaraf etgan***. Ushbu yangi protokolda, passiv kuzutuvchi Tridi Alisaga tegishli parolni bila olmaydi va Bob ham ushbu parolni bilishi talab etilmaydi. Bob Alisaga tegishli bo‘lgan parolning xeshini bilishning o‘zi yetarli.



1.9-rasm. Xesh asosida sodda autentifikatsiyalash protokoli

1.9-rasmda keltirilgan protokolda jiddiy muammo bu – ***takrorlash hujumining mavjudligidir***. Ya’ni, Tridi barcha yozishmalarni qayd etadi va ma’lum vaqtdan so‘ng uni takrorlaydi. Bu holda Tridi Alisa sifatida autentifikatsiyadan o‘tadi.

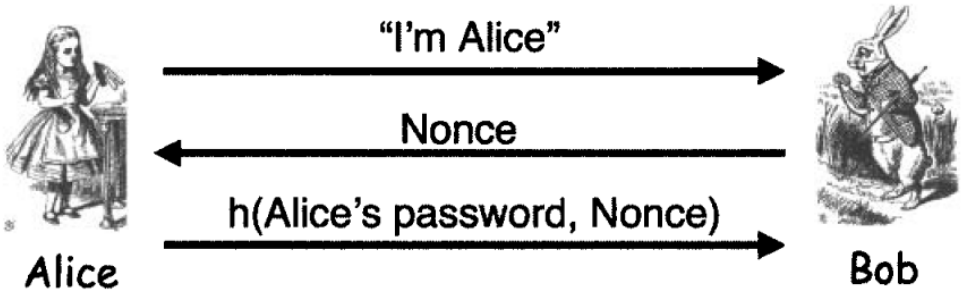
Alisani autentifikatsiyalash uchun Bob ***savol-javob*** mexanizmidan foydalanishga majbur. Ya’ni, Bob Alisaga savolni yuboradi va Alisadan keladigan javob asosida uni tekshiradi. Bu holda takrorlash hujumidan himoyalash uchun, Bob ***“bir martali sonlar (number used once, nonce)”dan*** foydalanishi shart. Ya’ni, Bob har safar unikal savolni yuboradi va unga mos javob hosil qilinadi. Bob qabul qilinayotgan javobni oldin yuborilmaganini aniqlash orqali takrorlash hujumi oldi olinadi yoki boshqa so‘z bilan aytganda ***nonce*** javobni takrorlanmaslikni oldini oladi. Ushbu imkoniyatni taqdim etuvchi protokol 1.10-rasmda aks ettirilgan.



1.10-rasm. Umumiy autentifikatsiya

Dastlab Alisaning paroliga asoslangan holda autentifikatsiyalash protokoli taqdim etiladi. Bunda faraz qilaylik parolni Alisa biladi va Bob ham tekshirish uchun uni bilish imkoniga ega.

Takrorlash hujumiga bardoshli bo‘lgan dastlabki protokol 1.11 – rasmda keltirilgan. Ushbu protokolda, *nonce* Bob tomonidan Alisaga yuboriladi. Alisa o‘zining paroli va *nonce* ni xeshlash orqali javobni amalga oshiradi. Bu holda *nonce* xabar yangiligini va takrorlash hujumidan himoyalashni ta’minlaydi.



1.11-rasm. Savol-javob

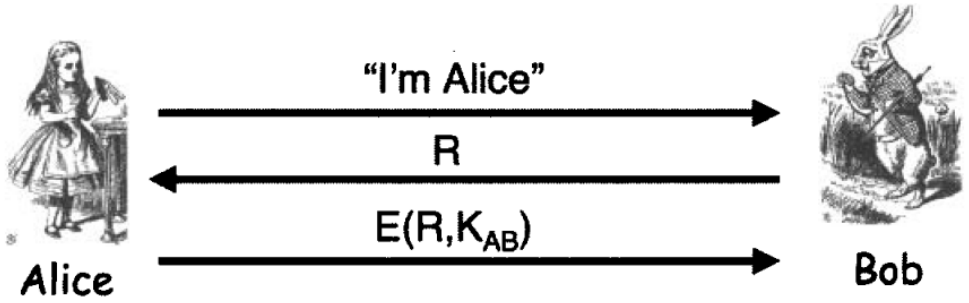
1.11-rasmda keltirilgan protokoldagi yagona muammo bu Bobni Alisaga tegishli parolni bilishi talab etiladi. Bundan tashqari, Alisa va Bob odatda inson ko‘rinishda bo‘lishdan ko‘proq mashina bo‘lishi mumkin va bu holda paroldan foydalanish mantiqsizdir. Bundan tashqari inson parolni esda saqlab yurishi uchun uni odatda murakkab tarzda tanlamaydi. Shuning uchun, Alisa va Bobni mashina bo‘lishi ehtimolini hisobga olib parolni o‘rniga kalitlardan foydalanish mumkin bo‘ladi.

Simmetrik kalitlar yordamida autentifikatsiya

Simmetrik kalitlarga asoslangan autentifikatsiyalashga o‘tishdan oldin ma’lum belgilanishlar keltirib o‘tiladi. Agar *ochiq matn R* bo‘lsa va *kalit K* bo‘lsa, u holda *shifrmatn S=E(P,K)* ga teng bo‘ladi va ochiq matn *P=D(C,K)* ga teng bo‘ladi. Bundan tashqari, ko‘rib o‘tilgan protokollarni tahlil qilganda unda foydalanilgan kriptografik algoritmlarni xavfsiz deb faraz qilinsin.

Faraz qilinsin, Alisa va Bob umumiy simmetrik kalit K_{AB} ga ega. Simmetrik kriptografiya bo‘lgani sababli, kalitni boshqa tomonlar bilmaydi. Alisa o‘zini Bobga autentifikatsiyadan o‘tkazishda ushbu kalitni bilishidan foydalanadi. Bundan tashqari protokol takrorlash hujumidan himoyalashi shart.

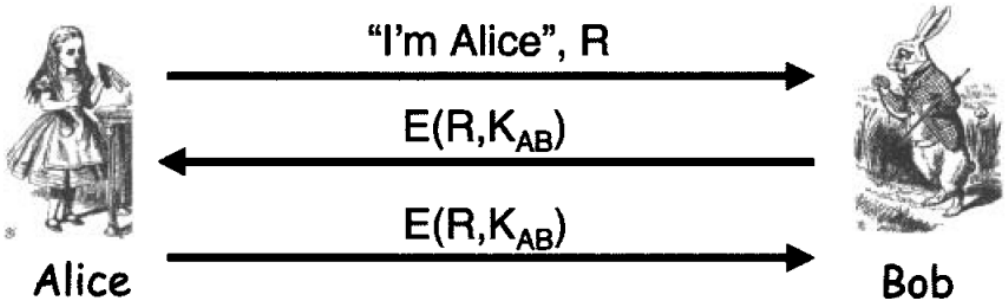
Simmetrik kalitga asoslangan autentifikatsiyalash protokolning birinchi ko‘rinishi 1.12-rasmda keltirilgan. Ushbu protokol yuqorida keltirilgan parolga asoslangan savol-javob protokoligi analog bo‘lib, farqli ravishda *nonce* va *parolni* xeshlash o‘rniga, *tasodifiy qiymat R* ni simmetrik kalit K_{AB} bilan shifrlaydi.



1.12-rasm. Simmetrik kalit asosida autentifikatsiya protokoli

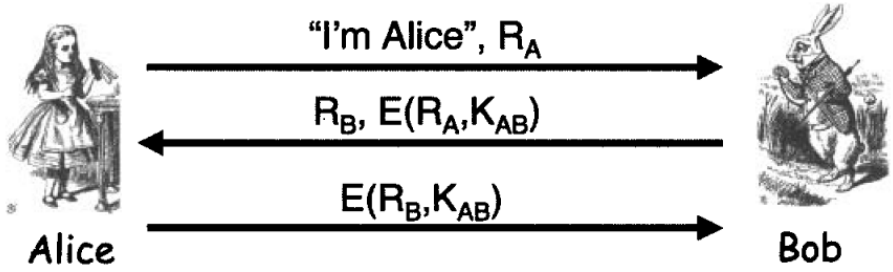
1.12-rasmda keltirilgan protokolda, Alisa *R* qiymatni kalit K_{AB} bilan shifrlay olishi sababli (Tridi esa shifrlay olmaydi), Bob uni autentifikatsiyalay oladi. Buning uchun Bob to‘g‘ri shifrlanganini bilishining o‘zi etarli. Ushbu protokol *tasodifiy qiymat R* dan foydalanilgani bois takrorlash hujumidan himoyalaydi. Biroq ushbu protokol ikki tomonlama autentifikatsiyalashni amalga oshirmaydi. Shuning uchun keyingi topshirish sifatida ikki tomonlama autentifikatsiyalash vazifasi olinadi.

Ikki tomonlama autentifikatsiyalash imkoniyatini beruvchi protokolning dastlabki ko‘rinishi 1.13 – rasmda aks ettirilgan. Bu protokol samarali va simmetrik kalitdan foydalangan bo‘lsada, unda ham zaiflik mavjud. Uchinchi xabar ikkinchi xabar bilan bir xil bo‘lib, u yuboruvchi haqida xech narsani isbotlay olmaydi, ya’ni, yuboruvchi Alisami yoki Tridimi ?



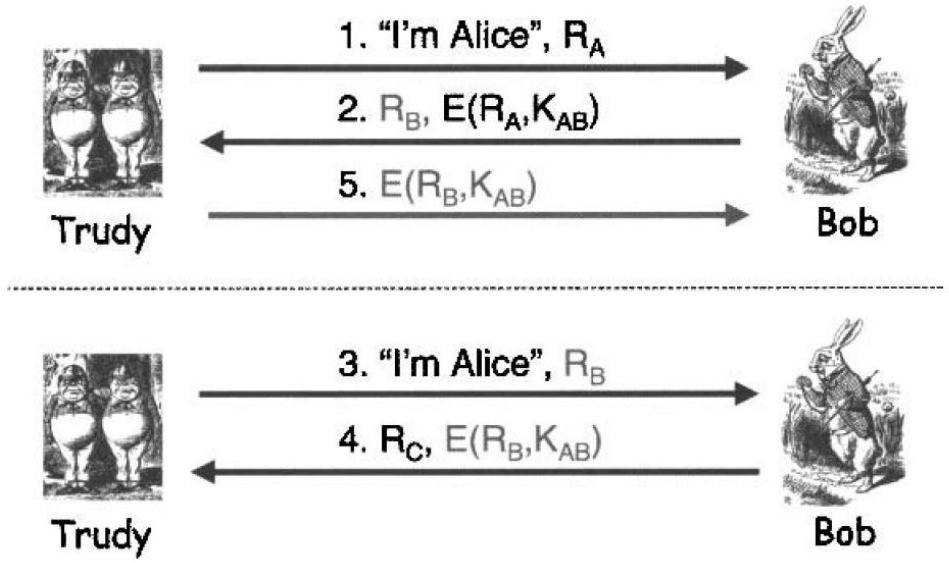
1.13-rasm. Ikki tomonlama autentifikatsiyami ?

Ikki tomonlama autentifikatsiyalashning xavfsiz usuli bu 1.12-rasmda keltirilgan protokol bo‘lib, faqat jarayonni ikki marta takrorlash talab etiladi, ya’ni, bir marta Alisani autentifikatsiyalash uchun keyingi marta Bobni autentifikatsiyalash uchun. Ushbu protokolning umumiy ko‘rinishi 1.14-rasmda aks ettirilgan. Bunda bir nechta xabarlarni birlashtirish orqali samaradorlik oshirilgan.



1.14-rasm. Xavfsiz ikki tomonlama autentifikatsiya ?

Ushbu protokol ko‘rinishidan xayratlanarli bo‘lsada, MiG-in-the-middle hujumiga o‘xshash hujumga bardoshsizdir. Ushbu hujum 11-rasmd aks ettirilgan. Bunda Tridi Bob bilan aloqani Alisa nomidan o‘rnatadi va R_A ni Bobga yuboradi. Protokolga asosan Bob uni shifrlaydi va uni R_B ga qo‘shib, Tridiga yuboradi. Bu holda Tridi kalitni bilmasligi sababli, uni shifrlay olmaydi. Biroq Tridi malakali bo‘lganligi sababli, Bob bilan yangi aloqani o‘rnatadi va unga yana Alisa ekanligini aytadi. Shuning bilan birga R_B ni o‘zini tasodifiy qiymati sifatida Bobga yuboradi. Protokolga asosan Bob $E(R_B,K_{AB})$ ni shifrlaydi va Tridiga yuboradi. Tridi esa ushbu ma’lumotdan birinchi bog‘lanishni tugallash uchun foydlanadi. Tridi ikkinchi ulanishdan chiqib ketadi va birinchi ulanish asosida Alisa sifatida autentifikatsiyadan o‘tadi.

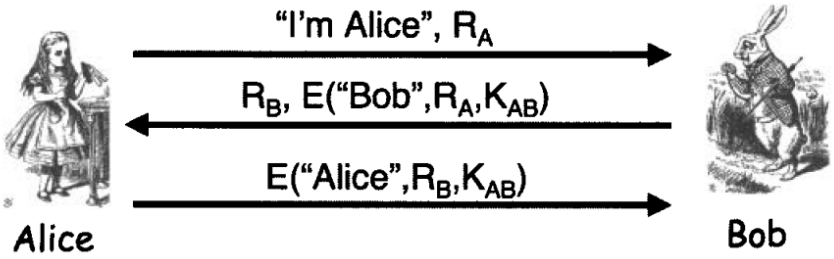


1.15-rasm. Tridining hujumi

Umumiy holda xulosa shundan iboratki, ***bir tomonlama autentifikatsiyalashda***

foydalanilgan protokol ikki tomonlama autentifikatsiyalash uchun o‘rinli emas.

1.16-rasmda 1.14-rasmda keltirilgan xavfsiz bo‘lmagan ikki tomonlama autentifikatsiyalash protokolini kichik o‘zgarishga uchragan ko‘rinishi keltirilgan. Bunga asosan, foydalanuvchining identifikatori *nonce*ga qo‘shilib, shifrlanadi. Ushbu kichik o‘zgarish 1.15-rasmda keltirilgan tahdidga bardoshli bo‘lishni ta’minlaydi.



1.16-rasm. Bardoshli ikki tomonlama autentifikatsiyalash protokoli

Yuqorida keltirilgan protokollardan shunday xulosa chiqadiki, protokolni ikki tomonda ham bir xil narsani bajarishi yaxshi g‘oya emas. Keyingi darslarda protokoldagi kichik o‘zgarishni uning xavfsizligida katta ta’sir qilishini ko‘rib chiqiladi.

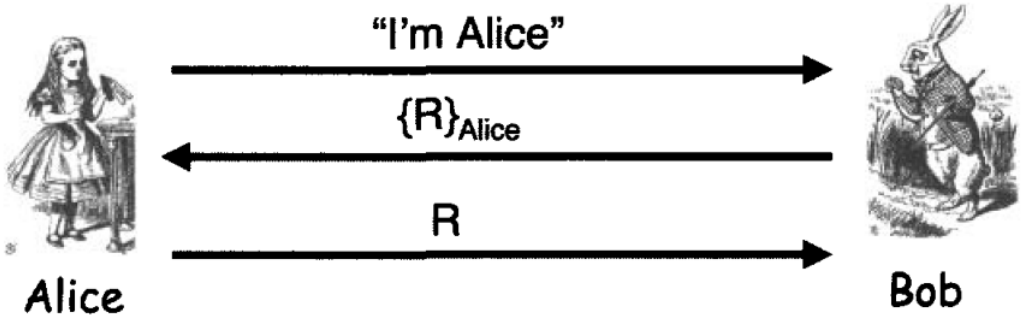
Ochiq kalitdan foydalangan holda autentifikatsiya

Bundan oldingi bo‘limda simmetrik kalitga asoslangan xavfsiz ikki tomonlama autentifikatsiyalash protokoli ko‘rib o‘tgan edik. Ushbu bo‘limda bu vazifani ochiq kalitni kriptografik tizimlar yordamida amalga oshirish masalasi bilan tanishib o‘tiladi. Bundan oldin quyidagi belgilanishlar olinadi. Alisaga tegishli bo‘lgan ochiq kalit bilan ma’lumot M ni shifrlash $C=\{M\}_{Alisa}$ kabi belgilansa, shunga asosan ochiq matnni $M=[C]_{Alisa}$ kabi belgilanadi. Imzolashda maxfiy kalitdan foydalaniladi va shifrlash/deshiarlash jarayonlari imzolash/imzoni tekshirish jarayonlari teskari bo‘lgani uchun

$$[\{M\}_{Alisa}]_{Alisa} = M \text{ va } \{[M]_{Alisa}\}_{Alisa} = M.$$

Ochiq kalitli kriptografik tizimlarda ochiq kalit bilan ixtiyoriy kishi ixtiyoriy amalni bajarishi mumkin va bunda maxfiy kalitdan faqat Alisa foydalanishi mumkin bo‘ladi.

Ochiq kalitli kriptografiyaga asoslangan autentifikatsiyalash protokolining birinchi ko‘rinishi 1.17-rasmda aks ettirilgan. Ushbu protokol Alisa maxfiy kalit bilan shifrmatnni deshifrlay olgani va R ni uchinchi xabarda yuborgani uchun Bobga Alisani autentifikatsiyalash imkonini beradi. Bunda R tasodifiy qiymat Bob tomonidan generatsiya qilingani bois, takrorlash hujumida himoyalangan. Shuning uchun, Tridi birinchi ulanishdan qayd etgan ma’lumotlarni keyingi ulanishlarda foydalana olmaydi.

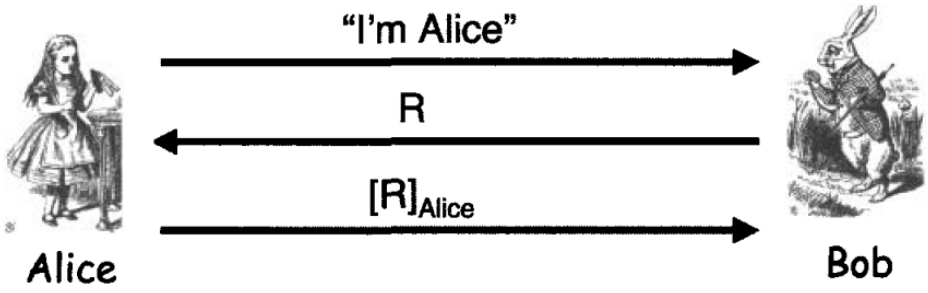


1.17-rasm. Ochiq kalit asosida autentifikatsiyalash

Biroq, Alisa autentifikatsiyada foydalangan kalitni shifrlashda ham foydalansa, 13-rasmda keltirilgan protokolda jiddiy muammo bo‘lishi aniq. Faraz qilaylik, Tridiga oldinroq tutib olingan Alisani ochiq kaliti bilan shifrlangan xabar bor, $C=\{M\}_{Alisa}$. Keyin, Tridi o‘zini Bob kabi tutib, *C* ni 2 xabarda Alisaga yuboradi va Alisa uni deshifrlab Tridiga qaytarib yuboradi. Tridi nuqtayi nazaridan, bundan yaxshiroq biror narsaga erishish imkonsiz. Buning tarbiyali tomoni shundaka, yagona kalit juftini ham ma’lumotni shifrlashda va imzolashda foydalanmaslik zarur.

1.17-rasmda keltirilgan protokolda ochiq kalitni shifrlashdan foydalanilgan. Ushbu protokolni imzolash orqali ham amalga oshirish mumkinmi ? Ha. Uning umumiy ko‘rinishi 1.18-

rasmda keltirilgan.



1.18-rasm. Raqamli imzo orqali autentifikatsiyalash

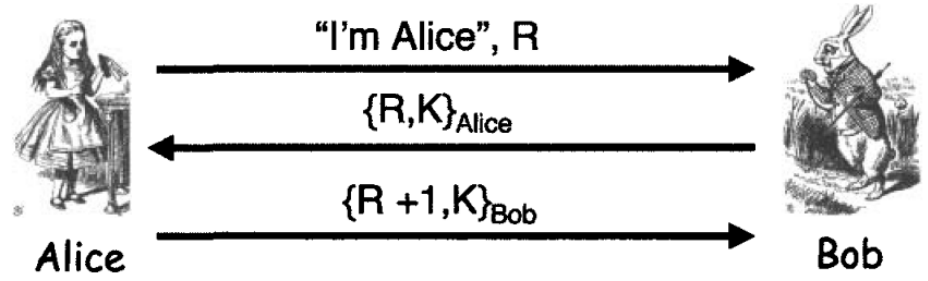
Ushbu protokol 1.17-rasmda keltirilgan protokolda mavjud bo‘lgan zaiflikga ega. Ushbu protokolda, agar Tridi o‘zini Bob kabi tutsa, unda Alisa tomonidan imzolangan biror ma’lumotga ega bo‘ladi. Ushbu muammoning yechimi bu imzolashda va shifrlashda turli kalit juftlaridan foydalanishdir. Va nihoyat, 1.17 va 1.18 – rasmlarda keltirilgan protokollar Alisa uchun bir xil. Ya’ni, har ikkala holda ham u maxfiy kalitidan foydalangan holda amalni bajaradi.

Sessiya kalitlari

Autentifikatsiyalash bilan birga odatda sessiya kaliti ham talab etiladi. Hattoki autentifikatsiyalashda simmetrik kalitlardan foydalanilgan taqdirda ham, har bir ulanishda ma’lumotlarni shifrlashda foydalanilgan kalitdan boshqa sessiya kalitlaridan foydalanish kerak. Sessiya kalitining maqsadi ixtiyoriy yagona kalit bilan shifrlanuvchi ma’lumotlar sonini kamaytirishdan iborat va u bir sessiya kaliti oshkor bo‘lganda ham bo‘lishi mumkin bo‘lgan zararni kamaytirishga xizmat qiladi. Sessiya kalitini xabarlar maxfiyligi yoki butunligini ta’minlashda (yoki har ikkalasini) foydalanish mumkin.

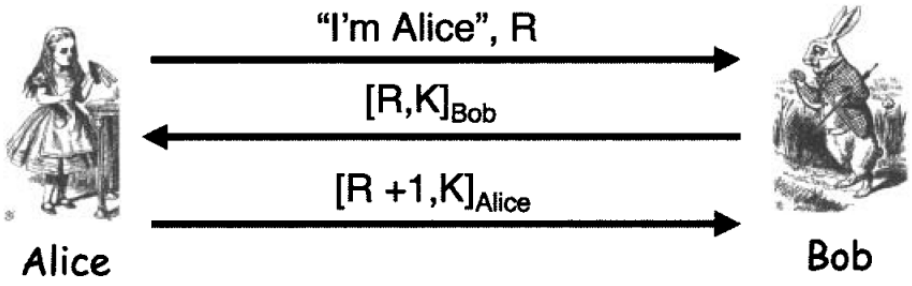
Shuning uchun sessiya kalitlarini o‘rnatish autentifikatsiya protollarining bir qismi sifatida bo‘lishi kerak. Ya’ni, autentifikatsiya jarayoni tugagandan so‘ng, tomonlan taqsimlangan simmetrik sessiya kaliti bilan ta’minlanishi shart. Shuning uchun autentifikatsiya protokollarini tahlil qilganda, e’tiborni nafaqat autentifikatsiyalashga balki sessiya kalitiga ham qaratish zarur.

Shundan kelib chiqqan holda keyingi ishlarda sessiya kalitlarini taqsimlash imkoniyatini beruvchi autentifikatsiya protokollarini ishlab chiqishga e’tibor beriladi. Ochiq kalitli kriptografiyaga asoslangan sessiya kalitlarini taqsimlash imkonini beruvchi autentifikatsiya protokoli 1.19-rasmda keltirilgan.



1.19-rasm. Autentifikatsiya va sessiya kaliti

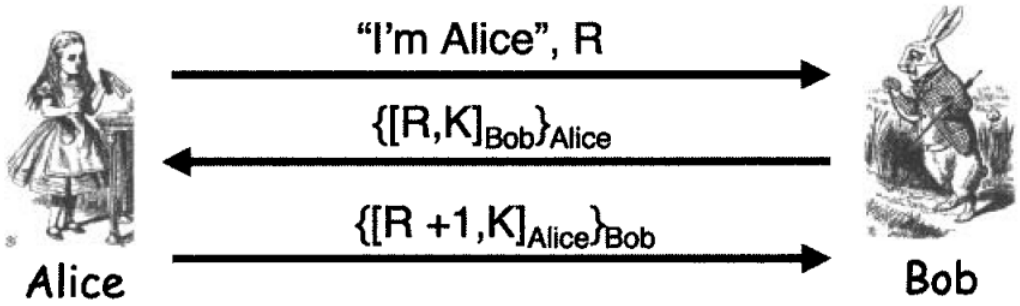
Ushbu protokoda mavjud bo‘lgan yagona muammo bu ikki tomonlama autentifikatsiyalashni amalga oshirilmaganligi, faqat Alisa autentifikatsiyadan o‘tkazilgan. Ammo ushbu muammoga to‘xtalishdan oldin, 1.19-rasmda keltirilgan protokolni shifrlash amalini o‘rniga imzolash amaliga o‘zgartiramiz va natijani tekshirib ko‘ramiz. Ushbu senariy 1.20-rasmda aks ettirilgan.



1.20-rasm. Imzoga asoslangan autentifikatsiya va sessiya kaliti

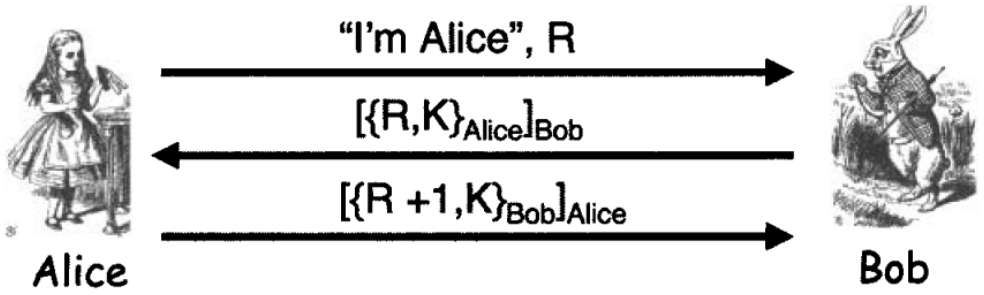
Biroq, 1.20-rasmda keltirilgan protokolda jiddiy muammo mavjud. Sessiya kaliti imzolangani bois, Bobni ochiq kalitini biluvchi ixtiyoriy odam sessiya kalitini qo‘lga kiritishi mumkin. Sessiya kalitini barcha uchun ochiq bo‘lishi jiddiy muammo sanaladi. Biroq ushbu protokol 1.19-rasmda keltirilgan protokolda amalga oshirilmagan ikki tomonlama autentifikatsiyani amalga oshirgan. 1.19 va 1.20-rasmlarda keltirilgan ikki protokoldan foydalanib, ham ikki tomonlama autentifikatsiyani va sessiya kalitini xavfsiz taqsimlash protokolini amalga oshirish mumkinmi? Ha. Buning ikkita yo‘lini quyida keltirib o‘tiladi.

Faraz qilaylik, protokollarda foydalanilgan imzolash yoki shifrlash amallari o‘rniga, imzolash va shifrlash amallarini ishlatiladi. 1.21-rasmda imzolash va shifrlashga asoslangan protokol keltirilgan. Ushbu protokol ham xavfsiz ikki tomonlama autentifikatsiyani ham sessiya kalitlarini almashinishni ta’minlaydi.



1.21-rasm. Ikki tomonlama autentifikatsiya va sessiya kaliti

Ushbu protokolni shifrlash va imzolash orqali ham amalga oshirsa bo‘ladi va u 1.22-rasmda aks ettirilgan.



1.22-rasm. Shifrlash va Imzolashga asoslangan ikki tomonlama autentifikatsiya

Ushbu protokolda $\{R, K\}_{Alisa}$ va $\{R+1, K\}_{Bob}$ qiymatlar barchaga ma’lum bo‘ladi. Ya’ni, Bob va Alisani ochiq kalitlarini bilgan ixtiyoriy inson ushbu ma’lumotni tutib olishi yoki o‘zgartirish mumkin. Bu muammo esa imzolash va shifrlashga asoslangan protokolda mavjud emas. Har ikkala holda ham agar buzg‘unchi ochiq kalitli shifrlash algoritmini buzsa, jiddiy muammo mavjud bo‘ladi. Biroq, protokollarni tahlil qilishda foydalanilgan kriptografik algoritmlar bardoshli deb qaralgan.

Nazorat savollari

1. Nosimmetrik shifrning simmetrik shifrdan farqi nima?
2. Elektron raqamli imzo, xesh-funksiya va autentifikatsiyadan nima maqsadda foydalaniladi?
3. Protokol nima? U qanday xususiyatlarga ega?
4. Protokolning vazifasi nimadan iborat?
5. Qanday protokol turlarini bilasiz? Ularga misollar keltiring.
6. Kriptografik protokoli ishtirokchilari qanday sinflarga bo‘linadi.
7. Kriptografik protokol necha guruhga bo‘linadi? Ularning ta’rifi va belgilanishi keltirib o‘ting.
8. KPning eng dolzarb xossalarini keltiring.
9. KPning xossalaridan qaysi biri muhim ahamiyatga ega va nima uchun?
10. KPlar qanday funksiyalarni bajaradi? KPlar funksiyasiga ko‘ra necha turga bo‘linadi?
11. Shifrlash protokoli bilan ERI protokoli jarayonlarining farqi nimada?
12. Identifikatsiya/autentifikatsiya protokoliga izoh bering.
13. Protokol xavfsizligiga qanday asosiy talablar qo‘yiladi?
14. Xavfsizlik talablariga javob beradigan protokollarni yaratishda qanday yondashuvlardan foydalaniladi?
15. Nollik bilimga asoslangan autentifikatsiyalash.
16. O‘rtaga turgan odam hujumini tushunting.
17. Simmetrik kriptotizimlarga asoslangan autentifikatsiyalash protokollari.
18. Ochiq kalitli kriptografik tizimlarga asoslangan autentifikatsiyalash protokollari.

2-ma’ruza. Kriptografik kalit almashish protokollari

Reja:

1. **Simmetrik kriptotizimlarga asoslangan kalitlarni taqsimlash protokollari**
2. **Asimmetrik kalitli algoritmlar yordamida mahfiy aloqani tashkil qilish protokollari**

Ushbu ma’ruzada simmetrik shifrlash algoritmi yordamida generatsiya qilingan kalitni almashish protokollari ko‘rib chiqiladi. Bu protokollarda axborot almashinuvi subyektlari bo‘lgan *A* va *B* foydalanuvchilar umumiy k_{AB}- kalitga ega deb qabul qilinadi. Bu protokollar, uchinchi ishonchli tomonning ishtirok etishi yoki etmasligiga bog‘liq ravishda ikki turga bo‘linadi. Avvalo uchinchi ishonchli tomon ishtirok etmagan protokollarni ko‘rib o‘tiladi. Buning uchun quyidagi belgilashlar kiritiladi:

Ye – shifrlash algoritmi;

t_A – vaqt belgisi;

r_A – *A* -foydalanuvchining tasodifiy soni;

n_A – *A* -foydalanuvchining generatsiya qilish tartib raqami;

V – B -foydalanuvchining identifikatsion raqami;

k_{AB} – ikkala tomonga ham ma’lum bo‘lgan kalit.

Simmetrik kalitli kriptotizimda foydalanuvchilardan tashqari kalitlarni tarqatuvchi tomon, ya’ni kalitlarni tarqatish markazi ham ishtrok etadi. Simmetrik kriptotizim yordamida kalitlarni almashish protokoli quyidagicha amalga oshiriladi:

1. A - foydalanuvchi V - foydalanuvchi bilan aloqa o‘rnatish uchun kalit tarqatuvchiga murojaat qiladi va seans kalitini so‘raydi.

2. Kalit tarqatuvchi seans kalitni generatsiya qiladi va bu kalitini ikki nusxada shifrlab, A - foydalanuvchiga uzatadi.

3. A - foydalanuvchi o‘ziga tegishli shifrlangan seans kalitini deshifrlaydi.

4. A - foydalanuvchi shifrlangan seans kalitining ikkinchi nusxasini V foydalanuvchiga uzatadi.

5. V - foydalanuvchi o‘zining shifrlangan kalitini deshifrlaydi.

6. A va V - foydalanuvchilar mahfiy aloqa uchun yuqorida hosil qilingan seans kalitidan foydalanadilar.

Bu protokolda seans kalitlar tarqatuvchini ishonchli tomon deb qabul qiladilar. Agar kriptoanalitik aktiv hujum yordamida yoki boshqa qandaydir usul bilan seans kalitlarini qo‘lga kiritsa, u holda kriptoanalitik aloqa tarmog‘iga ulanib, tarmoqdagi barcha almashinuvchi mahfiy ma’lumotlarni kuzatish yoki eshitish imkoniyatiga ega bo‘ladi.

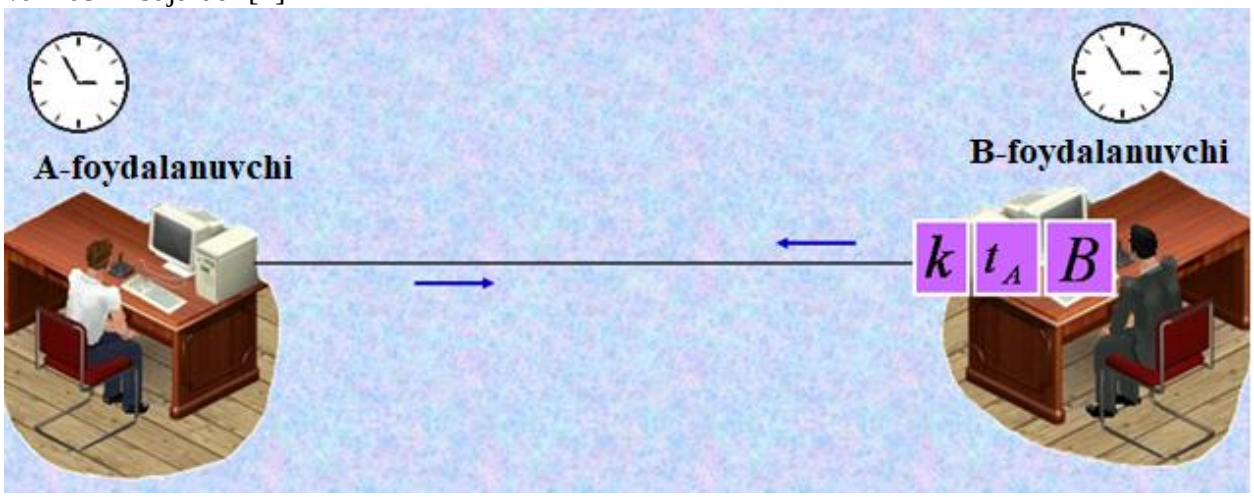
Yuqorida bayon qilingan tizimning yana bir kamchiligi shundaki, har bir kalit almashishda qatnashuvchi uchinchi tomon, ya’ni kalitlarni tarqatish markazi, mazkur tizimning nozik nuqtasi hisoblanadi. Agar unda biror kamchilik kuzatilsa, butun tizimga ta’sir etadi. Quyida shu kabi bir nechta protokollar haqida to‘xtalib o‘tiladi.

1 – protokol

Simmetrik shifrlash algoritmi yordamida generatsiya qilingan kalitni uzatish protokolining sodda ko‘rinishi – seans kalitini bir raundda uzatish. Butun protokol yagona ma’lumotdan tashkil topgan:

$$\text{A} \rightarrow \text{V}: \; E_{k_{AB}}(r_A, t_A, B).$$

V - foydalanuvchi umumiy kalit yordamida bu ma’lumotni deshifrlaydi. Bu holda r_A - seans kalit vazifasini bajaradi [2]



Xulosa

Agar ko‘rib chiqilgan protokolda:

1. Baqt belgisi uzatilmasa, kriptoanalitik aynan shu ma’lumotni qayta uzatishi mumkin.

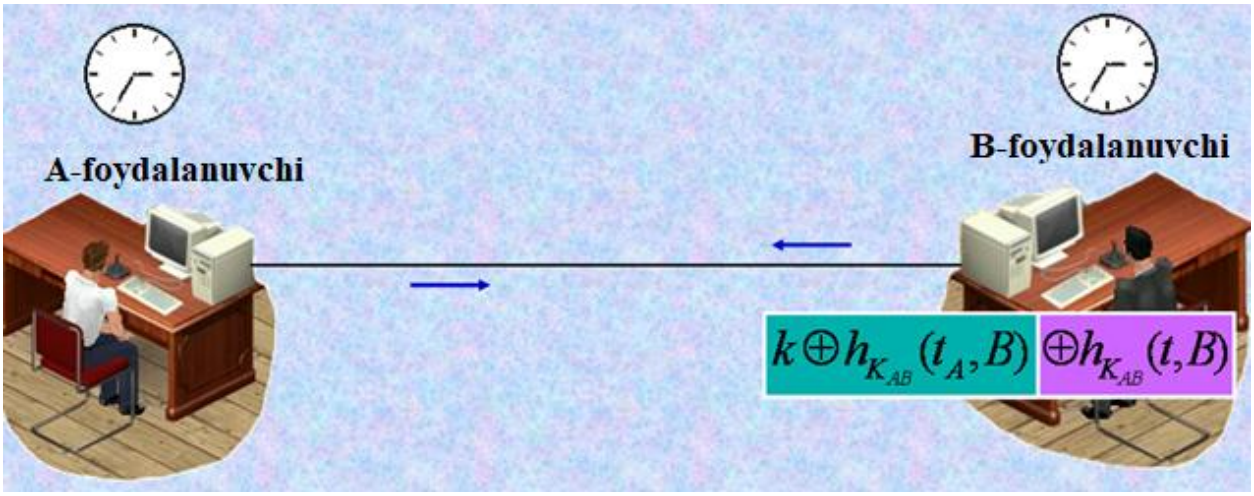
2. V - foydalanuvchining identifikatsion raqami ko‘rsatilmasa, kriptoanalitik bu ma’lumotni *A*-foydalanuvchining o‘ziga uzatishi mumkin va natijada *A* -foydalanuvchi ma’lumot V - foydalanuvchidan kelgan yoki kelmaganligini aniqlay olmaydi.

3. Seans kalit $f(r_A, r_B)$ funksiya yordamida hisoblab topilishi mumkin. Agar f funksiya sifatida bir tomonlama funksiyadan foydalanilsa, tomonlarning hech biri natijaviy kalitni nazorat qila olmaydi.

Yuqorida keltirilgan protokolda shifrlash algoritmi o‘rniga kalit yordamidagi xesh - funksiyadan foydalanish mumkin:

$$\text{A} \rightarrow \text{V}: \; < t_A, r_A \oplus h_{k_{AB}}(t_A, B) > .$$

V -foydalanuvchi ma’lumotni qabul qiladi. U ham kalit orqali xeshlash funksiyasini biladi. Qabul qilgan ma’lumotidan vaqt belgisini ajratib oladi. Uning keyingi vazifasi vaqt belgisi va o‘zining identifikatsiya raqamini birlashtirib kalitli xesh - funksiya yordamida xeshlashni amalga oshirish. Chiqqan $h_{k_{AB}}(t_A, B)$ natijani qolgan $r_A \oplus h_{k_{AB}}(t_A, B)$ ma’lumotga XOR amali bo‘yicha qo‘shiladi. Natijada r_A - seans kalit hosil bo‘ladi.



Agar tizim umumiy sinxron vaqtga ega bo‘lmasa, lekin kalitning yangiligiga ishonch hosil qilish talab qilinsa, u holda vaqt belgisini tartib raqam bilan almashtirish mumkin. U holda protokol quyidagi ko‘rinishga keladi:

2 – protokol

V - foydalanuvchi o‘zining n_B -tasodifiy sonini hosil qilib uni A-foydalanuvchiga uzatadi:

$$\mathrm{V} \to \mathrm{A}: n_B$$

A - foydalanuvchi bu tasodifiy sonni qabul qilib, unga o‘zi hosil qilgan seans kalitini va V - foydalanuvchining identifikatsiya raqamini birlashtirib ikkala foydalanuvchi uchun umumiy bo‘lgan kalit yordamida shifrlaydi hamda V - foydalanuvchiga uzatadi:

$$\mathrm{A} \to \mathrm{V}: E_{k_{AB}}(r_A, n_B, B).$$

V - foydalanuvchi n_B va V ni tekshirib, r_A -seans kalitining to‘g‘ri ekanligiga ishonch hosil qiladi. Xesh - funksiyadan foydalanilsa protokolning ko‘rinishi quyidagicha bo‘ladi:

$$\mathrm{V} \to \mathrm{A}: n_B$$
$$\mathrm{A} \to \mathrm{V}: r_A \oplus h_{k_{AB}}(n_B, B).$$

Ushbu protokolni shunday o‘zgartirish mumkinki, natijada $k = r_A$-seans kalitini bir tomon emas, balki ikkala tomon birgalikda generatsiya qiladilar [2].

A va V -foydalanuvchilar r_A va r_B -sonlaridan boshqa tasodifiy n_A va n_B - sonlarni generatsiya qiladilar. Bu yerda r_A va r_B - sonlari kalit materiallari sifatida foydalaniladi, n_A va n_B -sonlari esa kalitning yangi kalit ekanligini ta’minlaydi. U holda protokol quyidagicha amalga oshiriladi:

1) yuqorida keltirilgan protokol kabi V -foydalanuvchi o‘zining n_B - tasodifiy sonini A - foydalanuvchiga uzatadi:

$$\mathrm{V} \to \mathrm{A}: n_B \ ;$$

2) A -foydalanuvchi bu tasodifiy sonni qabul qiladi. O‘zaro autentifikatsiyani ta’minlash hamda seans kalitni birgalikda xosil qilish uchun quyidagi ma’lumotni V -foydalanuvchiga uzatadi:

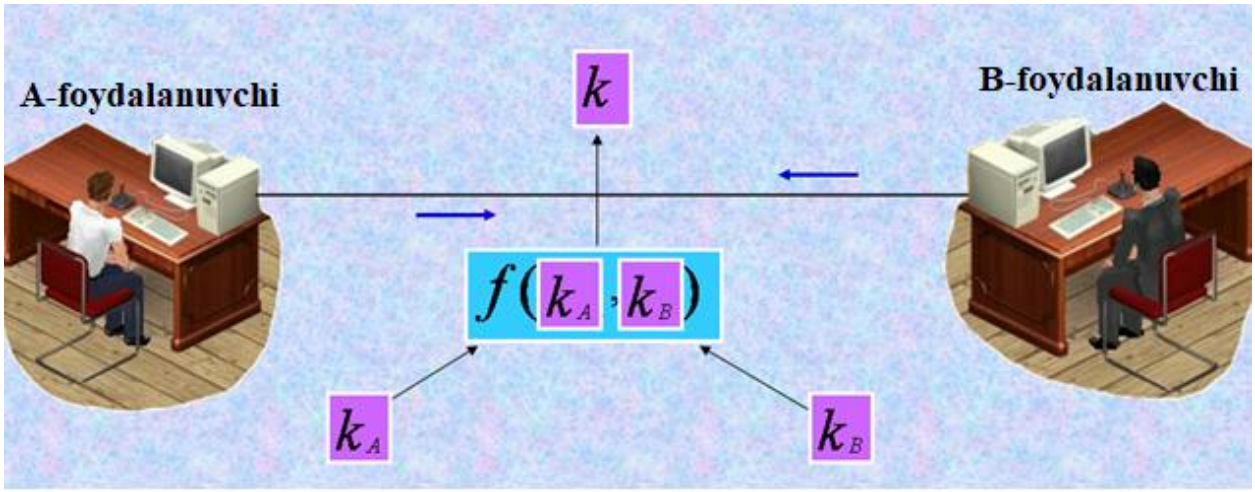
$$\mathrm{A} \to \mathrm{V}: E_{k_{AB}}(r_A, n_A, n_B, B);$$

3) V -foydalanuvchi ma’lumotni deshifrlab, n_B -tasodifiy sonni tekshiradi. Natija to‘g‘ri

bo‘lsa, A -foydalanuvchiga r_B, n_B, n_A, A - ni umumiy kalit bilan shifrlab uzatadi:

$$\text{V} \to \text{A}: E_{k_{AB}}(r_B, n_B, n_A, A);$$

4) Natijada har bir tomon umumiy kalitni oldindan kelishib olingan biror funksiya yordamida $k = f(r_A, r_B)$ qonuniyat bilan hisoblab topishi mumkin.



Quyida esa **Shamir protokoli** deb ataluvchi (kalitsiz) umumiy mahfiy ma’lumotdan foydalanmagan holda kalitni uzatish protokolini ko‘rib chiqiladi. Bu protokol qadamlariga muvofiq kalitning mahfiylik masalasi ta’minlanadi.

Shunday shifrlash va deshifrlash o‘zgartirishlari mavjudki barcha x -ma’lumotlar, k_1 va k_2 -kalitlar uchun quyidagi shart bajariladi:

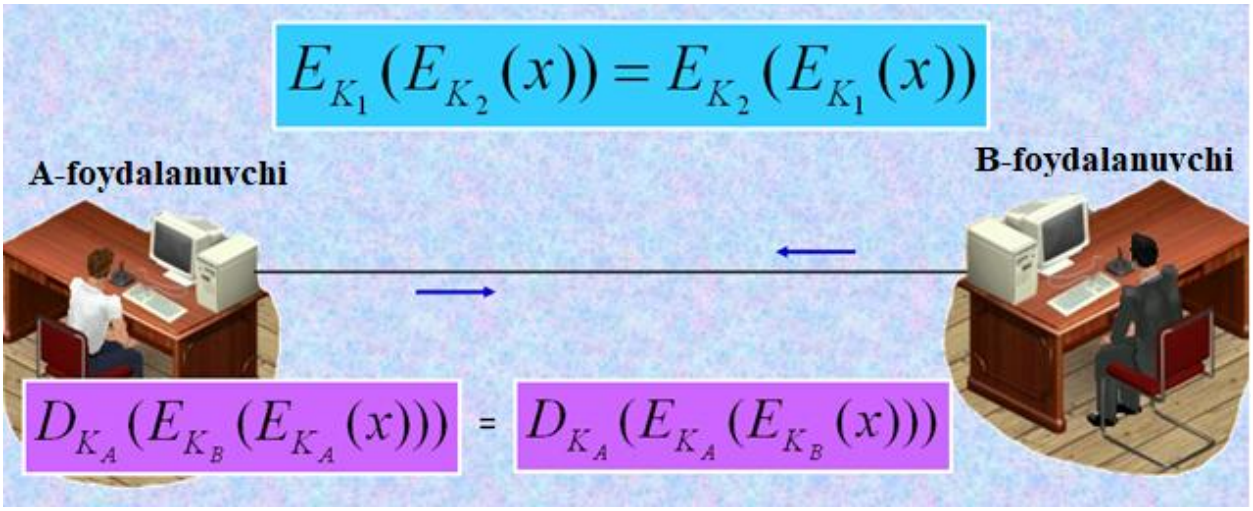
$$E_{k_1}(E_{k_2}(x)) = E_{k_2}(E_{k_1}(x)).$$

U holda A va V -foydalanuvchilar k -seans kalitini uzatuvchi quyidagi 3 - bosqichli protkoldan foydalanishlari mumkin:

(1) $\text{A} \to \text{V}: E_{k_A}(k)$,

(2) $\text{V} \to \text{A}: E_{k_B}(E_{k_A}(k))$,

(3) $\text{A} \to \text{V}: D_{k_A}(E_{k_B}(E_{k_A}(k)))$.



Xususan, Shamir protokolida modul bo‘yicha darajaga ko‘tarish amalidan foydalanish taklif etilgan, ya’ni $E_{k_A}(k) = k^{k_A} \bmod p$. Shunday qilib, bu protokolning kriptobardoshligi diskret logarifmlash masalasining murakkabligiga asoslangan. Shamir protokolining kamchiligi shundaki, bu protokolda autentifikatsiya masalasi hal etilmagan.

Nidxem-Shryoder protokoli

Rojer Nidxem va Mixael Shryoderlar tomonidan yaratilgan bu protokolda arbitr va

simmetrik kriptotizimdan foydalaniladi:

1. A - foydalanuvchi ishonchli tomonga (W) o‘zining ismini, V-foydalanuvchining ismini va o‘zining tasodifiy sonini uzatadi.

$$\mathrm{A} \to \mathrm{W}: A, B, R_A \quad .$$

2. 3 - ishonchli tomon seans kalitni generatsiya qiladi. Bu seans kalitni va A - foydalanuvchining ismini V - foydalanuvchi bilan umumiy bo‘lgan kalit orqali shifrlaydi. So‘ngra A -foydalanuvchi va o‘zi uchun umumiy bo‘lgan kalit yordamida A - foydalanuvchining tasodifiy soni, V- foydalanuvchining ismi, kalit va shifrmatnni shifrlaydi. Nihoyat u shifrlangan ma’lumotni A -foydalanuvchiga uzatadi:

$$\mathrm{W} \to \mathrm{V}: E_A(R_A, B, k, E_B(k, A)) \ .$$

3. A - foydalanuvchi ma’lumotni deshifrlab, k -kalitni oladi. U R_A va 1 - bosqichda uzatilgan R_A ni solishtiradi. So‘ngra A - foydalanuvchi ishonchli tomon shifrlagan ma’lumotni V -foydalanuvchiga uzatadi:

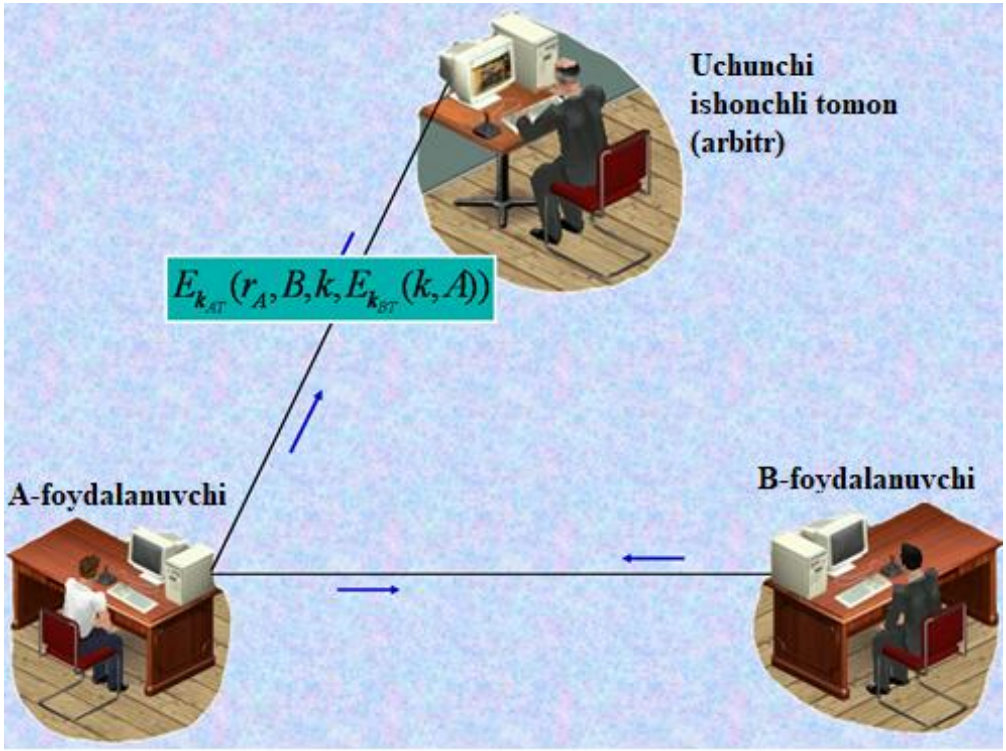
$$\mathrm{A} \to \mathrm{V}: E_B(k, A) \ .$$

4. V - foydalanuvchi bu ma’lumotni deshifrlaydi va k - kalitni oladi. So‘ngra u tasodifiy R_B - sonini generatsiya qiladi. Bu tasodifiy sonni k -kalit yordamida shifrlaydi va A - foydalanuvchiga uzatadi:

$$\mathrm{V} \to \mathrm{A}: E_k(R_B) \ .$$

5. A - foydalanuvchi k - kalit yordamida ma’lumotni deshifrlaydi. A- foydalanuvchi tasodifiy $R_B - 1$ - sonini generatsiya qiladi. Bu sonni k -kalit yordamida shifrlab qayta V - foydalanuvchiga uzatadi:

$$\mathrm{A} \to \mathrm{V}: E_k(R_B - 1) \ .$$



6. V - foydalanuvchi ma’lumotni deshifrlab, $R_B - 1$ - sonini tekshiradi va haqiqatdan A - foydalanuvchi bilan aloqa o‘rnatayotganiga ishonch hosil qiladi.

Bu protokolda R_A, R_B va $R_B - 1$ - sonlaridan takroran foydalaniladi. Agar kriptoanalitik avval foydalanilgan k -kalitni qo‘lga kiritsa, 3 - bosqichda A -foydalanuvchi nomidan V - foydalanuvchiga ma’lumot uzatishi mumkin.

Kriptoanalitik hujumi ketma-ketligi

1. Kriptoanalitik V - foydalanuvchiga quyidagi ma'lumotni uzatadi:

$$S \to V : E_B(k, A).$$

2. V -foydalanuvchi k - kalitni oladi, tasodifiiy R_B -sonini generatsiya qiladi va A - foydalanuvchiga quyidagi ma'lumotni uzatadi:

$$V \to A: E_k(R_B).$$

3. Kriptoanalitik ma'lumotni qo'lga kiritib, k -kalit yordamida ochadi. U V - foydalanuvchiga quyidagi ma'lumotni uzatadi:

$$S \to V: E_k(R_B - 1) \quad .$$

4. V - foydalanuvchi ma'lumotni deshifrlab $R_B - 1$ni olib tekshiradi. So'ngra A - foydalanuvchi bilan aloqa o'rnatayotganiga ishonch hosil qiladi.

Kriptoanalitik V - foydalanuvchini shu tartibda ishontirishi mumkin.

Xulosa. Bu kamchilikni bartaraf etish uchun vaqt belgisidan foydalanish maqsadga muvofiq bo'ladi. Chunki 2) - bosqichda ishonchli tomon ma'lumotiga vaqt belgisi qo'shiladi. Vaqt belgisi tizimda aniq va ishonchli vaqtni talab qiladi.

Agar kriptoanalitik A - foydalanuvchining umumiy kalitini qo'lga kiritsa, k -seans kalitga ham ega bo'lishi va B - foydalanuvchi bilan aloqa bog'lashi mumkin. Bu holat A - foydalanuvchi o'zining umumiy kalitini o'zgartirgan taqdirda ham davom etishi mumkin.

Asimmetrik kalitli algoritmlar yordamida mahfiy aloqani tashkil qilish protokollari

Elektron raqamli imzodan foydalanmagan holda kalitni uzatish protokollari

Seansning mahfiy bo'lgan k -kalitini uzatish uchun quyidagi qadamdan iborat bo'lgan protokolni ko'rib o'tamiz. Ushbu protokol bizga bir tomonlama identifikatsiyani ta'minlash uchun xizmat qiladi:

$$A \to B: E_{k_B}(k, t, A) \quad .$$

Bu yerda :

Ye – asimmetrik shifrlash algoritmi;

t – vaqt belgisi.

A - foydalanuvchi simmetrik shifrlash algoritmining kalitini, vaqt metkasini va o'zining identifikatsion raqamini birlashtirib, V - foydalanuvchining ochiq kaliti yordamida shifrlaydi va unga uzatadi. V - foydalanuvchi o'zining yopiq kaliti yordamida bu ma'lumotni deshifrlaydi. Natijada k -kalit, t - vaqt metkasi va A - foydalanuvchining identifikatsiya raqamiga ega bo'ladi. V - foydalanuvchi t - vaqt metkasini tekshiradi, agar to'g'ri bo'lsa k -kalitni haqiqiy deb qabul qiladi.

Tomonlar o'rtasida o'zaro **identifikatsiya**ni ta'minlash uchun esa quyidagi protokoldan foydalanish mumkin. Bu protokol quyidagicha amalga oshiriladi:

A - foydalanuvchi k_1 -kalitni va o'zining identifikatsiya raqamini birlashtirib, V - foydalanuvchining ochiq kaliti bilan shifrlaydi va V ga uzatadi

$$A \to B: E_B(k_1, A) \quad .$$

V - foydalanuvchi bu ma'lumotni deshifrlaydi, natijada k_1 va A ga ega bo'ladi. U k_1 - kalitni olib unga k_2 - kalitni birlashtirib, A -foydalanuvchining ochiq kaliti bilan shifrlaydi va A ga uzatadi

$$V \to A: E_A(k_1, k_2) \quad .$$

A - foydalanuvchi bu ma'lumotni o'zining yopiq kaliti bilan deshifrlab, k_1 va k_2 larga ega bo'ladi. k_1 kalitni tekshirib, V -foydalanuvchini identifikatsiyalaydi. Endi V -foydalanuvchi uni identifikatsiyalashi uchun k_2 -kalitni V -foydalanuvchining ochiq kaliti yordamida shifrlab, V - foydalanuvchiga uzatadi:

$$A \rightarrow B: E_B(k_2).$$

V - foydalanuvchi bu ma’lumotni o‘zi yopiq kaliti bilan deshifrlab esa k_2 - kalitni oladi. Agar bu kalit 2 - bosqichda yuborilgan k_2 - kalitga teng bo‘lsa, V - foydalanuvchi A ni identifikatsiyalaydi. Natijaviy k - kalit biror

$$k = f(k_1, k_2)$$

funksiya yordamida hisoblab topiladi.

Elektron raqamli imzodan foydalanib kalitni almashish protokollari

Elektron raqamli imzodan foydalanib generatsiya qilingan kalitni asimmetrik shifrlash algoritmidan foydalanib almashish protokollari uch turga bo‘linadi. Bu protokollarda elektron raqamli imzoni tekshirish algoritmi ikkala tomonga ham ma’lum deb hisoblanadi.

1) Raqamli imzo qo‘yilgan kalitni shifrlash:

$$A \rightarrow B: E_B(k, t, S_A(B, k, t)),$$

A - foydalanuvchi V ning identifikatsiya raqamini, generatsiya qilingan k - kalitni va t - vaqt belgisini birlashtirib, bu ma’lumotga o‘zining yopiq kaliti bilan elektron raqamli imzo qo‘yadi. Bundan so‘ng generatsiya qilingan k -kalitni, t -vaqt belgisini va raqamli imzo qo‘yilgan ma’lumotini birlashtirib, V - foydalanuvchining ochiq kaliti bilan shifrlaydi. Hosil bo‘lgan shifrmatnni V ga uzatadi. V - foydalanuvchi o‘zining yopiq kaliti bilan bu ma’lumotni deshifrlab, generatsiya qilingan k -kalitga, t - vaqt belgisiga va raqamli imzo qo‘yilgan ma’lumotga ega bo‘ladi. U A -foydalanuvchining ochiq kaliti yordamida elektron raqamli imzoni tekshiradi. Agar raqamli imzo to‘g‘ri bo‘lsa, bu ma’lumotni A - foydalanuvchi uzatganiga ishonch hosil qiladi va k - kalitni haqiqiy deb tan oladi.

2) Kalitni shifrlash va kalitga raqamli imzo qo‘yish:

$$A \rightarrow B: E_B(k, t), S_A(B, k, t)$$

A - foydalanuvchi generatsiya qilingan k -kalitni va t - vaqt belgisini birlashtirib, bu ma’lumotni V - foydalanuvchining ochiq kaliti bilan shifrlaydi. Shundan so‘ng u, V - foydalanuvchining identifikatsiya raqami, generatsiya qilingan k - kalit va t - vaqt belgisini birlashtirib, bu ma’lumotga o‘zining ochiq kaliti yordamida elektron raqamli imzo qo‘yadi. Keyin esa shifrmatnni va imzolangan ma’lumotni birlashtirib V - foydalanuvchiga uzatadi. V - foydalanuvchi bu ma’lumotni qabul qiladi va ma’lumotlarni ochishga kirishadi. Avvalo u o‘zining yopiq kalitidan foydalanib shifrmatnni deshifrlaydi va generatsiya qilingan k - kalit va t - vaqt metkasiga ega bo‘ladi. So‘ngra A - foydalanuvchining ochiq kaliti yordamida elektron raqamli imzoni tekshiradi. Agar elektron raqamli imzo to‘g‘ri bo‘lsa, bu ma’lumotni haqiqatdan ham A - foydalanuvchi uzatganiga ishonch hosil qiladi va k -kalitni haqiqiy deb qabul qiladi.

3) Shifrlangan kalitga elektron raqamli imzo qo‘yish:

$$A \rightarrow B: t, E_B(A, k), S_A(B, t, E_B(A, k))$$

A - foydalanuvchi o‘zining identifikatsiya raqamini va generatsiya qilingan k - kalitni birlashtirib, V - foydalanuvchining ochiq kaliti bilan shifrlaydi. Shundan so‘ng u, V - foydalanuvchining identifikatsiya raqamini, t - vaqt belgisini va shifrmatnni birlashtirib, bu ma’lumotga elektron raqamli imzo qo‘yadi. So‘ngra t - vaqt belgisi, shifrmatn va raqamli imzo qo‘yilgan ma’lumotni birlashtirib V - foydalanuvchiga uzatadi. V - foydalanuvchi bu malumotni oladi va uni ochishga kirishadi. U shifrmatnni o‘zining yopiq kaliti bilan ochadi, natijada A - foydalanuvchining identifikatsiya raqami va generatsiya qilingan k - kalitga ega bo‘ladi. So‘ngra u raqamli imzo qo‘yilgan ma’lumotni A - foydalanuvchining ochiq kaliti yordamida tekshiradi. Agar raqamli imzo to‘g‘ri bo‘lsa, generatsiya qilingan k - kalitni haqiqiy deb qabul qiladi.

SKEY dasturi

Ma’lumotning xavfsizligini ta’minlash uchun SKEY (ma’lumotning haqiqiyligini tekshiruvchi) dasturidan foydalanish mumkin. Bu dastur quyidagicha amalga oshiriladi.

A – foydalanuvchi autentifikatsiya masalasini hal qilish uchun tasodifiyR sonini kiritadi. Kompyuter $f(R)$, $f(f(R))$, $f(f(f(R)))$, ... qiymatlarini hisoblaydi. Bu qiymatlarni mos xolda $x_1, x_2, x_3 \ldots x_{100}$ deb belgilaymiz. A foydalanuvchi bu ro‘yxatni qog‘ozga yozib oladi va berkitadi. Bundan tashqari, kompyuter x_{101} qiymatni shifrlanmagan xolda saqlaydi.

A – foydalanuvchi sistemaga birinchi marta kirishi uchun o‘z ismini va x_{101} qiymatini kiritadi. Kompyuter $f(x_{100})$ning qiymatini hisoblaydi va x_{101} bilan solishtiradi. Agar qiymatlar teng bo‘lsa, xaqiqatdan ham A – foydalanuvchi ekanligini tasdiqlaydi. So‘ngra kompyuter ma’lumotlar bazasidagi x_{101} qiymatni x_{100}bilan almashtirib qo‘yadi. A – foydalanuvchi esa x_{100} ning qiymatini o‘z ro‘yxatidan o‘chiradi.

Keyinchalik A – foydalanuvchi har safar sistemaga kirishida oxirgi o‘chirilmagan sonni kiritadi, masalan i. Kompyuter $f(x_i)$ qiymatni hisoblaydi va ma’lumotlar bazasida saqlanayotgan x_{i+1} son bilan solishtiradi. SKEY dasturida har bir son bir marta ishtirok etadi. Bunday xolatda esa kriptoanalitik hech qanday foydali ma’lumotga ega bo‘la olmaydi.

MTI protokoli

MTI protokolining nomi uning mualliflari hisoblangan *T. Matsumoto I. Takashima va X. Imai*lar sharafiga qo‘yilgan. Bu protokol ham Diffi-Xellman protokoliga o‘xshash bo‘lib, uning kriptobardoshliligi chekli maydonda diskret logarifmlashga asoslangan [14, 20]. Biroq undan farqli tomoni shundaki, MTI protokolida kriptobardoshliligini oshirish maqsadida qo‘shimcha a va b o‘zgaruvchilardan foydalaniladi. Ushbu protokolning amallar ketma-ketligi quyidagicha bajariladi. Eng avvalo A va V -foydalanuvchilar katta tub son p va uning primitiv ildizi α ning qiymati haqida kelishib oladilar.

A - foydalanuvchi o‘z maxfiy kaliti a, $1 \le a \le p-2$ ni generatsiya qiladi va bu kalit yordamida

$$z_A = \alpha^a \bmod p$$

ifodani xisoblaydi. A -foydalanuvchi hosil bo‘lgan qiymatni V-foydalanuvchiga uzatadi:

$$\text{A} \to \text{V}: \; z_A = \alpha^a \bmod p,$$

V - foydalanuvchi bu ma’lumotni qabul qiladi. U o‘zining yopiq kaliti b, $1 \le b \le p-2$ ni generatsiya qiladi. Bu yopiq kalit yordamida

$$z_B = \alpha^b \bmod p$$

ifodani hisoblaydi va natijani A -foydalanuvchiga uzatadi:

$$\text{V} \to \text{A}: \; z_B = \alpha^b \bmod p.$$

A - foydalanuvchi z_B ni qabul qiladi. A va V -foydalanuvchilar umumiy mahfiy kalitni generatsiya qilish uchun mos holda o‘zlarining x, $1 \le x \le p-2$ va y, $1 \le y \le p-2$ tasodifiy sonlarini generatsiya qilishlari zarur. A -foydalanuvchi o‘zining tasodifiy x -sonini generatsiya qilib,

$$\alpha^x \bmod p$$

ifodani hisoblaydi va uni V -foydalanuvchiga uzatadi:

$$\text{A} \to \text{V}: \; \alpha^x \bmod p.$$

V - foydalanuvchi bu ma’lumotni qabul qiladi. U o‘zining tasodifiy y - sonini generatsiya qilib, $\alpha^y \bmod p$ ifodani hisoblaydi. Hosil bo‘lgan natijani A - foydalanuvchiga uzatadi. Shu vaqtdan boshlab, V - foydalanuvchi α^x va z_A - ma’lumotlarga ega. Endi u o‘zining tasodifiy soni va yopiq kalitidan foydalanib quyidagi ifodani hisoblaydi:

$$k = (\alpha^x)^b \cdot z_A^y,$$

$$\text{V} \to \text{A}: \; \alpha^y \bmod p.$$

A - foydalanuvchi bu ma'lumotni qabul qiladi. Endi A - foydalanuvchi α^y va z_B - ma'lumotlarga ega. U o'zining tasodifiy soni va yopiq kalitidan foydalanib ushbu ifodani hisoblaydi: $k = (\alpha^y)^a \cdot z_B^x$.

Natijaviy kalitning umumiy ko'rinishi esa quyidagicha:

$$k = (\alpha^y)^a \cdot z_B^x = (\alpha^x)^b \cdot z_A^y = \alpha^{xb+ya} \bmod p.$$

MTI protokoli shu tartibda amalga oshiriladi. Unda kriptoanalitikning ihtiyoriy almashtirishi tomonlardagi kalitning qiymati turlicha bo'lishiga olib keladi. Bu esa uzatilayotgan ma'lumotni o'qish imkoniyatini butunlay yo'qotadi.

Quyida MTI prtokoli uchun ham misol keltiriladi.

$p = 9531$

$\alpha = 1647$

$A:\ a = 126$

$A:\ Z_a = \alpha^a \bmod p = 1647^{126} \bmod 9531 = 3375$

$A \to B:\ Z_a = 3375$

$B:\ b = 98$

$B:\ Z_B = \alpha^b \bmod p = 1647^{98} \bmod 9531 = 8775$

$B \to A:\ Z_B = 8775$

$A:\ x = 8643$

$A:\ X = \alpha^x \bmod p = 1647^{8643} \bmod 9531 = 972$

$A \to B:\ X = 972$

$B:\ k_1 = (\alpha^x)^b Z_A^y \bmod p = X^b Z_A^y \bmod p = 972^{98} \cdot 3375^{6983} \bmod 9531 = 3564$

$B:\ y = 6983$

$B:\ Y = \alpha^y \bmod p = 1647^{6983} \bmod 9531 = 4131$

$B \to A:\ Y = 4131$

$A:\ k_2 = (\alpha^y)^a Z_b^x \bmod p = Y^a Z_b^x \bmod p = 4131^{126} \cdot 8775^{972} \bmod 9531 = 3564$

javob: $k_1 = k_2 = k = 3564.$

Dass protokoli

Dass protokoli ochiq kalitli shifrlash algoritmi va simmetrik shifrlash algoritmidan foydalanadi. Shunday qilib, A va V -foydalanuvchilar o'zlarining yopiq kalitlariga egalar. Arbitr ularning ochiq kalitlariga raqamli imzo qo'yadi.

1) A - foydalanuvchi V - foydalanuvchining ismini arbitrga uzatadi:

$$\mathrm{A} \to \mathrm{W}:\ B.$$

2) Arbitr V - foydalanuvchining ismini va ochiq kalitini birlashtirib o'zining yopiq kaliti bilan raqamli imzo qo'yadi. Hosil bo'lgan shifrmatnni A - foydalanuvchiga uzatadi:

$$\mathrm{W} \to \mathrm{A}:\ S_{\mathrm{W}}(B, k_B).$$

3) A - foydalanuvchi haqiqatdan ham V -foydalanuvchining ochiq kalitini olganligini aniqlash uchun arbitrning raqamli imzosini tekshiradi. U tasodifiy seans kalit k va tasodifiy ochiq/yopiq juftlik kalit k_p ni generatsiya qiladi. A -foydalanuvchi vaqt belgisini tasodifiy seans kalit yordamida shifrlaydi. So'ngra hayotiy vaqt, o'zining ismi va ochiq/yopiq juftlik kalitlarni birlashtirib o'zining yopiq kaliti k_A yordamida raqamli imzo qo'yadi. Nihoyat u k - seans kalitni V - foydalanuvchining ochiq kaliti yordamida shifrlaydi va unga juftlik k_p - kalit yordamida raqamli imzo qo'yadi. A - foydalanuvchi bularning barchasini V - foydalanuvchiga uzatadi:

$$\mathrm{A} \to \mathrm{V}:\ E_k(t_A), S_{\mathrm{k_A}}(L, A, k_P), S_{k_P}(E_{k_B}(k)).$$

4) V - foydalanuvchi A - foydalanuvchining ismini arbitrga uzatadi:

$$V \to W: \ A.$$

5) Arbitr A - foydalanuvchining ismi va uning ochiq kalitini birlashtirib, o‘zining yopiq kaliti bilan raqamli imzo qo‘yadi. Natijani V -foydalanuvchiga uzatadi:

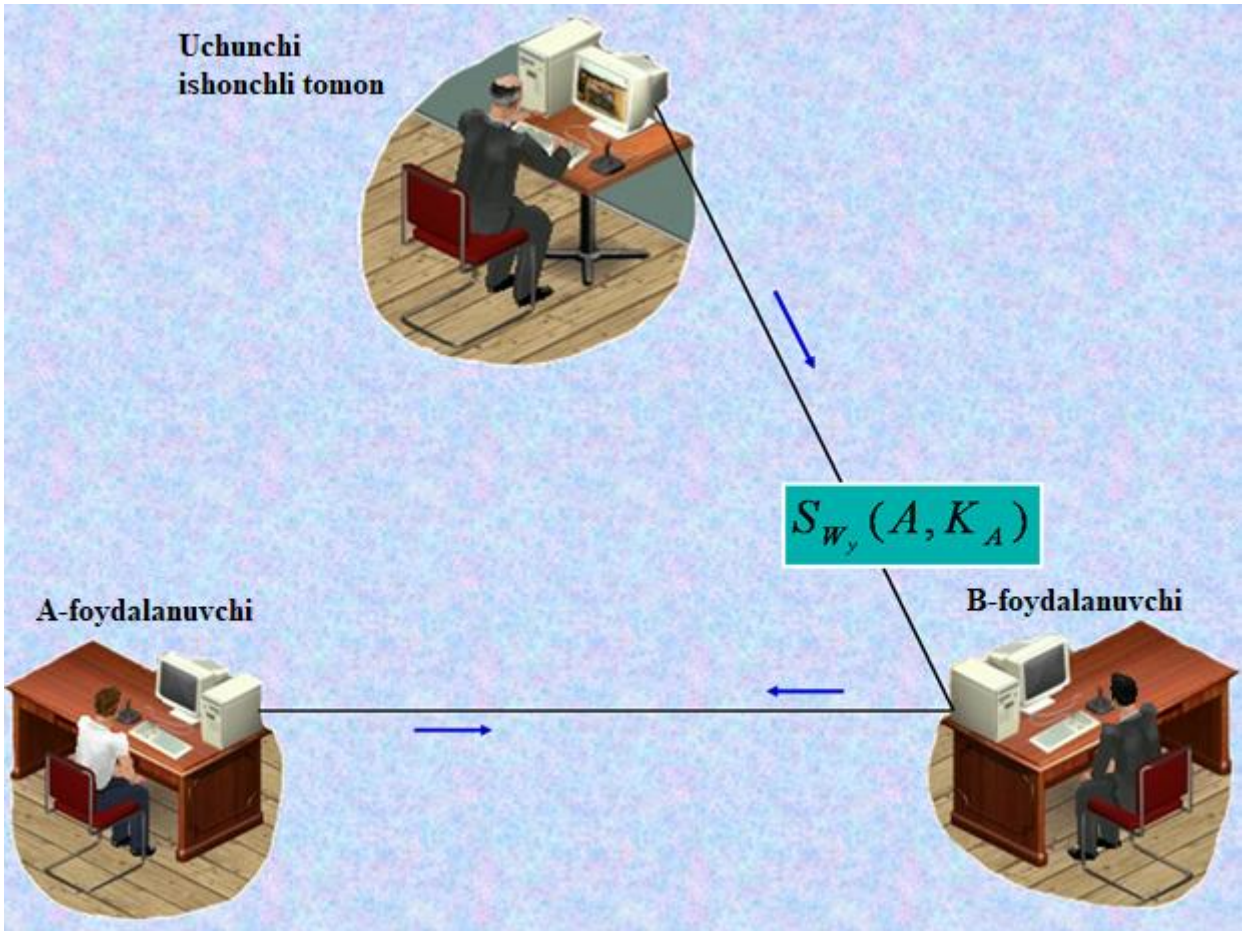
$$W \to V: \ S_W(A, k_A).$$

6) V - foydalanuvchi arbitrning haqiqatdan ham A -foydalanuvchining ochiq kalitini olganligini aniqlash uchun arbitrning raqamli imzosini tekshiradi. So‘ngra u, A - foydalanuvchi uzatgan ma’lumotdagi A - foydalanuvchining raqamli imzosini tekshiradi. Imzo haqiqiy bo‘lsa, ochiq/yopiq juftlik k_p - kalitni haqiqatdan A - foydalanuvchi uzatganiga ishonch hosil qiladi. So‘ngra ushbu kalit yordamida ikkinchi raqamli imzoni tekshiradi. Agar bu imzo ham haqiqiy bo‘lsa, shifrmatnni o‘zining yopiq kaliti yordamida deshifrlaydi va natijada k - tasodifiy seans kalitga ega bo‘ladi. Endi u seans kalit yordamida birinchi shifrmatnni deshifrlaydi va vaqt belgisini oladi. Agar bu vaqt belgisi belgilangan vaqt intervali oralig‘ida bo‘lsa, ma’lumotning haqiqiyligiga, qayta uzatilmagan ekanligiga yana bir karra ishonch hosil qiladi.

7) Agar ikki tomonlama identifikatsiya talab qilinsa, V -foydalanuvchi yangi vaqt belgisini seans kalit yordamida shifrlab, A -foydalanuvchiga uzatadi:

$$V \to A: \ E_k(t_B) \ .$$

8) A - foydalanuvchi k -seans kalit yordamida t_B - vaqt belgisini deshifrlaydi va ma’lumotning xozirgi paytda uzatilganiga ishonch xosil qiladi. Ana shu holda ikki tomonlama identifikatsiya ta’minlanadi.



***Denning-Sacco* protokoli**

Ushbu protokolda ham asimmetrik shifrlash algoritmidan foydalaniladi. Arbitr barcha foydalanuvchilarning ochiq kalitlarini saqlovchi ma’lumotlar bazasini boshqaradi.

1) A - foydalanuvchi o‘zining va V -foydalanuvchining ismini arbitrga uzatadi:

$$A \to W: \ A, B \ .$$

2) Arbitr V - foydalanuvchining ismini va ochiq kalitini birlashtirib o‘zining yopiq kaliti yordamida raqamli imzo qo‘yadi. Shuningdek u A - foydalanuvchining ismini va ochiq kalitini birlashtirib o‘zining yopiq kaliti bilan raqamli imzo qo‘yadi. Arbitr ikkala ma’lumotni ham A - foydalanuvchiga uzatadi:

$$W \to A: \ S_W(B, k_B), S_W(A, k_A)$$

1) A - foydalanuvchi arbitrning ochiq kaliti yordamida raqamli imzoni tekshiradi. Agar imzo to‘g‘ri bo‘lsa, V - foydalanuvchining ochiq kalitini haqiqiy deb qabul qiladi. U tasodifiy seans kalit va vaqt metkasini birlashtirib, o‘zining yopiq kaliti yordamida raqamli imzo qo‘yadi. Natijani V - foydalanuvchining ochiq kaliti yordamida shifrlaydi. A -foydalauvchi bu shifrmatnga arbitrdan qabul qilgan ikkita ma’lumotni birlashtirib V -foydalanuvchiga uzatadi:

$$A \to V: \ E_B(S_A(k, t_A))S_W(A, k_A), S_W(B, k_B) \ .$$

2) V - foydalanuvchi arbitrning ochiq kalitidan foydalanib uning raqamli imzosini tekshiradi. Agar imzo to‘g‘ri bo‘lsa, A - foydalanuvchining ochiq kalitini haqiqiy deb qabul qiladi. U o‘zining yopiq kaliti bilan shifrmatnni deshifrlaydi. Xosil bo‘lgan ma’lumotdan A - foydalanuvchining raqamli imzosini tekshiradi. Agar imzo haqiqiy bo‘lsa, seans kalit sifatida k ni qabul qiladi. Shuningdek, V - foydalanuvchi vaqt metkasini tekshirib, ma’lumotning yaqin vaqt ichida uzatilganiga ishonch hosil qiladi.

Shu vaqtdan boshlab, A va V - foydalanuvchilar seans kalitga ega bo‘ldilar va xavfsiz seans aloqasini o‘rnatishlari mumkin bo‘ladi.

Biroq bu protokolning quyidagicha kamchiligi mavjud. V - foydalanuvchi A - foydalanuvchining nomidan ish ko‘rishi mumkin.

1) V - foydalanuvchi arbitrga o‘zining va 3 chi S -foydalanuvchining ismini uzatadi:

$$V \to W: \ B, C \ .$$

2) Arbitr V - foydalanuvchining ismini va ochiq kalitini birlashtirib o‘zining yopiq kaliti yordamida raqamli imzo qo‘yadi. Shuningdek u S- foydalanuvchining ismini va ochiq kalitini birlashtirib o‘zining yopiq kaliti bilan raqamli imzo qo‘yadi. Arbitr ikkala ma’lumotni ham V foydalanuvchiga uzatadi:

$$W \to V: \ S_W(B, k_B), S_W(C, k_C) \ .$$

3) V - foydalanuvchi arbitrning ochiq kaliti yordamida raqamli imzoni tekshiradi. Agar imzo to‘g‘ri bo‘lsa, S - foydalanuvchining ochiq kalitini haqiqiy deb qabul qiladi. U A - foydalanuvchidan olgan raqamli imzo qo‘yilgan ma’lumotni S - foydalanuvchining ochiq kaliti yordamida shifrlaydi. V - foydalanuvchi bu shifrmatnga arbitrdan qabul qilgan ikkita ma’lumotni birlashtirib, S - foydalanuvchiga uzatadi:

$$V \to S: \ E_C(S_A(k, t_A))S_W(A, k_A), S_W(C, k_C) \ .$$

4) S - foydalanuvchi arbitrning ochiq kalitidan foydalanib uning raqamli imzosini tekshiradi. Agar imzo to‘g‘ri bo‘lsa, A - foydalanuvchining ochiq kalitini haqiqiy deb qabul qiladi. U o‘zining yopiq kaliti bilan shifrmatnni deshifrlaydi. Hosil bo‘lgan ma’lumotdan A - foydalanuvchining raqamli imzosini tekshiradi. Agar imzo xaqiqiy bo‘lsa, seans kalit sifatida k ni qabul qiladi. Shuningdek S - foydalanuvchi vaqt belgisini tekshirib, ma’lumotning yaqin vaqt ichida uzatilganiga ishonch hosil qiladi.

Endi S - foydalanuvchi o‘zini A - foydalanuvchi bilan aloqa o‘rnatgan deb hisoblaydi. V - foydalanuvchi esa uni osongina aldadi. Haqiqatdan ham vaqt belgisi o‘zining vaqt intervalidan o‘tgungacha V - foydalanuvchi tarmoqdagi ixtiyoriy foydalanuvchini aldashi mumkin. Lekin buni osongina bartaraf etish mumkin. Buning uchun 3) - bosqichda shifrlanishi kerak bo‘lgan ma’lumotga foydalanuvchilarning ismlarini qo‘shish kerak:

$$E_B(S_A(A, B, k, t_A))S_W(A, k_A), S_W(B, k_B) \ .$$

Endi V - foydalanuvchi eski ma’lumotni S - foydalanuvchiga takroran uzata olmaydi, chunki bu faqat A va V - foydalanuvchilar seans aloqasi uchun yaratilgani yaqqol namoyon bo‘ladi. Ana

shu tarzda *Denning - Sacco* protokolining kamchiligi bartaraf etiladi.

Woo-Lam **protokoli**

Bu protokolda ham asimmetrik shifrlash algoritmidan foydalaniladi.

1) A - foydalanuvchi o‘zining va V - foydalanuvchining ismini arbitrga uzatadi:

$$\text{A} \to \text{W}: \ A, B \ .$$

2) Arbitr V - foydalanuvchining ochiq kalitiga o‘zining yopiq kaliti bilan raqamli imzo qo‘yadi va A foydalanuvchiga uzatadi:

$$\text{W} \to \text{A}: \ S_W(k_B) \ .$$

3) A - foydalanuvchi arbitrning imzosini tekshiradi. Agar imzo to‘g‘ri bo‘lsa, V - foydalanuvchining ochiq kalitini xaqiqiy deb qabul qiladi. So‘ngra u o‘zining tasodifiy sonini V - foydalanuvchining ochiq kaliti bilan shifrlaydi. A - foydalanuvchi o‘zining ismini va shifrmatnni V -foydalanuvchiga uzatadi:

$$\text{A} \to \text{V}: \ A, E_B(R_A) \ .$$

4) V - foydalanuvchi o‘zining yopiq kaliti yordamida shifrmatnni deshifrlaydi va natijada A - foydalanuvchining tasodifiy soniga ega bo‘ladi. Endi u tasodifiy sonni arbitrning ochiq kaliti yordamida shifrlaydi. So‘ngra A - foydalanuvchining, o‘zining ismi va shifrmatnni arbitrga uzatadi:

$$\text{V} \to \text{W}: \ A, B, E_{k_T}(R_A) \, .$$

5) Arbitr A - foydalanuvchining ochiq kaliti k_A ga o‘zining yopiq kaliti biln raqamli imzo qo‘yadi. U A - foydalanuvchining tasodifiy soni, seans kalit, A va V - foydalanuvchilarning ismlarini birlashtirib, o‘zining yopiq kaliti bilan raqamli imzo qo‘yadi va V - foydalanuvchining ochiq kalitidan foydalanib shifrlaydi. Arbitr ikkala ma’lumotni ham V - foydalanuvchiga uzatadi:

$$\text{W} \to \text{B}: \ S_W(k_A), E_{k_B}(S_W(R_A, k, A, B)) \, .$$

6) V - foydalanuvchi arbitrning raqamli imzosini tekshiradi. Agar raqamli imzo to‘g‘ri bo‘lsa, A - foydalanuvchining ochiq kalitini haqiqiy deb qabul qiladi. So‘ngra u shifrmatnni o‘zining yopiq kaliti bilan deshifrlaydi. Hosil bo‘lgan ma’lumotdagi abitrning raqamli imzosini tekshiradi. Agar imzo to‘g‘ri bo‘lsa, k -seans kalitni xaqiqiy deb qabul qiladi. Endi V -foydalanuvchi 5) - bosqichda arbitrdan qabul qilgan raqamli imzo qo‘yilgan ma’lumotga o‘zining tasodifiy sonini birlashtirib, A -foydalanuvchining ochiq kaliti bilan shifrlaydi. Hosil bo‘lgan shifrmatnni A-foydalanuvchiga uzatadi:

$$\text{V} \to \text{A}: \ E_{k_A}(S_W(R_A, k, A, B), R_B) \, .$$

7) A - foydalanuvchi shifrmatnni o‘zining yopiq kaliti yordamida deshifrlaydi. U arbitrning raqamli imzosini tekshiradi. Agar imzo to‘g‘ri bo‘lsa, k - seans kalitni haqiqiy deb qabul qiladi. So‘ngra V -foydalanuvchining tasodifiy sonini k -seans kalit yordamida shifrlab V - foydalanuvchiga uzatadi:

$$\text{A} \to \text{V}: \ E_k(R_B) \ .$$

8) V - foydalanuvchi seans kalit yordamida shifrmatnni deshifrlaydi. Hosil bo‘lgan o‘zining tasodifiy sonini o‘zgargan yoki o‘zgarmaganligini aniqlaydi. Agar u o‘zgarmagan bo‘lsa, A - foydalanuvchi bilan aloqa o‘rnatilganiga ishonch hosil qiladi. Ana shu tarzda ikki tomonlama identifikatsiya ta’minlanadi.

Nazorat savollari

1. Simmetrik shifrlash tizimlari asosida kalitlarni almashish.
2. Shamir protokoli.
3. Nidxam-Shryoder protokoli.
4. ERIga asoslangan autentifikatsiyalash va kalitlarni almashinish protokoli.
5. SKEY dasturi va uning maqsadi.
6. MTI protokoli va Diffi-Xelman protokolining farqi.

7. Denning-Sacco protokoli.

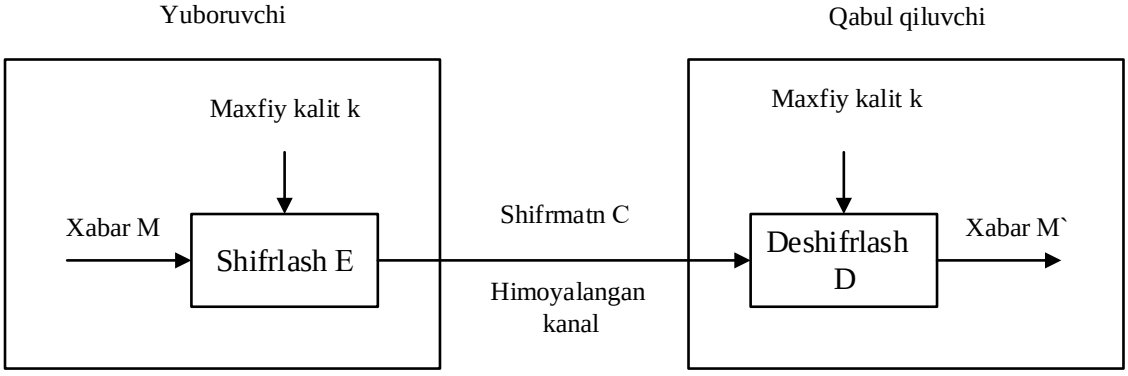
3-ma’ruza. Ochiq kalitlar infratuzilmasi

Reja:

1. Sertifikatlarni boshqarish infratuzilmasi.
2. PKCS standartlari.
3. DER kodirovkasi.
4. Ochiq kalitlar infratuzilmasidan foydalanuvchi xizmatlar.
5. X.509 sertifikati.

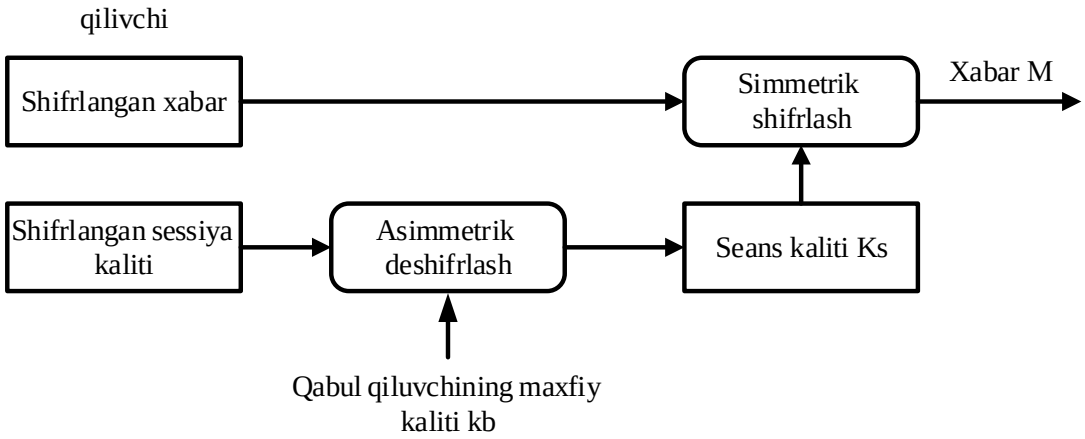
Sertifikatlarni boshqarish infratuzilmasi

Ma’lumotlarni tarmoq orqali shifrlab yuborishda odatda simmetrik kriptotizimlardan foydalaniladi. Simmetrik kriptografik tizimlar 5zining tezkorli va ishonchligi bilan ajralib turadi. Quyidagi 3.1 – rasmda simmetrik kriptotizimlarning umumiy ko‘rinishi keltirilgan bo‘lib, ma’lumotni shifrlashda va deshifrlash aynan bir kalitdan foydalaniladi. Bu esa o‘z navbatida yagona kalitni tomonlar orasida xavfsiz almashinish vazifasi qo‘yadi.



3.1– rasm. Maxfiy kalitli kriptografik tizim

Ushbu muammoni bartaraf etishda odatda ochiq kalitli kriptografik tizimlardan foydalaniladi. Buning uchun simmetrik kriptografik tizim kaliti ochiq kalitli kriptografik usulda shifrlanib yuboriladi. Qabul qiluvchi esa dastlab maxfiy kalit bilan shifrlangan kalitni deshifrlaydi va u bilan shifrlangan ma’lumotni deshifrlaydi (3.2 - rasm).



3.2– rasm. Gibrid shifrlash usuli

Gibrid usul bir qarashda xavfsiz ko‘rinsada, aslida bu usul ham xavflardan holi emas. Bulardan biri bu – ochiq kalitni kimga tegishli eganligini bilishdir. Agar kalitni shifrlashdan oldin ochiq kalitni kimga tegishligini aniqlamaslik, juda katta xavfga olib keladi. Ochiq kalitni kimga tegishligini aniqlash va uni butunligini ta’minlashda amalda keng qo‘llanilayotgan usullardan biri bu – ochiq kalitlar infratuzilmasidir (*public key infrastructre*).

Ochiq kalit kriptografiyasi uchun kalitlarni boshqarishda quyidagi ikkita talab qo‘yiladi:

- ***Shaxsiy kalit maxfiyligini saqlash.*** Kalitni boshqarishning hayotiy sikli davomida barcha tomonlardan maxfiy tarzda saqlanishi shart.
- ***Ochiq kalit kafolati.*** Ochiq kalitli kriptografiyada ochiq kalit domenda ochiq bo‘lishi va barchaga ko‘rinishi shart. Joriy holda ochiq kalitning to‘g‘riligiga hech qanday kafolat bo‘lmaydi. Shuning uchun ochiq kalitlarni butunligini ta’minlash uchun qo‘shimcha biror vosita yoki usuli talab etiladi.

Ochiq kalitlar yoki *sertifikatlar infratuzilmasi* (public key infrastructure, PKI) ochiq kalitlardan xavfsiz foydalanish uchun kerakli bo‘lgan barcha “narsalar”ni o‘zi ichiga oladi. Ochiq kalitlar infratuzilmasi ochiq kalitni identifikatsiyalash va uning taqsimotini ta’minlaydi. PKI anatomiyasi quyidagi komponentlarni o‘z ichiga oladi:

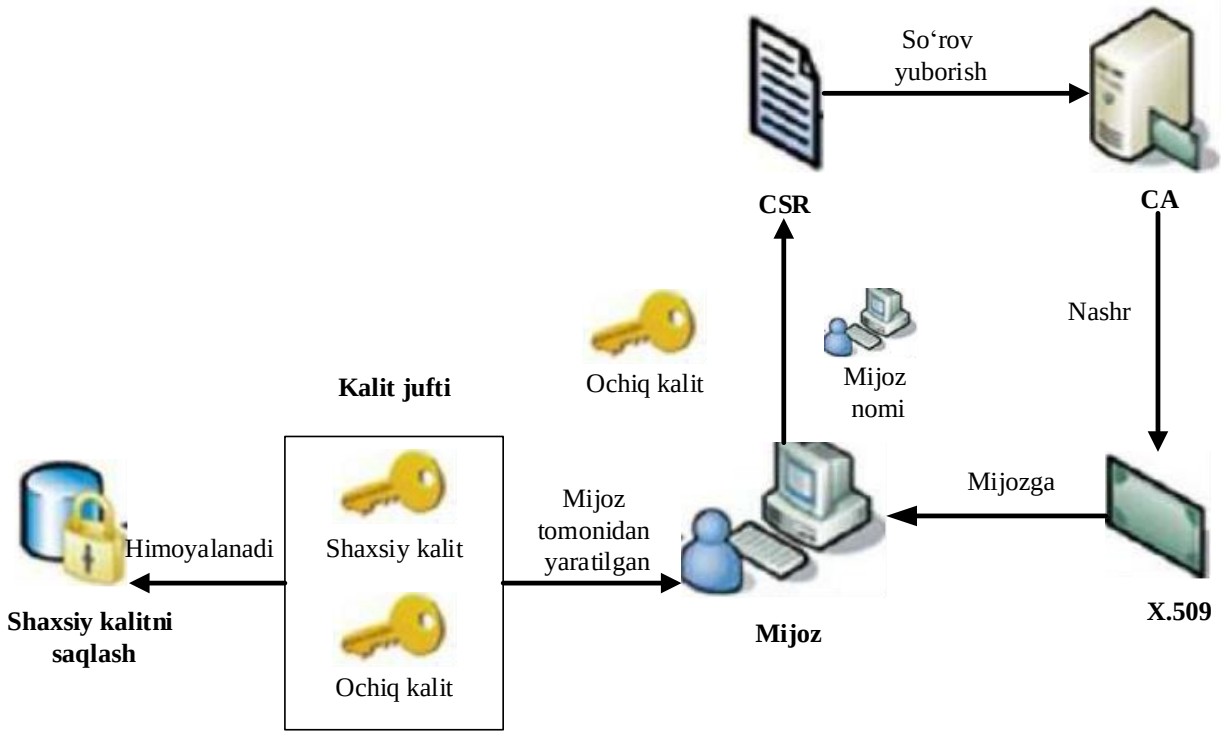
- ochiq kalit sertifikatini, odatda “raqamli sertifikat” shaklida bo‘ladi (masalan, X.509);
- shaxsiy kalit tokenlari;
- sertifikatlash markazi;
- ro‘yxatga olish markazi;
- sertifikatlarni boshqarish tizimi.

Raqamli sertifikat (yoki ochiq kalit sertifikati yoki qisqacha sertifikat) odatda foydalanuvchi nomi, uning ochiq kaliti va *tegishli tashkilot* (*certificate authority, CA*) tomonidan tasdiqlangan imzodan iborat bo‘ladi. Masalan, A tomonning sertifikati quyidagidan iborat bo‘ladi:

$$M = (\text{А томон номи}, \text{А томоннинг очиқ калити}) \text{ ва } S = [M]_{CA}$$

Sertifikatni tekshirish uchun B tomondan $\{S\}_{CA}$ ni hisoblash va uni M ma’lumotga mosligini tasdiqlash talab etiladi. Agar tegishli tashkilot tomonidan qo‘yilgan imzo tasdiqlanganda A tomonning ochiq kalitini haqiqiyligi tasdiqlanadi.

Umumiy holda raqamli sertifikatni olish jarayoni quyidagi 3.3 – rasmda ko‘rsatilgan ketma – ketlikda amalga oshiriladi.



3.3 – rasm. Raqamli sertifikatni olish jarayoni

Ushbu holatda tegishli tomonning *ishonchli* bo‘lishi juda muhim. Agar ushbu tashkilot imzo qo‘yish bilan bog‘liq bo‘lgan biror zaiflikni yoki xatolikni amalga oshirgan holda, jiddiy muammo kelib chiqish aniq.

Sertifikatlash markazi (CA)ning muhim vazifalari quyidagilardan iborat:

- *kalit juftini generatsiyalash* – SA va mijoz kelishgan holda yoki undan mustaqil ravishda kalit juftini generatsiyalashi mumkin;

- *raqamli sertifikatni yaratish* – mijoz o‘zi haqidagi ma’lumotlarni taqdim etgandan so‘ng SA uni butunligini ta’minlash uchun unga o‘z imzosini qo‘yadi;
- *sertifikatlarni chop etish* – SA mijoz sertifikatini qolgan foydalanuvchilar topishi va undan foydalana olishlari uchun uni chop etishi shart;
- *sertifikatlarni tekshirish* – SA mijozlarni raqamli sertifikatlariga qo‘ygan imzosini mijozlar tekshirishi uchun o‘zining ochiq kalitini foydalanish uchun e’lon qilishi shart;
- *sertifikatlarni bekor qilish* – muayyan vaqtda SA sertifikatlarni mijoz talabiga ko‘ra bekor qilishni amalga oshirishi va bekor qilingan sertifikatlar ro‘yxatini taqdim etishi shart.

Ochiq kalitli kriptogarfiyada quyidagi to‘rt toifadagi sertifikatlardan foydalaniladi:

- *1- toifa* – bu sertifikatlarni pochta manzilni taqdim etish orqali olish mumkin;
- *2 –toifa* – bu toifadagi sertifikatlar qo‘shimcha shaxsiy ma’lumotlarni taqdim etishni talab etadi;
- *3 – toifa* – ushbu sertifikatlarni mijoz identifikatorlari tasdiqlangandan so‘ng sotib olinishi kerak bo‘ladi;
- *4 – toifa* – juda yuqori kafolatni talab etuvchi tashkilotlar yoki hukumat tomonidan foydalaniladi.

Ro‘yxatga olish markazi. SA odatda tashkilot yoki korxona talab etgan sertifikat uchun ularni ma’lumotlarini tekshirish uchun uchinchi tomondan - *ro‘yxatga olish markazidan* foydalanishi mumkin. Bu markaz mijozlar uchun SA kabi ko‘rinsada, biroq ular imzo qo‘ya olmaydi.

Sertifikatlarni boshqarish tizimi. Sertifikatlarni nashr etish, vaqtinchalik yoki butunlay to‘xtatib qo‘yish, yangilash va bekor qilish sertifikatlarni boshqarish tizimi tomonidan amalga oshiriladi. Sertifikatlarni o‘chirib tashlash jarayoni mavjud emas va u bekor qilingan bo‘lsada keyinchalik qonuniy sohada foydalanilishi mumkinligi bilan xarakterlanadi.

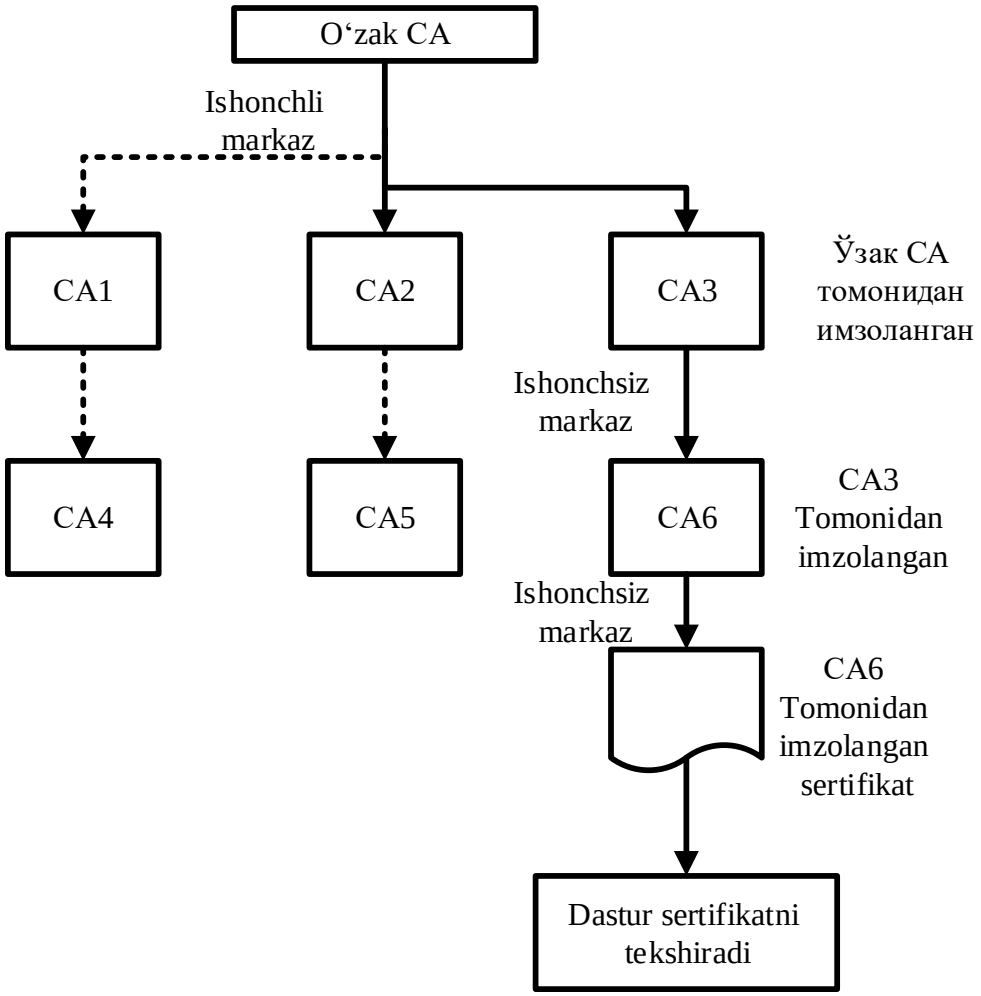
*Shaxsiy kalit tokenlari.*Ochiq kalit o‘zining butunligini talab etsa, imzolovchi kalit yoki shaxsiy kalit esa o‘zining maxfiyligini talab etadi. Buning uchun odatda kalitni kompyuterda saqlash orqali amalga oshirish mumkin. Ammo, bu holda hujumchi tomonidan kompyuter boshqaruvi qo‘lga olinsa, kalitning maxfiyligi buziladi. Shuning uchun kalitlar ko‘chib yuruvchi saqlagichlarda parol yordamida himoyalanadi.

Turli ishlab chiqaruvchilar shaxsiy kalitlarni saqlash uchun turli formatlardan foydalanadilar. Masalan, *Entrust* tashkiloti *.epf* formatdan foydalansa, *Verisign, GlobalSign* va *Baltimore*lar *.p12* formatidan foydalanadilar.

SA iyerarxiyasi. Global tarmoqning murakkab tuzilishga egaligi va katta hajmga egaligi natijasida barcha foydalanuvchilarni yagona SAga murojaatlarini ta’minlash murakkab vazifa. Bundan tashqari, yagona SA dan foydalanilganda, uning obro‘sizlantirilishi natijasida butun tizimning ishi to‘xtab qolishi mumkin.

Bunday hollarda esa *iyerarxiyali sertikat modeli*dan foydalanish mumkin. Bunda aloqa o‘rnatmoqchi bo‘lgan ikki tomon umumiy bo‘lmagan SA tomonidan berilgan sertifikatlardan foydalanish imkoniyatiga ega bo‘ladi (3.4 - rasm).

- O‘zak SA (root CA) iyerarxiyaning boshida bo‘lib, uning sertifikati o‘zini – o‘zi imzolagan sertifikat bo‘ladi.
- O‘zak SA ga qism SA bo‘lgan markazlar esa o‘zak SA tomonidan imzolanadi (rasmdagi SA1 va SA2 lar).
- Qism SAga qism SA lar esa o‘zlaridan yuqori SAlar tomonidan imzolanadi (rasmdagi SA5 va S6).

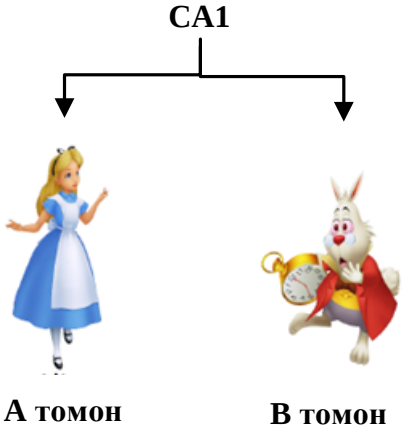


3.4 – rasm. SA iyerarxiyasi

Ochiq kalitli kriptografiyada quyidagi turdagi PKI arxitekturalaridan foydalaniladi:

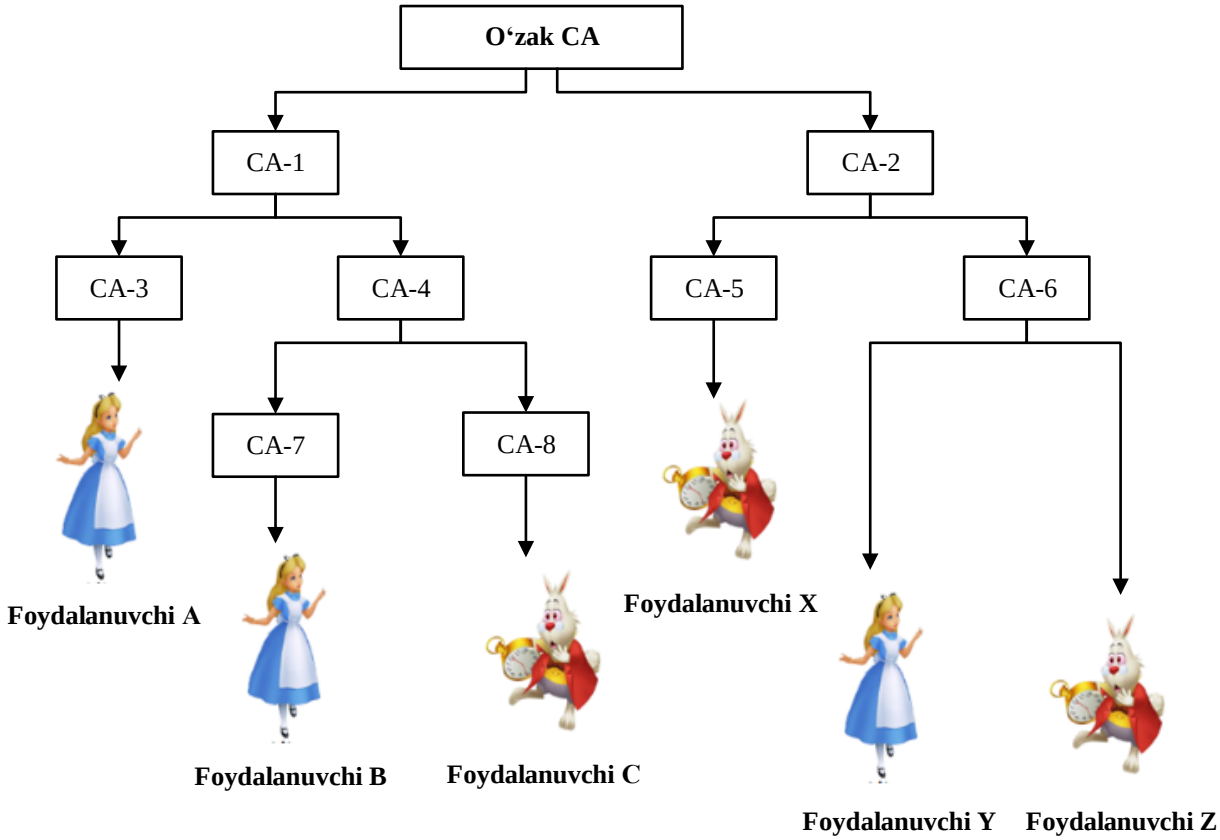
- Yagona SA ga asoslangan arxitektura;
- Tashkilot PKI ga asoslangan arxitektura;
- Gibrid PKI arxitektura.

Yagona SAga asoslangan arxitektura eng keng tarqalgan asos arxitekturalardan biri sanalib, tomonlar aynan ushbu SA tomonidan tanishtiriladi (3.5 - rasm).



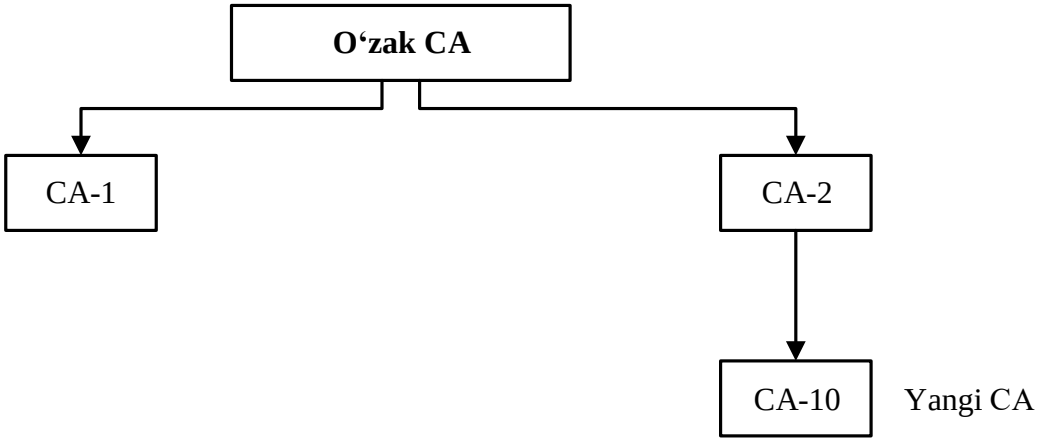
3.5 – rasm. Yagona SA arxitekturasi

Yagona SA ga asoslangan arxitekturani amalga oshirish oson sanalsada, juda keng miqiyosda foydalanishda muammolar vujudga kelishi mumkin. Sertifikatlardan foydalanish darajasini kengaytirish uchun tashkilot miqiyosida yoki undan kattaroq hududlarda tashkilot PKI arxitekturasidan foydalaniladi (6 - rasm).



3.6 – rasm. Tashkilot PKI arxitekturasi

Amalda 5taga yaqin global holda sertifikat beruvchi tashkilot bo‘lib, odatda ular kelishgan holda ishlaydi. Buning uchun ular gibrid arxitekturadan foydalanadilar. Bundan foydalanuvchilarning ayrimlari boshqa SA foydalanuvchilari bo‘lsa qolganlari boshqa SAlar foydalanuvchilari bo‘ladi. Biroq, gibrid arxitektura asosida turli SAlardan bo‘lgan foydalanuvchilar bir birlari bilan ma’lumot almashinish imkoniyatiga ega bo‘ladilar (3.7 - rasm).



3.7 – rasm. Gibrid PKI arxitekturasi

Mijozlar odatda SAlarga yangi sertifikat yoki uni bekor qilish bilan bog‘liq bo‘lgan so‘rovlar bilan murojaat qiladilar. Ushbu so‘rovlar odatda maxsus protokollar orqali formallashtiriladi. Bu protokollari PKIlarni boshqarish protokollari deb atalib, ular ichida keng foydalanilayotganlari quyidagilar:

- PKCS#10;
- PKCS#7;
- Certificate Management Protocol (CMP);
- Certificate Management using CMS (CMC) ;
- Simple Certificate Enrollment Protocol (SCEP).

PKCS#10 va PKCS#7 protokollari Public Key Cryptographic Standards (PKCS) protokollar to‘plamlarining a’zosi bo‘lib, PKI kriptografiyasi uchun turli standartlarni tasniflaydi.

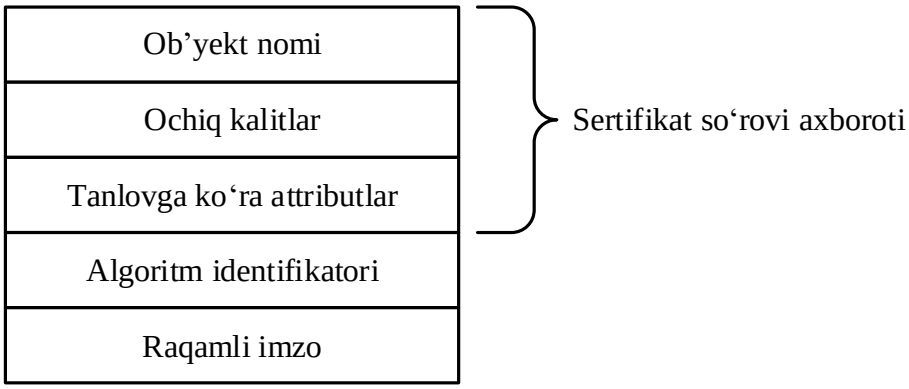
Quyidagi 1 - jadvalda umumiy holdagi PKCS standartlari va ularning nomlari keltirilgan.

3.1 – jadval

PKCS standartlari nomlari

Standart raqami	**Standart nomi**
PKCS#1	RSA kriptografik standarti
PKCS#2	Ushbu standart hozirda PKCS#1 bilan birlashtirilgan
PKCS#3	Diffi-Xelman kalitlarni almashinish standarti
PKCS#4	Ushbu standart hozirda PKCS#1 bilan birlashtirilgan
PKCS#5	Parollarga asoslangan autentifikatsiyalash standarti
PKCS#6	Kengaytirilgan Sertifikat sintaksisi standarti
PKCS#7	Kriptografik xabar sintaksisi standarti
PKCS#8	Shaxsiy kalit axboroti sintaksisi standarti
PKCS#9	Tanlangan attributlar turlari
PKCS#10	Sertifikat so‘rovi sintaksisi standarti
PKCS#11	Kriptogrfik token interfeysi standarti
PKCS#12	Shaxsiy axborotni almashinish sintaksisi standarti
PKCS#13	Elliptik egri chiziq kriptgrafiyasi standarti
PKCS#14	Psevdotasodifiy sonlar generatori standarti
PKCS#15	Kriptografik token axborot formati standarti

Quyidagi 3.8 - rasmda PKCS#10 standarti bo‘yicha mijoz tomonidan so‘rovlarni shakllantirish so‘rovi va sertifikat ko‘rinishi keltirilgan.



3.8– rasm. PKCS#10 standarti bo‘yicha sertifikat so‘rovi formati

X.509 sertifikat standarti

Kriptografiyada X.509 standarti ochiq kalitli infratuzilmalar (public key infrastructure (PKI)) va imtiyozga asoslangan boshqarish infratuzilmalari (Privilege Management Infrastructure (PMI)) uchun mo‘ljallangan. Ushbu protokol ko‘plab Internet protokollari, xususan, HTTPS uchun asos bo‘lgan SSL/TLS protokolida veb brauzerda xavfsiz kanalni qurishda foydalaniladi. Bundan tashqari u offlayn ilovalarda, masalan, elektron raqamli imzoda foydalaniladi. Ushbu sertifikat ochiq kalit, identifikatorlar (uzel nomi, organizatsiya yoki tashkilot nomi), sertifikat bergan tashkilot nomi va imzosi yoki o‘zi imzolaganligini tasdiqlovchi ma’lumotlardan iborat. Ushbu sertifikat ishonchli tashkilot tomonidan imzolanganda yoki boshqa vositalar orqali tasdiqlanganda, ushbu sertifikatni olgan odam undagi ochiq kalit bilan ikkinchi tomon bilan aloqa o‘rnatishi mumkin yoki shaxsiy kalit bilan imzolangan imzoni ushbu ochiq kalit bilan tekshirish mumkin bo‘ladi.

X.509 standarti asosidagi sertifikatlar odatda tijoriy tashkilotlar yoki ularning offislari va ochiq holda generatsiya qilinishi mumkin. Bundan tashqari ushbu sertifikatda, sertifikatni amal qilish muddati va unda qanday algoritmlardan foydalanilganligi qayd etiladi.

X.509 sertifikati International Telecommunications Union standartlash bo‘limi tomonidan aniqlangan va ASN.1 (Abstract Syntax Notation One) interfeysni ifodalash tiliga asoslangan.

Ushbu sertifikat 1988-yil 3-iyulda yaratilgan va X.500 standarti to‘plamiga kiritilgan. Sertifikatni olish uchun tashkilot quyidagi ketma-ketlikdagi amallarni bajaradi.

Talabgor o‘ziga tegishli bo‘lgan raqamli sertifikatni olishi uchun CSR (*sertificate signing request)* so‘rovini sertifikatni berish markaziga (*certificate authority*) yuboradi. CSR ni generatsiya qilishdan oldin talabgor dastlab kalit juftlarini generatsiya qiladi va maxfiy kalitni sir saqlaydi. CSR so‘rovi talabgorning ochiq kalit ma’lumoti, identifikator ma’lumotlari (masalan, domen nomi) va butunlik himoyasi (masalan, raqamli imzo) dan iborat bo‘ladi. CSR uchun eng ko‘p foydalanilgan format bu - *PKCS #10* va ba’zi veb brauzerlar tomonidan generatsiya qilinadigan *SPKAC (Signed Public Key and Challenge)* formatlaridir.

Quyida PKCS#10 formatidagi CSR so‘rovini Base64 kodidagi ko‘rinishi keltirilgan:

```
-----BEGIN CERTIFICATE REQUEST-----
MIICzDCCAbQCAQAwgYYxCzAJBgNVBAYTAkVOMQ0wCwYDVQQIDARub25lMQ0wCwYD
VQQHDARub25lMRIwEAYDVQQKDAlXaWtpcGVkaWExDTALBgNVBAsMBG5vbmUxGDAW
BgNVBAMMDyoud2lraXBlZGlhLm9yZzEcMBoGCSqGSIb3DQEJARYNbm9uZUBub25l
LmNvbTCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBAMP/U8RlcCD6E8AL
PT8LLUR9ygyygPCaSmIEC8zXGJung3ykElXFRz/Jc/bu0hxCxi2YDz5IjxBBOpB/
kieG83HsSmZZtR+drZIQ6vOsr/ucvpnB9z4XzKuabNGZ5ZiTSQ9L7Mx8FzvUTq5y
/ArIuM+FBeuno/IV8zvwAe/VRa8i0QjFXT9vBBp35aeatdnJ2ds50yKCsHHcjvtr
9/8zPVqqmhl2XFS3Qdqlsprzbgksom67ObJGjaV+fNHNQ0o/rzP//Pl3i7vvaEG
7Ff8tQhEwR9nJUR1T6Z7ln7S6cOr23YozgWVkEJ/dSr6LAopb+cZ88FzW5NszU6i
57HhA7ECAwEAAaAAMA0GCSqGSIb3DQEBBAUAA4IBAQBn8OCVOIx+n0AS6WbEmYDR
SspR9xOCoOwYfamB+2Bpmt82R01zJ/kaqzUtZUjaGvQvAaz5lUwoMdaO0X7I5Xfl
sllMFDaYoGD4Rru4s8gz2qG/QHWA8uPXzJVAj6X0olbIdLTEqTKsnBj4Zr1AJCNy
/YcG4ouLJr140o26MhwBpoCRpPjAgdYMH60BYfnc4/DILxMVqR9xqK1s98d6Ob/+
3wHFK+S7BRWrJQXcM8veAexXuk9lHQ+FgGfD0eSYGz0kyP26Qa2pLTwumjt+nBPl
rfJxaLHwTQ/1988G0H35ED0f9Md5fzoKi5evU1wG5WRxdEUPyt3QUXxdQ69i0C+7
-----END CERTIFICATE REQUEST-----
```

Talabgor CSR so‘rovini imzolovchi tashkilot ham dastlab kalit juftlarini generatsiya qiladi va CSRni imzolashda foydalaniladigan maxfiy kalitni sir tutadi. CSR so‘rovi imzolovchi tashkilot tomonidan imzolangandan so‘ng, imzo, imzo algoritmi va o‘zi haqidagi ma’lumotlarni qo‘shib X.509 sertifikatini hosil qiladi.

Imzolovchi tashkilotlarning ishonchli *root* sertifikatlari barcha ishchi tizimlarda (masalan, brauzerlarda) uzatiladi. Brauzerlarda, masalan, Internet Explorer, Firefox, Opera, Safari va Chromeda ushbu root sertifikatlar oldindan o‘rnatilgan bo‘ladi. X.509 v3 sertifikatining tuzulishi quyidagicha:

- **Certificate**
  - **Version** (versiya)
  - **Serial Number** (serial raqami)
  - **Algorithm ID** (algoritm ID si)
  - **Issuer** (sertifikat beruvchi tashkilot, emitent)
  - **Validity** (amal qilish muddati)
    - **Not Before**
    - **Not After**
  - **Subject** (sertifikat oluvchi tashkilot, istemolchi)
  - **Subject Public Key Info** (istemolchi ochiq kalit ma’lumoti)
    - **Public Key Algorithm** (ochiq kalit algoritmi)
    - **Subject Public Key** (ochiq kalit)
  - **Issuer Unique Identifier (optional)** (emitentning takrorlanmas identifikatori)

- **Subject Unique Identifier (optional)** (istemolchining takrorlanmas identifikatori)
- **Extensions (optional)** (kengaytirilgan imkoniyatlari)
  - ...
- **Certificate Signature Algorithm** (sertifikatda foydalanilgan ERI algoritmi)
- **Certificate Signature** (sertifikat qo‘yilgan imzo)

Misol tariqasida quyida *wikipedia.org* sayti uchun berilgan sertifikat berilgan:

```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            10:e6:fc:62:b7:41:8a:d5:00:5e:45:b6
    Signature Algorithm: sha256WithRSAEncryption
        Issuer: C=BE, O=GlobalSign nv-sa, CN=GlobalSign
Organization Validation CA - SHA256 - G2
        Validity
            Not Before: Nov 21 08:00:00 2016 GMT
            Not After : Nov 22 07:59:59 2017 GMT
        Subject: C=US, ST=California, L=San Francisco,
O=Wikimedia Foundation, Inc., CN=*.wikipedia.org
        Subject Public Key Info:
            Public Key Algorithm: id-ecPublicKey
                Public-Key: (256 bit)
                pub:

04:c9:22:69:31:8a:d6:6c:ea:da:c3:7f:2c:ac:a5:

af:c0:02:ea:81:cb:65:b9:fd:0c:6d:46:5b:c9:1e:

ed:b2:ac:2a:1b:4a:ec:80:7b:e7:1a:51:e0:df:f7:

c7:4a:20:7b:91:4b:20:07:21:ce:cf:68:65:8c:c6:
                    9d:3b:ef:d5:c1
                ASN1 OID: prime256v1
                NIST CURVE: P-256
        X509v3 extensions:
            X509v3 Key Usage: critical
                Digital Signature, Key Agreement
            Authority Information Access:
                CA Issuers -
URI:http://secure.globalsign.com/cacert/gsorganizationva
lsha2g2r1.crt
                OCSP -
URI:http://ocsp2.globalsign.com/gsorganizationvalsha2g2
            X509v3 Certificate Policies:
                Policy: 1.3.6.1.4.1.4146.1.20
                  CPS:
https://www.globalsign.com/repository/
                Policy: 2.23.140.1.2.2
            X509v3 Basic Constraints:
                CA:FALSE
            X509v3 CRL Distribution Points:
```

```
                Full Name:

URI:http://crl.globalsign.com/gs/gsorganizationvalsha2g2
.crl
            X509v3 Subject Alternative Name:
                DNS:*.wikipedia.org,
DNS:*.m.mediawiki.org, DNS:*.m.wikibooks.org,
DNS:*.m.wikidata.org, DNS:*.m.wikimedia.org,
DNS:*.m.wikimediafoundation.org, DNS:*.m.wikinews.org,
DNS:*.m.wikipedia.org, DNS:*.m.wikiquote.org,
DNS:*.m.wikisource.org, DNS:*.m.wikiversity.org,
DNS:*.m.wikivoyage.org, DNS:*.m.wiktionary.org,
DNS:*.mediawiki.org, DNS:*.planet.wikimedia.org,
DNS:*.wikibooks.org, DNS:*.wikidata.org,
DNS:*.wikimedia.org, DNS:*.wikimediafoundation.org,
DNS:*.wikinews.org, DNS:*.wikiquote.org,
DNS:*.wikisource.org, DNS:*.wikiversity.org,
DNS:*.wikivoyage.org, DNS:*.wiktionary.org,
DNS:*.wmfusercontent.org, DNS:*.zero.wikipedia.org,
DNS:mediawiki.org, DNS:w.wiki, DNS:wikibooks.org,
DNS:wikidata.org, DNS:wikimedia.org,
DNS:wikimediafoundation.org, DNS:wikinews.org,
DNS:wikiquote.org, DNS:wikisource.org,
DNS:wikiversity.org, DNS:wikivoyage.org,
DNS:wiktionary.org, DNS:wmfusercontent.org,
DNS:wikipedia.org
            X509v3 Extended Key Usage:
                TLS Web Server Authentication, TLS Web
Client Authentication
            X509v3 Subject Key Identifier:

28:2A:26:2A:57:8B:3B:CE:B4:D6:AB:54:EF:D7:38:21:2C:49:5C
:36
            X509v3 Authority Key Identifier:

keyid:96:DE:61:F1:BD:1C:16:29:53:1C:C0:CC:7D:3B:83:00:40
:E6:1A:7C

    Signature Algorithm: sha256WithRSAEncryption

8b:c3:ed:d1:9d:39:6f:af:40:72:bd:1e:18:5e:30:54:23:35:
         ...
```

Sertifikatning eng so‘ngi bandida (Signature Algorithm) sertifikatning undan yuqorigi bandida joylashgan ma’lumotga qo‘yilgan imzo va imzo qo‘yish uchun foydalanilgan algoritm nomi keltirilgan. Ushbu imzoni sertifikat beruvchi tashkilot o‘zining maxfiy kaliti asosida amalga oshiradi.

X.509 sertifikatlari turli kengaytmali fayllar ko‘rinishida bo‘lishi mumkin. Ammo, ko‘p hollarda ushbu kengaytmalar boshqa turdagi fayllarda masalan, shaxsiy kalitda ham foydalaniladi [28].

- .pem – (Privacy-enhanced Electronic Mail) Base64da kodlangan DER sertifikat, "-----BEGIN CERTIFICATE-----" va "-----END CERTIFICATE-----" lar bilan chegaralanadi;
- .cer, .crt, .der – odatda binar DER ko‘rinishda, ammo, Base64da kodlangan sertifikat ko‘proq ommalashgan;
- .p7b, .p7c – ma’lumotsiz PKCS#7 SignedData tuzilmasi, faqat sertifikat yoki sertifikatni bekor qilish ro‘yxatidan iborat bo‘ladi (CRL);
- .p12 – PKCS#12 formati bo‘lib, sertifikatdan (ochiq) va parol bilan himoyalangan shaxfiy kalitlardan iborat bo‘ladi;
- .pfx – PFX, PKCS#12ning oldingi ko‘rinishi bo‘lib, odatda PKCS#12 formatidagi ma’lumotlardan iborat bo‘ladi, masalan, IIS (Internet Information Services) da ushbu kengaytmali fayllar generatsiya qilinadi.

X.509 sertifikatini ko‘plab kriptografik kutubxonalar tomonilan yaratish mumkin. Quyida *BouncyCastle* kutubxonasi tomonidan ushbu sertifikatni yaratishda foydalanilgan C# kod keltirilgan [31].

```
     private void button1_Click(object sender, EventArgs e)
          {
               var KeyGenerate = new RsaKeyPairGenerator();
               KeyGenerate.Init(new   KeyGenerationParameters(new
SecureRandom(new CryptoApiRandomGenerator()), 1024));
               AsymmetricCipherKeyPair            kp            =
KeyGenerate.GenerateKeyPair();
               var gen = new X509V3CertificateGenerator();
               var certName = new X509Name("CN=CA");
               var serialNo = new BigInteger("1",10);
               gen.SetSerialNumber(serialNo);
               gen.SetSubjectDN(certName);
               gen.SetIssuerDN(certName);
               gen.SetNotAfter(DateTime.Now.AddYears(100));
               gen.SetNotBefore(DateTime.Now);
               gen.SetSignatureAlgorithm("SHA1WITHRSA");
               gen.SetPublicKey(kp.Public);
               var myCert = gen.Generate(kp.Private);
               byte[]                    result                    =
DotNetUtilities.ToX509Certificate(myCert).Export(X509ContentType
.Cert);
               FileStream  fs  =  new  FileStream("D:\\test1.crt",
FileMode.CreateNew);
               fs.Write(result, 0, result.Length);
               fs.Flush();
               fs.Close();
          }
```

Hosil bo‘lgan faylni 16 sanoq tizimi redaktori tomonidan namoyish etilganda quyidagi ko‘rinishda bo‘ladi:

```
30 82 01 8F 30 81 F9 A0  03 02 01 02 02 01 01 30
0D 06 09 2A 86 48 86 F7  0D 01 01 05 05 00 30 0D
31 0B 30 09 06 03 55 04  03 0C 02 43 41 30 20 17
0D 31 33 30 39 31 35 31  35 33 35 30 32 5A 18 0F
32 31 31 33 30 39 32 32  31 35 33 35 30 32 5A 30
0D 31 0B 30 09 06 03 55  04 03 0C 02 43 41 30 81
9F 30 0D 06 09 2A 86 48  86 F7 0D 01 01 01 05 00
03 81 8D 00 30 81 89 02  81 81 00 8D 80 B5 8E 80
8E 94 D1 04 03 6A 45 1A  54 5E 7E EE 6D 0C CB 0B
82 03 F1 7D C9 6F ED 52  02 B2 08 C3 48 D1 24 70
C3 50 C2 1C 40 BC B5 9D  F8 E8 A8 41 16 7B 0B 34
1F 27 8D 32 2D 38 BA 18  A5 31 A9 E3 15 20 3D E4
0A DC D8 CD 42 B0 E3 66  53 85 21 7C 90 13 E9 F9
C9 26 5A F3 FF 8C A8 92  25 CD 23 08 69 F4 A2 F8
7B BF CD 45 E8 19 33 F1  AA E0 2B 92 31 22 34 60
27 2E D7 56 04 8B 1B 59  64 77 5F 02 03 01 00 01
30 0D 06 09 2A 86 48 86  F7 0D 01 01 05 05 00 03
81 81 00 0A 1C ED 77 F4  79 D5 EC 73 51 32 25 09
61 F7 00 C4 64 74 29 86  5B 67 F2 3D A9 39 34 6B
3C A9 92 B8 BF 07 13 0B  A0 9B DF 41 E2 8A F6 D3
17 53 E1 BA 7F C0 D0 BC  10 B7 9B 63 4F 06 D0 7B
AC C6 FB CE 95 F7 8A 72  AA 10 EA B0 D1 6D 74 69
5E 20 68 5D 1A 66 28 C5  59 33 43 DB EE DA 00 80
99 5E DD 17 AC 43 36 1E  D0 5B 06 0F 8C 6C 82 D3
BB 3E 2B A5 F1 94 FB 53  7B B0 54 22 6F F6 4C 18
1B 72 1C
```

Yaratilgan faylni Windows operatsion tizimi vositasidan foydalanib namoyish etilganda quyidagi ko‘rinishda bo‘ladi:

```
Имя сертификата CA
Издатель        CA
Версия сертификата      3
Серийный номер  0x1
Недействителен до...    15.09.2013 15:35:00 GMT
Недействителен после... 22.09.2113 15:35:00 GMT
Цифровая подпись (SHA-1)        F9 AD 58 B5 50 3D F6 36 5E B8 89 D4 DC C8 5F CC 25 4B 9
3 A2
Цифровая подпись (SHA-256)      42 02 24 20 4E 8F 3A 3E 31 38 88 E5 C5 E7 C3 03 14 3A A
6 52 EA 78 B9 77 42 5B 99 EB 4B BA 23 82
Открытый ключ(1024 битный)              Алгоритм открытого ключа        rsaEncryption
Модуль
00: 8D 80 B5 8E 80 8E 94 D1 04 03 6A 45 1A 54 5E 7E
10: EE 6D 0C CB 0B 82 03 F1 7D C9 6F ED 52 02 B2 08
20: C3 48 D1 24 70 C3 50 C2 1C 40 BC B5 9D F8 E8 A8
30: 41 16 7B 0B 34 1F 27 8D 32 2D 38 BA 18 A5 31 A9
40: E3 15 20 3D E4 0A DC D8 CD 42 B0 E3 66 53 85 21
50: 7C 90 13 E9 F9 C9 26 5A F3 FF 8C A8 92 25 CD 23
60: 08 69 F4 A2 F8 7B BF CD 45 E8 19 33 F1 AA E0 2B
70: 92 31 22 34 60 27 2E D7 56 04 8B 1B 59 64 77 5F
Экспонента      01 00 01

Подпись         Алгоритм подписи        sha1WithRSAEncryption
Подпись
00: 0A 1C ED 77 F4 79 D5 EC 73 51 32 25 09 61 F7 00
10: C4 64 74 29 86 5B 67 F2 3D A9 39 34 6B 3C A9 92
20: B8 BF 07 13 0B A0 9B DF 41 E2 8A F6 D3 17 53 E1
30: BA 7F C0 D0 BC 10 B7 9B 63 4F 06 D0 7B AC C6 FB
40: CE 95 F7 8A 72 AA 10 EA B0 D1 6D 74 69 5E 20 68
50: 5D 1A 66 28 C5 59 33 43 DB EE DA 00 80 99 5E DD
60: 17 AC 43 36 1E D0 5B 06 0F 8C 6C 82 D3 BB 3E 2B
70: A5 F1 94 FB 53 7B B0 54 22 6F F6 4C 18 1B 72 1C
```

16 lik sanoq tizimidagi DER formatda kodlangan ma’lumotni tushunishda osonlik uchun quyidagi sodda qoidadan foydalaniladi. Birinchi yoziluvchi bayt ma’lumot turini, ikkinchi bayt

esa ma’lumot uzunligini va keyingilar ma’lumotni o‘zini ifodalaydi. Quyidagi 3.2 – jadvalda ma’lumot turlari keltirilgan.

3.2 – jadval

DER kodirovkasida mavjud ma’lumot turlari

Tip nomi	**Qisqa tasnif**	**DER kodirovkasida tipning taqdim etilishi**
SEQUENCE	Ma’lumot strukturasi tasnifi uchun foydalaniladi va turli tiplardan iborat	30
INTEGER	Butun son	02
OBJECT IDENTIFIER	Butun sonni taqdimi	06
UTCTime	Vaqt turi, biror yil uchun ikki raqamdan iborat	17
GeneralizedTime	Kengaytirilgan vaqt turi, biror yil uchun 4 ta raqamdan iborat	18
SET	Turli tipdagi ma’lumot strukturalarining tasnifi	31
UTF8String	Qator ma’lumotlari tavsifi	0S
NULL	NULL qiymat	05
BIT STRING	Bitlar ketma-ketligini saqlash uchun tip	03

Masalan, 02 03 01 00 01 DER kodirovkasidagi ma’lumotning tasnifiy quyidagicha:

02 – butun son, 03 – ma’lumot uzunligi, ya’ni 3 bayt va ma’lumotning o‘zi 16 lik sanoq tizimda 010001=65537 ga teng.

Ochiq kalitlar infratuzilmasidan foydalanuvchi xizmatlar

Hozirda aynan PKI ga asoslangan ko‘plab protokollar amalda global tarmoq bo‘ylab xavfsizlikni ta’minlashda keng qo‘llanilmoqda. Bular ichida quyidagilar alohida ahamiyat kasb etadi:

- SSL/TLS;
- S/MIME;
- IPSec.

SSL/TLS protokoli. Internet tarmog‘i bo‘ylab ma’lumotlarni uzatishda Transmission Control Protocol/Internet Protocol (TCP/IP) protokolidan foydalaniladi. Qolgan turli ilova – maxsus protokollari, masalan, HyperText Transport Protocol (HTTP), Lightweight Directory Access Protocol (LDAP) va Internet Messaging Access Protocol (IMAP) protokollari ham TCP/IP protokollaridan keng foydalanadi. Ushbu protokollari TCP/IPdan foydalanish uchun aynan TCP/IP sathida yoki undan yuqorida ishlashi talab etiladi. Biroq, aksartiyat uzatilayotgan ma’lumotlar uchun, yuboruvchi va qabul qiluvchi ma’lumotlardan tashqari, xavfsizlik talabi bajarilishi shart. Ushbu muammoni bartaraf etish uchun 1990-yillarning boshida Netscape tomonidan SSL (Secure Socket Layer) protokoli ishlab chiqildi. Ushbu protokol ma’lumot uzatish uchun maxfiy kanalni qurish uchun yaratildi. SSL amallari TCP/IP sathi va undan yuqori sathlarda amalga oshiriladi.

S/MIME protokoli. Hozirgi kunda *e-mail* aloqa almashinishning eng samarali, foydali va tezkor xizmatlaridan biri. *E-mail* xabarlari shaxsiy axborot, biznesga oid axborotdan iborat bo‘lib, Internet yoki intranet orqali yuboriladi. Internet tomonidan joriy holda ta’minlangan *e-mail* xizmatlarida odatda yuboriladigan pochta ma’lumotlari *ochiq holda* uzatiladi. Bu holda uzatilgan axborotni tutib olgan ixtiyoriy tahdidchi uni o‘qishi mumkin bo‘ladi. Bu holda muammolarni bartaraf etish uchun ochiq standart sanalgan *Secure Multipurpose Internet Mail Extension (S/MIME)* protokolidan foydalanish mumkin.

S/MIME standarti *e-mail* xavfsizligiga bag‘ishlangan bo‘lib, *e-mail* pochta uzatilishidagi uzilishlarni oldini oladi. *S/MIME* ikkita muhim xususiyatga ega:

- *Autentifikatsiyalash:* S/MIME protokoli *e-mail* pochta yuboruvchisi va qabul qiluvchisini haqiqiyligini tekshirish uchun elektron raqamli imzodan foydalanadi.
- *Maxfiylik:* S/MIME protokoli *e-mail* xabarlarni shifrlash orqali maxfiyligini ta'minlaydi.

S/MIME zaruryati. 1982-yilda Internet standarti *RFC 822* da *e-mail* formati formallashtirildi. Masalan, u ochiq xabarlarni madadlaydi va ma'lumotlar uzunligini cheklangan holda qabul qiladi (1000 ta belgi). Internet tarmog'ining keng rivojlanish natijasi o'laroq ishonchliroq va kengaytirilgan *e-mail* formati kerak edi.

1992-yilda esa Internet Engineering Task Force (IETF) tomonidan yangi *e-mail* formati standarti ishlab chiqildi. Ushbu standart Multipurpose Internet Mail Extensions (MIME) nomi bilan ma'lum. Ushbu standart ASCII belgilari sanalmagan *e-mail* xabari formati tasnifini aniqlaydi. MIME formati odatiy *e-mail* da matnni, grafikni va audioni o'z ichiga oladi.

Bundan tashqari MIME da xabarni xavfsizligi kafolatlanmagan. *MIME* formatida *e-mail* xabar xavfsizligini ta'minlash uchun *RSA Data Security Inc.* tomonidan Secure Multipurpose Internet Mail Extensions (S/MIME)ning 2.0 versiyasi ishlab chiqildi. S/MIME protokoli o'zida kriptografik himoya usullarini qo'llash orqali xavfsizlik ta'minotini amalga oshiradi. Yuboruvchi xabarni raqamli imzolaydi. Qo'yilgan imzo xabarni autentifikatsiyalash vazifasini bajaradi. Yuboruvchi shuningdek X.509 sertifikatining 3 versiyasiga asosida mavjud ochiq kalit bilan xabarni shifrlaydi.

S/MIME bilan foydalanilgan kriptografik algoritmlar. Ushbu protokolda quyidagi simmetrik kriptografiy algoritmlardan foydalaniladi:

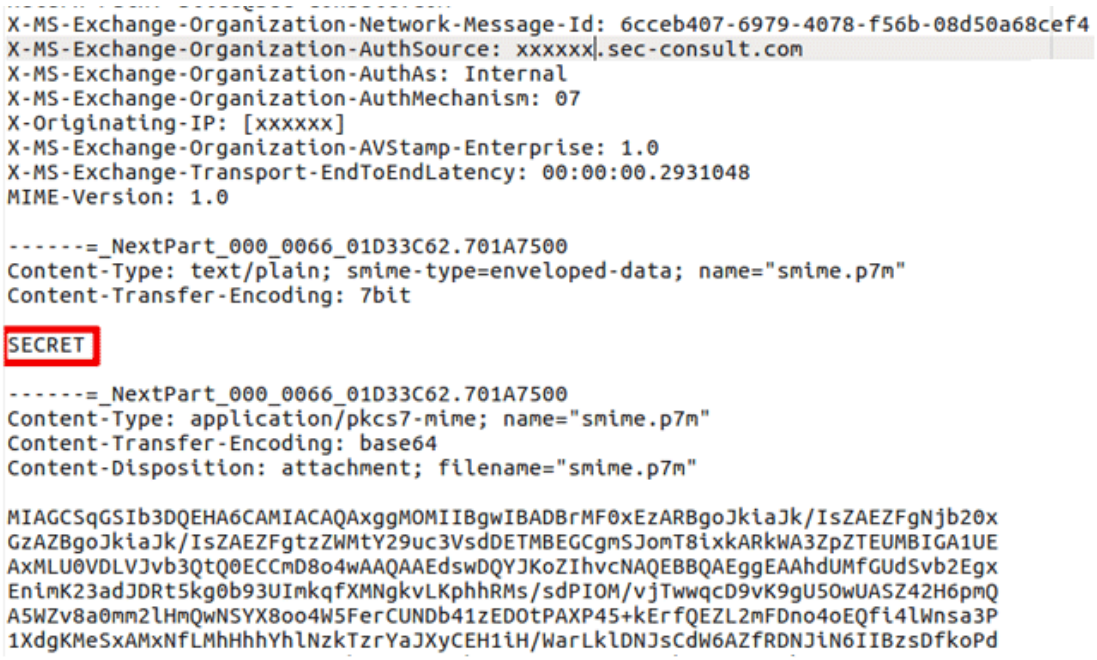
- DES;
- Triple−DES;
- RC2.

Ma'lumotni butunligini ta'minlash uchun SHA1 xesh funksiyasidan foydalaniladi.

Raqamli imzoni shakllantirishda S/MIME protokoli PKCS#7 formatidan foydalaniladi. S/MIME protokolining afzalligi quyidagilardan iborat:

- S/MIME protokoli transport MIME ma'lumotlari bilan ham foydalanilishi mumkin, HTTP;
- S/MIME protokoli xavfsizlikning asos xususiyatlari bo'lgan, butunlik, maxfiylik va autentifikatsiyani ta'minlaydi;
- S/MIME protokoli moslashuvchan bo'lib, turli *e-mail* ilovalardan foydalanuvchi mijozlar orasida xavfsiz ma'lumot almashinuvini ta'minlaydi;
- S/MIME protokoli qolgan turli ishlab chiqaruvchilar, Microsoft, Lotus, tomonidan elektron pochtaning xavfsiz vositasi ekanligi tasdiqlangan.

Ushbu protokol ko'plab posta mijozlari, Microsoft Outlook, Mozilla, The Bat! va h. lar tomonidan keng foydalaniladi. Quyidagi 9 – rasmda S/MIME protokolida imzo chekilgan xabarning fragmenti keltirilgan.



3.9 – rasm. S/MIME protokol ma’lumoti fragmenti

Nazorat savollari

1. Ochiq kalitlar infratuzilmasi va uning vazifasi.
2. Ochiq kalitlar infratuzilmasida foydalanilgan standartlar.
3. X.509 standarti.
4. Ochiq kalitlar infratuzilmasidan foydalanuvchi xizmatlar.

4-Mavzu: Xavfsiz aloqa protokollari (2 SOAT)

Reja

1. SSH protokoli.
2. SFTP va FTPS protokollari.
3. SSL va IPSec protokollari.

Secure Shell protokoli. SSH protokoli aloqa tarmog‘ida, masofadan turib amal bajarish, ikki tarmoq foydalanuvchisi orasida xavfsiz kanal hosil qilish uchun foydalaniladigan kriptografik tarmoq protokolidir. Ushbu algoritm xavfsiz tarmoq orqali maxfiy aloqani tashkil etish uchun foydalaniladi va bunda SSH kliyent va SSH server orasida xavfsiz kanal hosil qilinadi. Ushbu protokolning ikki SSH-1 va SSH-2 variantlari mavjud.

Ushbu protokol Unix yoxud LINUX sistemalariga resurslarga murojaatni amalga oshirishda foydalaniladigan asosiy yutilitalardan sanalib, WINDOWS operatsion tizimi foydalanuvchilari uchun ham moslashtirilgan. Ushbu protokol Telnet yoki boshqa xavfsiz bo‘lmagan protokollar (Bekreley rsh, rexec, rlogin) o‘rnini bosish maqsadida ishlab chiqilgan. Ushbu protokolda shifrlashdan foydalanish orqali ma’lumotning butunligi va konfidensialligini ta’minlash amalga oshirilgan (Lekin, Edvard Snovden tomonidan bazida NSA (National Security Agency) tomonidan SSHni deshifrlash orqali ma’lumotdan yashirincha foydalanilgan deb ham aytilgan).

SSH protokoli quyidagi imkoniyatlarni beradi:

- Xavfsiz login bilan bog‘lanishni;
- Xavfsiz ma’lumot almashishni ochiq (ishonchsiz) kanal orqali amalga oshirishni ta’minlaydi.

SSH protokollari quyidagilarga asoslanadi:

- Ochiq kalitli shifrlash algoritmlariga yoki
- Raqamli sertifikatlarga yoki
- Parollarga.

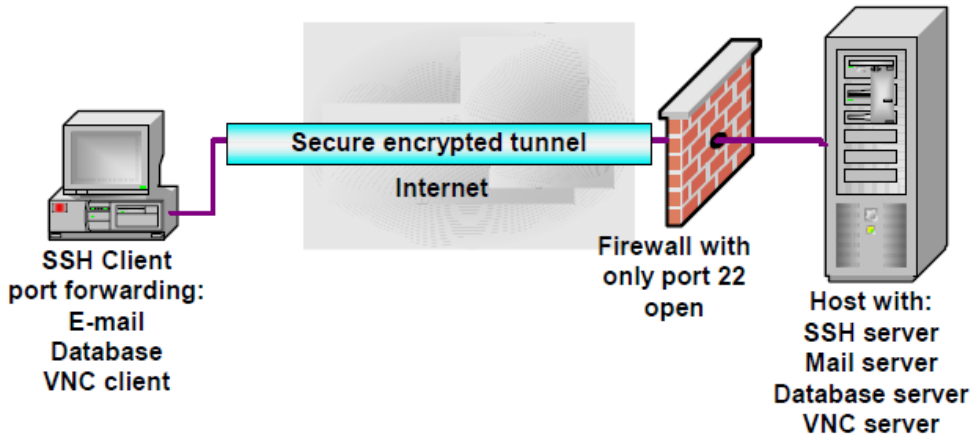
Ushbu protokolning ikki turdagi varianti, pullik va bupul turlari mavjud.

SSH vazifalari

- Xavfsiz buyruq-oynasi (command-shell)
- Xavfsiz fayl transferi
- Port forwarding

Xavfsiz Command-shell. Command shell tizimi Linux, Unix, Windows operatsion tizimlarda mavjud bo‘lib, asosan dasturiy vositalarni yuklashda va boshqa buyruqlarni bajarishda foydalaniladi. Xavfsiz command-shell ilovasi masofadan turib, buyruqlarni bajarishda, fayllarni tahrir qilishda, katalog tarkibini ko‘rishda va ma’lumot bazasini boshqarishda foydalanilishi mumkin. Ushbu tizimdan tarmoq administratori masofadan turib, o‘z vazifalarini bajarishda, xizmatlarni boshqarishda va boshqa amallarni bajarishda foydalanishi mumkin. Bunda barcha buyruqlar xavfsiz kanal orqali yuboriladi.

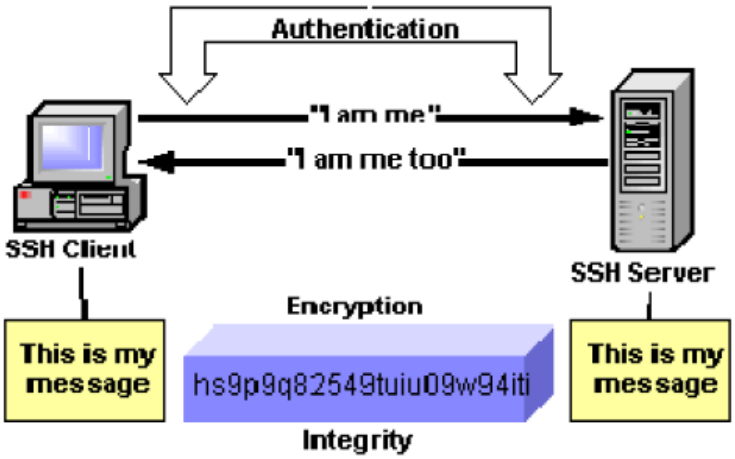
Port forwarding. SSH ning ushbu imkoniyati, TCP/IP xizmati orqali amalga oshiriluvchi, e-mail, istemolchi ma’lumoti bazasi va hak. ilovalardan xavfsiz kanal orqali foydalanish uchun zamin yaratadi. Ushbu xizmat ba’zida tunellash kabi xizmatni amalga oshirib, TCP/IP ilovalarini xavfsiz kanal orqali amalga oshiradi. Port forwarding xizmati o‘rnatilgandan so‘ng, himoyalangan kanal orqali bir tomondan (foydalanuvchi qism) ikkinchi tomonga (server tomonga) ma’lumot jo‘natiladi. Bunda hosil qilingan yagona himoyalangan kanal orqali ko‘plab ilovalar ma’lumotlari yuborilishi mumkin. Ba’zi ilovalarni boshqarishda buyruqlar oynasini o‘zi yetarli sanalmaydi, grafik interfeys orqali boshqarish ta’lab etiladi. Ushbu holda SSH ushbu xizmati orqali masofadagi ilova bilan kriptografik himoyalangan kanal hosil qilinadi. Bunga misol qilib, Virtual Network Client (VNC) ni misol qilib olish mumkin.



Xavfsiz fayl transferi. Secure File Transfer Protocol (SFTP) protokoli SSH protokoli asosida ishlab chiqilgan bo‘lib, bunda FTP protokolida mavjud ko‘plab zaifliklar oldi olingan. Birinchidan SFTP foydalanuvchi login/parolini va yuborilayotgan ma’lumotini shifrlab jo‘natadi. Ikkinchidan ushbu protokol SSH ning porti (22 port) orqali ishlaydi. Bundan tashqari FTP protokolida mavjud bo‘lgan Network Address Translations (NAT) muammosi uchramaydi.

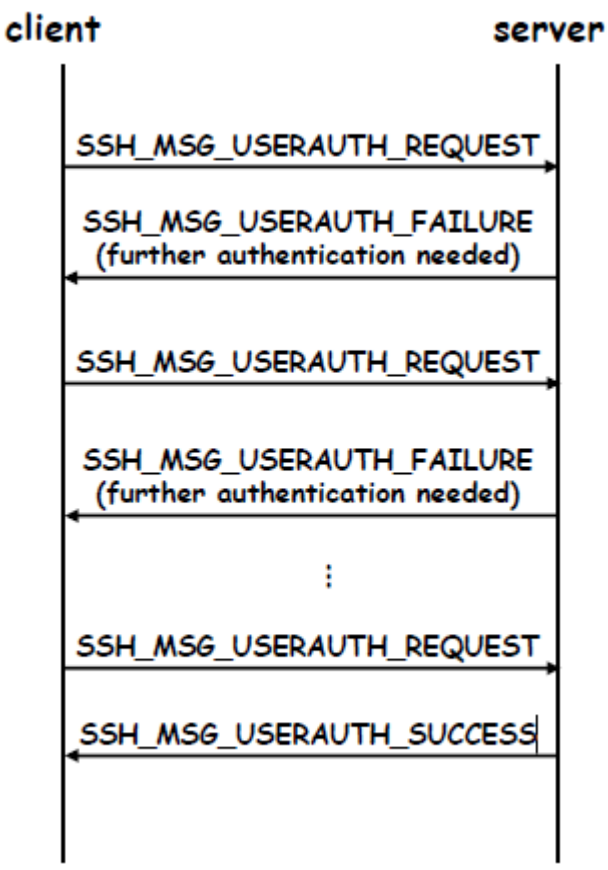
SSH ning protokol asosi

- Foydalanuvchi autentifikatsiyasi (User authentication);
- Hostga asoslangan autentifikatsiyasilash (Host authentication);
- Ma’lumotni shifrlash;
- Ma’lumot butunligi.



Foydalanuvchi autentifikatsiyasi (User authentication). Foydalanuvchini haqiqiyligini ta’minlashda SSH tizimi quyidagi turdagi autentifikatsiyalash vositalaridan foydalaniladi:

- Parol asosida;
- Ochiq kalitli shifrlash algoritmlariga asoslangan autentifikatsiyalash usullari;
- Kerberos, NTLM va boshqalar.



USERAUTH_REQUEST
- Foydalanuvchi nomi
- Xizmat nomi
- Metod nomi

USERAUTH_FAILURE
- Foydalanilayotgan autentifikatsiyalash metodlari nomlari
- bayroq holati
  - True: dastlabki autentifikatsiya muvofaqiyatli ammo yaxshiroq autentiyaikaflash zarur
  - False: oldingi autentifikatsiyalash muofaqiyatsiz

USERAUTH_SUCCESS
- Autentifikatsiya tugadi va server xizmat so‘rovini kutishni boshlaydi

Parol asosida autentifikatsiyalash. Ushbu usul boshqa autentifikatsiyalash usullariga qaraganda ko‘p uchrab, bunda parol va logini asosida foydalanuvchi haqiqiyligi ta’minlanadi. Ba’zi protokollar, FTP, Telnet protokollari login va parolni kanalda ochiq holatda yuboradi. Bu esa buzg‘unchiga tarmoqni tinglash va ularni qo‘lga kiritish imkonini beradi. Bundan farqli ravishda SSH protokolida login va parol tarmoqda shifrlangan holatda yuboriladi.

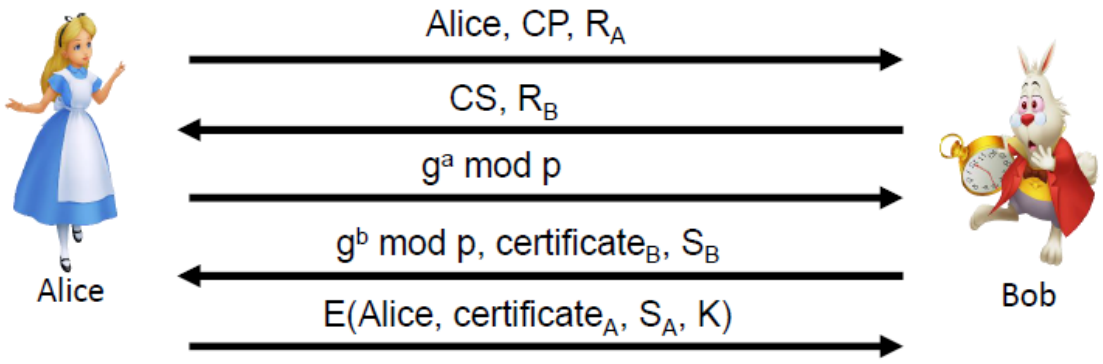
SSH_MSG_USERAUTH_REQUEST
- Foydalanuvchi ismi
- Xizmat nomi

- Parol
- FALSE (bayroq holati FALSE)
- Parol

Ushbu so‘rovga server quyidagicha javob berishi mumkin:
SSH_MSG_USERAUTH_FAILURE,
SSH_MSG_USERAUTH_SUCCESS, yoki
SSH_MSG_USERAUTH_PASSWD_CHANGEREQ

Ochiq kalitli shifrlash algoritmlariga asoslangan autentifikatsiyalash usullari. Ushbu usul SSH tizimida keng foydalaniladigan autentifikatsiyalash usullaridan biridir. Bunda kalit uzunligi 1024 bitdan 2048 bit oralig‘ida bo‘ladi. Ushbu usulda foydalanuvchi ochiq kalitlari serverda saqlanadi. Bundan tashqari foydalanuvchi maxfiy kalitga mos parolga ega bo‘lib, buzg‘unchi maxfiy kalitni bilganda ham parolsiz tizimni boshqara olmaydi.

Quyida sertifikatlarga asoslangan soddalashtirilgan SSH protokoli keltirilgan:



Bu yerda:

CP = “crypto proposed”, and CS = “crypto selected”

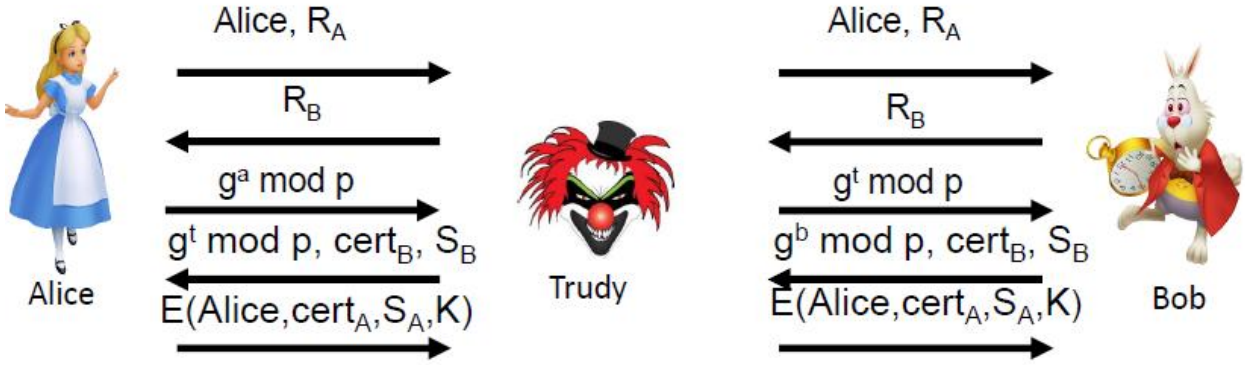
$H = h(Alice,Bob,CP,CS,R_A,R_B,g^a \bmod p,g^b \bmod p,g^{ab} \bmod p)$

$S_B = [H]_{Bob}$

$S_A = [H, Alice, certificate_A]_{Alice}$

$K = g^{ab} \bmod p$

SSH da MIM hujumi



Alisa quyidagini hisoblaydi:

$H_a = h(Alice,Bob,CP,CS,R_A,R_B,g^a \bmod p,g^t \bmod p,g^{at} \bmod p)$

Ammo Bob quyidagiga imzo chekadi:

$H_b = h(Alice,Bob,CP,CS,R_A,R_B,g^t \bmod p,g^b \bmod p,g^{bt} \bmod p)$

Ma’lumotni shifrlash

Yuborilayotgan ma’lumot boshqalar tushuna olmasligi uchun shifrlash algoritmlari yordamida shifrlanadi. Bunda SSH protokoli blokli shifrlash algoritmlari sanalgan (DES, 3DES, Blowfish, AES, va Twofish) lardan foydalanadi. Ma’lumot almashinishdan oldin ikki tomon orasida foydalanilanishi kerak bo‘lgan kripografik algoritmlar kelishib olinadi. Autentifikatsiya jarayonidan so‘ng, umumiy kalit tanlanib, ushbu kalit asosida foydalanuvchilar ma’lumotni shifrlab yuborishadi.

Ma’lumot butunligi.

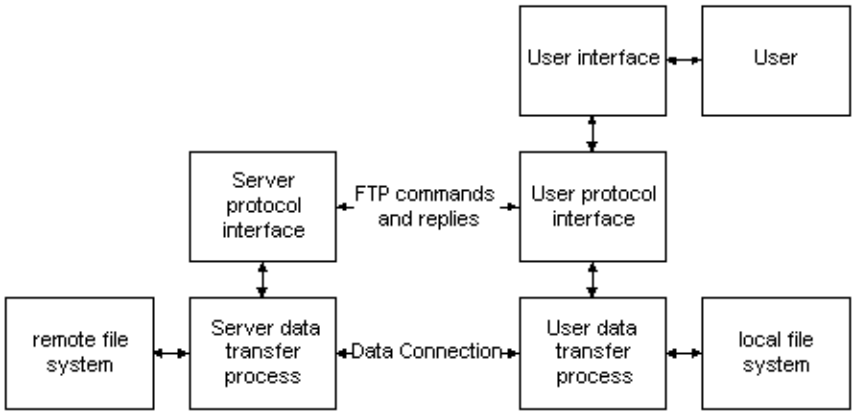
Ma’lumot uzatilish jarayonida buzg‘unchi tomonidan ma’lumotni yo‘q qilinishga urinish yoki ma’lumotni o‘zgartirish holatlari kuzatiladi. Ushbu holatlarni oldini olish va tekshirish uchun SSH tizimlarida ma’lumot butunligini ta’minlash algoritmlari foydalaniladi. SSH1 protokolida ma’lumotni butunligini tekshirishda oddiy 32 bitli CRC ma’lumotni tekshirish tizimidan foydalanilgan bo‘lsa, SSH2 tizimida esa MAS (Message Authentication Code) tizimlaridan foydalanilgan.

CRC (Cyclic Redundancy Check). Ushbu tizim ma’lumotni butunligini tekshirishda xatolikni tekshiruvchi kodlardan foydalanadi. Ushbu tizim W. Wesley Peterson tomonidan 1961-yilda ixtiro qilingan bo‘lib, 32 bitli CRC tizim Ethernet uchun foydalaniladi.

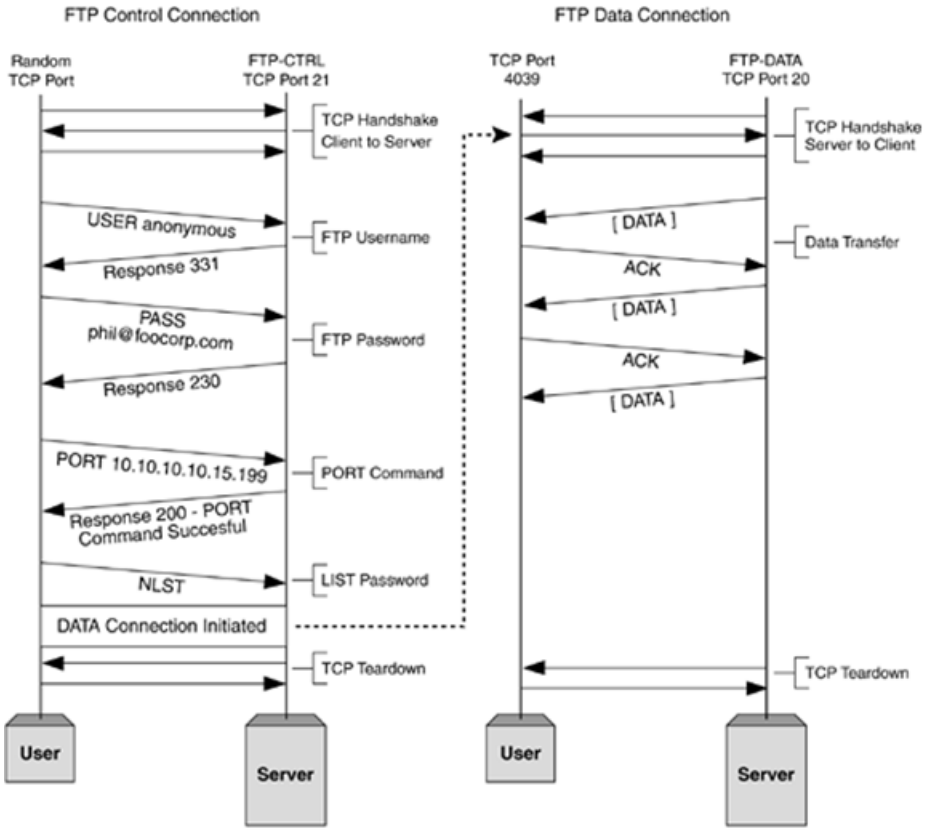
SFTP va FTPS protokollari

FTP: File Transfer Protocol

Mijoz-server muhitida ishlovchi fayllar almashinishini ta’minlovchi protokol. Login va parolga asoslangan autentifikatsiyani ta’minlaydi. Biroq server anonimlikga sozlangan bo‘lsa anonim foydalanuvchi ham ulanishi mumkin. FTP server so‘rovlarni 21 portdan qabul qiladi. FTP protokoli ikki rejimda: passiv va aktiv rejimlarda ishlashi mumkin.

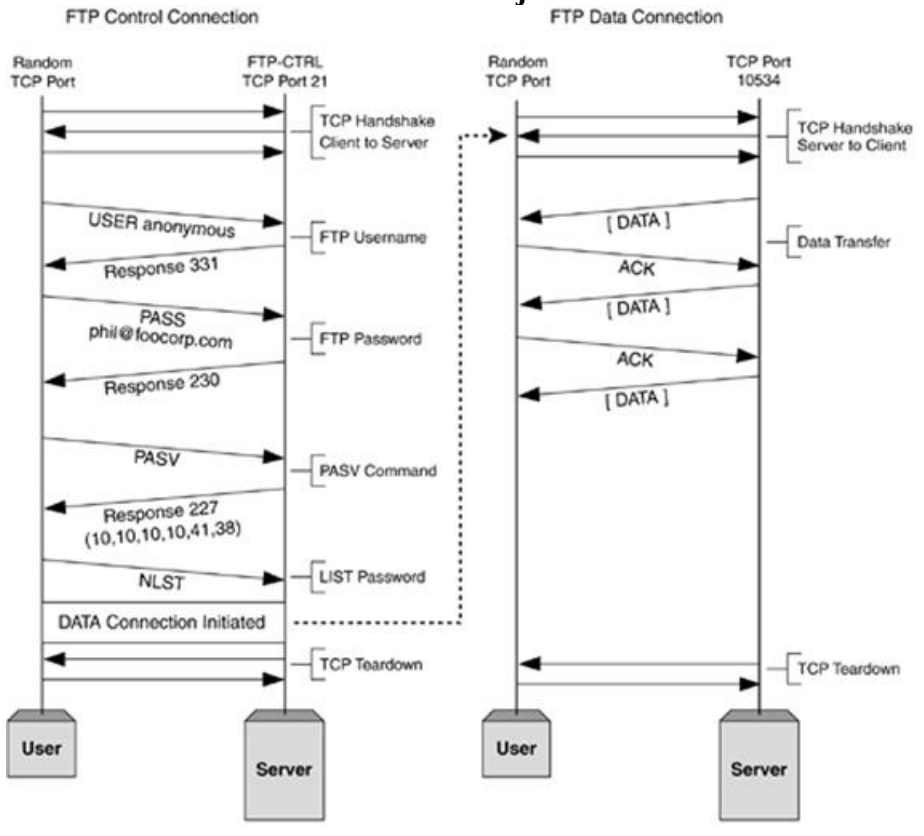


AKTIV rejim



- PORT buyrug‘i asosida mijoz o‘zining IP manzili va portini taqdim etmoqda:
  - IP=10.10.10.10 va Port=15*256+199*1=4039.
- Bu rejimda xavfsizlik muammosi mavjud bo‘lib, yuboriluvchi IP va PORT raqami foydalanuvchi bilan bir tarmoqda bo‘lgan boshqa foydalanuvchi manzillari bilan almashinishi mumkin.
- Shuning uchun passiv rejimdan foydalaniladi.

PASSIV rejim



FTP protokolining muammolari

FTP protokolining tahlili

- Qo‘pol kuch hujumi.
- PORT buyrug‘ini o‘zgartirishga asoslangan hujum.
- Paketlarni tutib olish va tahlil qilish.

Shuning uchun FTP o‘rniga xavfsiz fayl almashinish usullaridan foydalanish zarur.

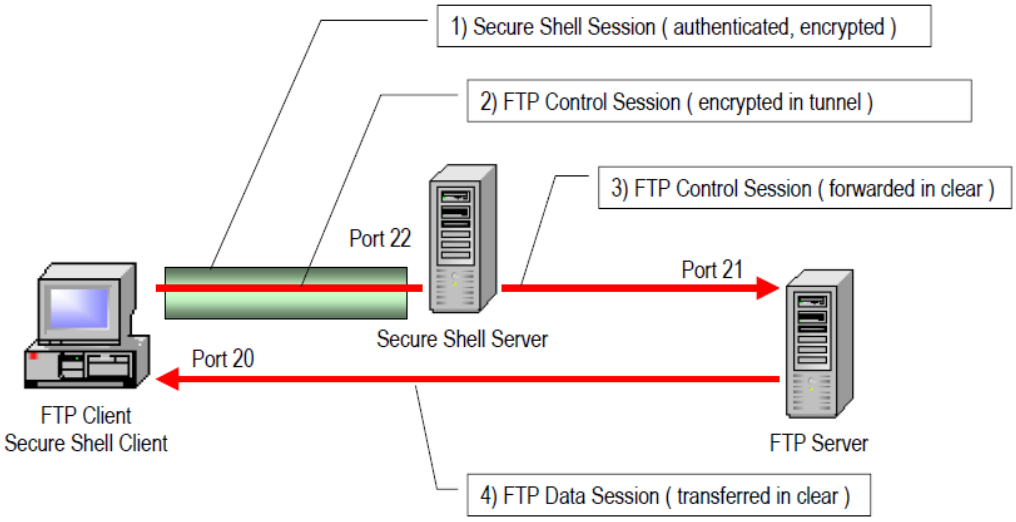
FTP dan oldin alohida shifrlash

- Yuboriluvchi faylni o‘zi yuborilishdan oldin boshqa biror dastur yordamida shifrlanadi.
- Biroq, barcha muammolarni bartaraf etmaydi.
    - Foydalanuvchi parolini himoyalamaydi;
    - Ma’lumot butunligini ta’minlamasligi mumkin.
    - Parolni himoyalanmasligi sababli, buzg‘unchi tomondagi deshifrlangan axborotga ega bo‘lishi mumkin.
    - Foydalanuvchi shishilinchda yoki bilmasdan faylni shifrlamasdan yuborishi mumkin.
    - Tomonlar turli operatsion tizimlarda bo‘lsa, foydalanilgan shifrlash dasturini topishdagi muammo bo‘lishi mumkin.

SSH orqali FTP

- Ikki tomon orasida xavfsiz tunel hosil qilinadi.
- SSH ulanish o‘rnatilgandan so‘ng FTP amalga oshiriladi.
- Shuning uchun ushbu usul *ma’lumot butunligi* va *maxfiyligini*, *server identifikatsiya tasdig‘i* va *parollar maxfiyligini* ta’minlaydi.
- FTP ikki tomon orasida zarur bo‘lgan ma’lumot formati almashinuvini ta’minlaydi.

SSH1 orqali FTP

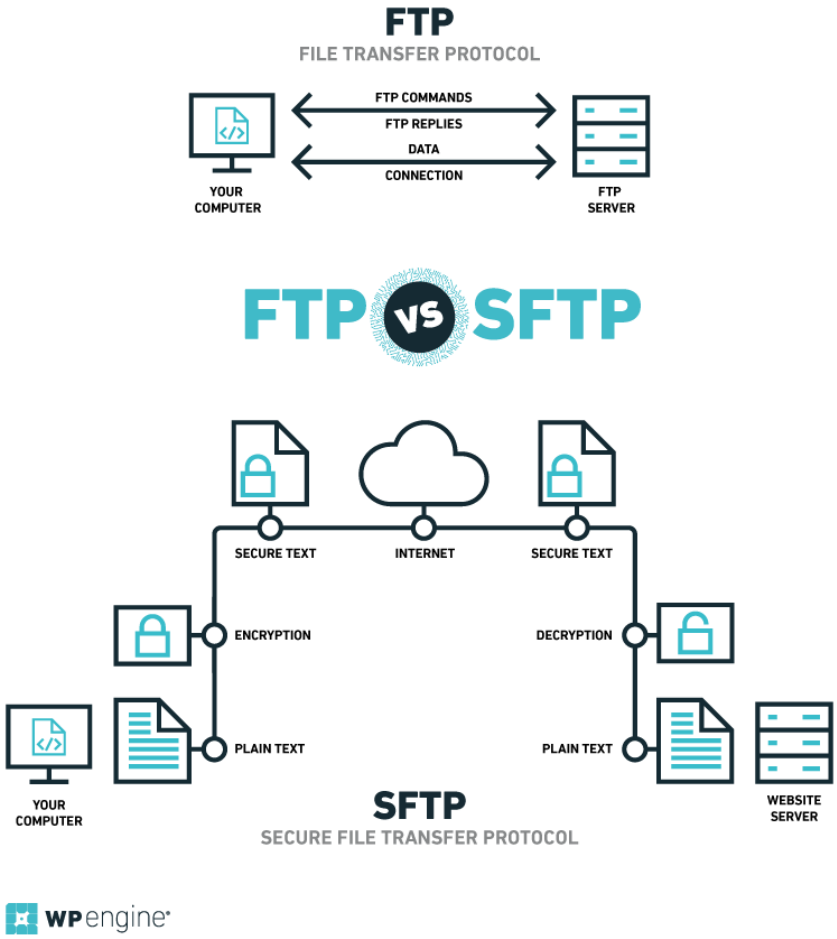


SFTP: SSH File Transfer protocol

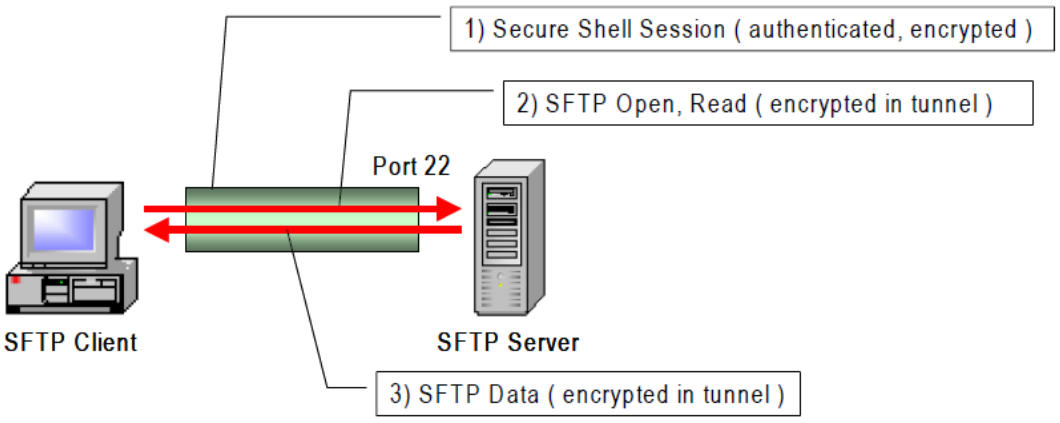
- Ushbu protokol fayllarni nazoratlashni, fayl uzatilishini va fayllar boshqaruvini ta’minlaydi.
- Ushbu protokol Internet Engineering Task Force (IETF) tomonidan SSHga qo‘shimcha sifatida ishlab chiqilgan.
- SFTP bu SSH orqali FTP emas.
- Parolni saqlash muammosini bartaraf etgan, uzutilishda ma’lumot maxfiyligi va butunligini ta’minlaydi.
- SFTP protokoli o‘zi alohida protokol bo‘lib, FTPga aloqasi yo‘q.
- SFTP protokoli FTP kabi bir xil vazifani amalga oshiradi. Biroq, xavfsizlikni ta’minlagan holda.

- SFTP protokoli 22 portda ishlaydi.

SFTP: SSH File Transfer protocol



SFTP: SSH File Transfer protocol



IPSec va FTP

- IPSec tarmoq sathida autentifikatsiya, ma’lumot butunligi va maxfiyligini ta’minlaydi.
- Barcha tizimlarda ham IPSec mavjud emas.
- Mavjud bo‘lsa ham ularni sozlash murakkab vazifa.
- Tomonlar orasida o‘rnatilgan IPSec aloqa orqali uzatilgan paketlar maxfiyligi va butunligi ta’minlangan holatda uzatiladi (FTP ga o‘xshash).

TLS (SSL) orqali FTP (FTPS)

- TLS protokoli odatda veb brauzer va server orasida ma’lumotlarni xavfsiz uzutish uchun ishlatiladi.
- FTPS protokoli FTP protokoli ustiga parol maxfiyligi va serverni tekshirish imkoniiyatini qo‘shadi.
- Shuningdek, uzatiluvchi ma’lumot maxfiyligini ham ta’minlaydi.

Ma’lumotlarni xavfsiz uzatish usullari tahlili

Usul	Parol maxfiy-ligi	Serverni tekshirish	Ma’lumot maxfiy-ligi	Ma’lumot butunligi	Avtomat-lashganligi
Alohida shifrlash	Yo‘q	Yo‘q	Bor	Bo‘lishi mukin	Yo‘q
SFTP	Bor	Bor, shaxsiy kalit	Bor	Bor	Bor
SSH orqali FTP	Bor	Bor, shaxsiy kalit	Bor	Bor	Yo‘q
IPSec orqali FTP	Bor	Bor, shaxsiy kalit, yoki SA	Bor	Bor	Bor, sozlash murakkab
TLS orqali FTP (FTPS)	Bor	Bor, shaxsiy kalit, yoki SA	Bor	Bor	Yo‘q

SSL protokoli

Transport Layer Security (TLS) dastlab yaratilgan Secure Sockets Layer (SSL) protokolining davomchisi sanalib, kompyuter tarmog‘ida aloqa xavfsizligini ta’minlash uchun yaratilgan va bir nechta kriptografik protokollar va algoritmlardan tashkil topgan. Ushbu protokolda X.509 sertifikatidan foydalanilgan bo‘lib, tomonlarni autentifikatsiyalashda assimetrik shifrlash algoritmlaridan foydalaniladi.

X.509 sertifikati. Kriptografiyada X.509 standarti ochiq kalitli infratuzilmalar (public key infrastructure (PKI)) va imtiyozga asoslangan boshqarish infratuzilmalari (Privilege Management Infrastructure (PMI)) uchun mo‘ljallangan.

Ushbu X.509 v3 sertifikatining tuzulishi quyidagicha:

- **Certificate**
  - **Version** (versiya)
  - **Serial Number** (serial raqami)
  - **Algorithm ID** (algoritm ID si)
  - **Issuer** (sertifikat beruvchi tashkilot, emitent)
  - **Validity** (amal qilish muddati)
    - **Not Before**
    - **Not After**
  - **Subject** (sertifikat oluvchi tashkilot, istemolchi)
  - **Subject Public Key Info** (istemolchi ochiq kalit ma’lumoti)

        - **Public Key Algorithm** (ochiq kalit algoritmi)
        - **Subject Public Key** (ochiq kalit)
    - **Issuer Unique Identifier (optional)** (emitentning takrorlanmas identifikatori)
    - **Subject Unique Identifier (optional)** (istemolchining takrorlanmas identifikatori)
    - **Extensions (optional)** (kengaytirilgan imkoniyatlari)
        - ...
- **Certificate Signature Algorithm** (sertifikatda foydalanilgan ERI algoritmi)
- **Certificate Signature** (sertifikat qo‘yilgan imzo)

Misol tariqasida quyida OpenSSL asosida hosil qilingan sertifikat berilgan:

$ openssl x509 -in freesoft-certificate.pem -noout -text

Certificate:
  Data:
    Version: 1 (0x0)
    Serial Number: 7829 (0x1e95)
    Signature Algorithm: md5WithRSAEncryption
    Issuer: C=ZA, ST=Western Cape, L=Cape Town, O=Thawte Consulting cc,
        OU=Certification Services Division,
        CN=Thawte Server CA/emailAddress=server-certs@thawte.com
    Validity
      Not Before: Jul 9 16:04:02 1998 GMT
      Not After : Jul 9 16:04:02 1999 GMT
    Subject: C=US, ST=Maryland, L=Pasadena, O=Brent Baccala,
        OU=FreeSoft, CN=www.freesoft.org/emailAddress=baccala@freesoft.org
    Subject Public Key Info:
      Public Key Algorithm: rsaEncryption
      RSA Public Key: (1024 bit)
        Modulus (1024 bit):
          00:b4:31:98:0a:c4:bc:62:c1:88:aa:dc:b0:c8:bb:
          33:35:19:d5:0c:64:b9:3d:41:b2:96:fc:f3:31:e1:
          66:36:d0:8e:56:12:44:ba:75:eb:e8:1c:9c:5b:66:
          70:33:52:14:c9:ec:4f:91:51:70:39:de:53:85:17:
          16:94:6e:ee:f4:d5:6f:d5:ca:b3:47:5e:1b:0c:7b:
          c5:cc:2b:6b:c1:90:c3:16:31:0d:bf:7a:c7:47:77:
          8f:a0:21:c7:4c:d0:16:65:00:c1:0f:d7:b8:80:e3:
          d2:75:6b:c1:ea:9e:5c:5c:ea:7d:c1:a1:10:bc:b8:
          e8:35:1c:9e:27:52:7e:41:8f
        Exponent: 65537 (0x10001)
  Signature Algorithm: md5WithRSAEncryption
    93:5f:8f:5f:c5:af:bf:0a:ab:a5:6d:fb:24:5f:b6:59:5d:9d:
    92:2e:4a:1b:8b:ac:7d:99:17:5d:cd:19:f6:ad:ef:63:2f:92:
    ab:2f:4b:cf:0a:13:90:ee:2c:0e:43:03:be:f6:ea:8e:9c:67:
    d0:a2:40:03:f7:ef:6a:15:09:79:a9:46:ed:b7:16:1b:41:72:
    0d:19:aa:ad:dd:9a:df:ab:97:50:65:f5:5e:85:a6:ef:19:d1:
    5a:de:9d:ea:63:cd:cb:cc:6d:5d:01:85:b5:6d:c8:f3:d9:f7:
    8f:0e:fc:ba:1f:34:e9:96:6e:6c:cf:f2:ef:9b:bf:de:b5:22:
    68:9f

Sertifikatning eng so‘ngi bandida (Signature Algorithm) sertifikatning undan yuqorigi bandida joylashgan ma’lumotga qo‘yilgan imzo va imzo qo‘yish uchun foydalanilgan algoritm nomi keltirilgan. Ushbu imzoni sertifikat beruvchi tashkilot o‘zining maxfiy kaliti asosida amalga oshiradi.

Internet tarmoq protokollari orasida olinganda, SSL/TLS protokoli ilova (Application) sathida ishlaydi. Ushbu sathni OSI modelida yekvivalent bo‘lgan sathlarda ifodalansa, SSL/TLS 5-seans (session) sathdan boshlanadi va oltinchi (presentation) sathda ishlaydi. Sessiya sathida amalga oshirilayotgan sessiya uchun umumiy kalitni va shifrlash algoritmlarini tanlash uchun assimetrik shifrlash algoritmlari yordamida handshake (qo‘l berib ko‘rishish) jarayoni amalga oshiriladi. Shundan so‘ng oltinchi sathda hosil qilingan umumiy kalit bilan simmetrik shifrlash asosida aloqa kanallari maxfiyligi ta’minlanadi. Har ikki SSL/TLS protokol yarim transport sathida ishlaydi, ya’ni paketni shifrlashda.

TLS (yoki SSL) protokoli kliyent-server arxitekturasida ishlaydi va kliyent serverga ulunish uchun TLS aloqani o‘rnatilganligini ko‘rsatishi shart. Buning uchun odatda ikki usuldan foydalaniladi:

- TLS bog‘lanishlar uchun alohida port berish (masalan, HTTPS uchun 443 port orqali);
- Protokolga asoslangan usul.

Kliyent va server TLS bog‘lanishidan foydalanishga roziligini “qo‘l siqish” (handshake) amaliyoti orqali bildiradi. Ushbu jarayon davomida kliyent va server xavfsiz aloqani tashkil etish uchun kerakli bo‘lgan ko‘plab parametrlarni kelishib oladilar:

- Ushbu jarayon dastlab kliyent serverga TLS bog‘lanishni amalga oshirgandan so‘ng boshlanib, unda serverga foydalinilishi mumkin bo‘lgan kriptografik algoritmlarning ro‘yxatini yuboradi.
- Server qabul qilingan ro‘yxatdan kerakli algoritmlarni tanlaydi.
- Shundan so‘ng server o‘zining ochiq kalitidan tashkil topgan va ishonarli tashkilot tomonidan berilgan raqamli sertifikatini yuboradi.
- Kliyent bog‘lanishni boshlashdan oldin kelgan sertifikatni haqiqiyligini tekshiradi.
- Shundan so‘ng, kliyent maxfiy aloqani hosil qilish uchun kerakli bo‘ladigan sessiya kalitini hosil qilish uchun foydalaniladigan tasodifiy sonni serverning ochiq kaliti bilan shifrlab yuboradi.
- Qabul qilingan tasodifiy son orqali har ikkala tomon shifrlash va deshifrlash uchun kerakli bo‘lgan kalitni hosil qiladi.

Natijada himoyalangan aloqa hosil qilinadi va aloqa tugagunga qadar ikki tomon orasidagi ma’lumot shifrlangan holda uzatiladi.

Tarmoqda ma’lumotni xavfsizligini ta’minlash muammosini hal qilishga urinishlar natijasida Netscape tomonidan SSL protokolining birinchi versiyasi ishlab chiqildi ammo ommaga foydalanish uchun tarqatilmadi. Sababi, ushbu birinchi versiyada jiddiy xavfsizlik muaamosi mavjud edi. Shundan so‘ng ushbu kamchiliklar bartaraf etilib, 1995-yilda ikkinchi versiya shaklida amalda foydalanish uchun taqdim etildi. Ushbu ikkinchi versiya ham amalda jiddiy kamchiliklarga egaligi aniqlanib, 1996-yilda SSLv3.0 versiyasi taqdim etildi.

Shundan so‘ng SSLv3.0 da mavjud kamchiliklar bartaraf etilib, 1999-yilda TLS 1.0 taqdim etildi. SSLv3.0 va TLS 1.0 o‘rtasida katta farq mavjud bo‘lmay, TLS 1.0 protokoli SSLv3.0 ga qaraganda o‘zining moslashuvchanligi bilan ajralib turdi.

TLS 1.1 protokoli 2006-yil aprel oyida ommaga taqdim etildi va oldingi versiyadan asosiy farqi, cipher-block chaining (CBC) rejimiga qarshi himoya usuli qo‘shildi.

TLS 1.2 protokoli 2008-yil avgust oyida taqdim etilib, quyidagi o‘zgarishlarni o‘zida mujassam etgan:

- MD5-SHA-1 algoritmlari SHA-256 algoritmi bilan almashtirilgan;
- Ushbu protokoldan boshlab shifrlash algoritmi sifatida AES shifrlash algoritmi qo‘shildi.
- Autentifikatsiyalash shifrlash algoritmi sifatida Galois/Counter Mode (GCM) rejimiga asoslangan AES algoritmidan foydalanila boshlandi.

Hozirgi kunda kelib, TLS 1.3 protokoli ustida ishlar amalga oshirilmoqda va unda odingi versiyalarga mavjud zaif algoritmlar almashtirilishi ko‘zga tutulmoqda.

TLS/SSL protokolida foydalanilgan raqamli sertifikatlarni yaratuvchi, uchinchi ishonchli

tomon sifatida qatnashgan tashkilotlarning 2015-yil boshidagi ko‘rsatkichi quyida ko‘rsatilgan:

O‘rin	Tashkilot	Foydalanilishi	Bozordagi ulushi
1.	Comodo	6.6%	33.6%
2.	Symantec Group	6.5%	33.2%
3.	Go Daddy Group	2.6%	13.2%
4.	GlobalSign	2.2%	11.3%
5.	DigiCert	0.6%	2.9%

Kalit taqsimlash protokollari. Kliyent-server orasida himoyalangan kanal hosil qilinishidan oldin, ikki tomon orasida umumiy kalit hosil qilinishi va keyinchalik barcha aloqalar ushbu kalit bilan shifrlanishi zarur. Amalda quyidagi kalit taqsimlash protokollari foydalaniladi: RSA asosida ochiq kalit va maxfiy kalitdan foydalanish orqali (TLS handshake protokolida foydalanilgan TLS_RSA protokoli), Diffie-Hellman (TLS_DH), vaqtinchalik Diffie-Hellman (TLS_DHE), Elliptic Curve Diffie-Hellman (TLS_ECDH), vaqtinchalik Elliptic Curve Diffie-Hellman (TLS_ECDHE), anonim Diffie-Hellman (TLS_DH_anon), oldindan kelishilgan kalit asosida (pre-shared key (TLS_PSK)) va xavfsiz tasodifiy parol (Secure Remote Password (TLS_SRP)) asosida amalga oshiriladi.

TLS_DH_anon va TLS_ECDH_anon usullari “o‘rtada turgan odam” (Man-in-the-middle attack) hujumiga bardoshsiz bo‘lib, amalda ularning o‘rnida TLS_DHE vaTLS_ECDHE butunlay xavfsizlikni (forward secrecy) ta’minlash maqsadida foydalaniladi.

Quyidagi jadvalda protokol versiyalarida foydalanilgan kalitlarni taqsimlash protokollari keltirilgan.

Algoritmlar	SSL 2.0	SSL 3.0	TLS 1.0	TLS 1.1	TLS 1.2
RSA	Bor	Bor	Bor	Bor	Bor
DH-RSA	Yo‘q	Bor	Bor	Bor	Bor
DHE-RSA (forward secrecy)					
ECDH-RSA	Yo‘q	Yo‘q	Bor	Bor	Bor
ECDHE-RSA (forward secrecy)					
DH-DSS	Yo‘q	Bor	Bor	Bor	Bor
DHE-DSS (forward secrecy)					
ECDH-ECDSA	Yo‘q	Yo‘q	Bor	Bor	Bor
ECDHE-ECDSA (forward secrecy)					
PSK	Yo‘q	Yo‘q	Bor	Bor	Bor
PSK-RSA					
DHE-PSK (forward secrecy)					
ECDHE-PSK (forward secrecy)					
SRP	Yo‘q	Yo‘q	Bor	Bor	Bor
SRP-DSS					
SRP-RSA					
Kerberos	Yo‘q	Yo‘q	Bor	Bor	Bor
DH-ANON (insecure)	Yo‘q	Yo‘q	Bor	Bor	Bor
ECDH-ANON (insecure)	Yo‘q	Yo‘q	Bor	Bor	Bor
GOST R 34.10-94 / 34.10-2001	Yo‘q	Yo‘q	Bor	Bor	Bor

Kalit taqsimlash protokollari asosida hosil qilingan kalit bilan quyidagi simmetrik shifrlash algoritmlari yordamida ma’lumotni shifrlab jo‘natiladi. Jadvalda mashhur hujum turlariga qarshi protokolning holati keltirilgan. Bu yerda:

N/A – mavjud emas, S – xavfsiz, D – shartlarni kamaytirishga bog‘liq, L – kam darajada xavfsiz.

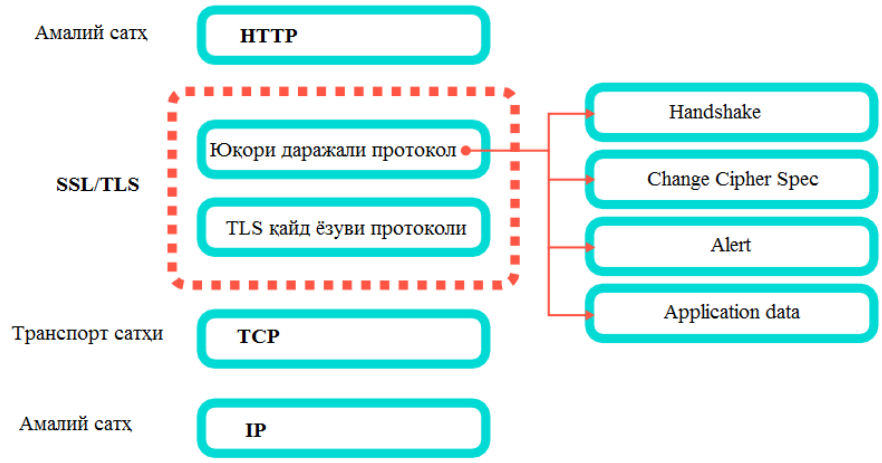
Shifrlash algoritmi			Protokol versiyasi				
Turi	Algoritm	Kalit uzunligi	SSL 2.0	SSL 3.0	TLS 1.0	TLS 1.1	TLS 1.2
Blokli shifrlash algoritmlari	AES GCM	256, 128	N/A	N/A	N/A	N/A	S
	AES CCM		N/A	N/A	N/A	N/A	S
	AES CBC		N/A	N/A	D	S	S
	Camellia GCM	256, 128	N/A	N/A	N/A	N/A	S
	Camellia CBC		N/A	N/A	D	S	S
	ARIA GCM	256, 128	N/A	N/A	N/A	N/A	S
	ARIA CBC		N/A	N/A	D	S	S
	SEED CBC	128	N/A	N/A	D	S	S
	3DES EDE CBC	112	I	I	L	L	L
	GOST 28147-89 CNT	256	N/A	N/A	S	S	S
	IDEA CBC	128	I	I	D	S	N/A
	DES CBC	56	I	I	I	I	N/A
	RC2 CBC	40	I	I	I	N/A	N/A
Oqimli shifrlash	ChaCha20-Poly1305	256	N/A	N/A	N/A	N/A	S
	RC4	128	I	I	I	I	I

Quyida esa ma’lumotni autentifikatsiyalash kodlarini SSL/TLS protokol versiyalarida ishlatilish holati keltirilgan.

Algoritm	SSL 2.0	SSL 3.0	TLS 1.0	TLS 1.1	TLS 1.2
HMAC-MD5	Ha	Ha	Ha	Ha	Ha
HMAC-SHA1	Yo‘q	Ha	Ha	Ha	Ha
HMAC-SHA256/384	Yo‘q	Yo‘q	Yo‘q	Yo‘q	Ha
AEAD	Yo‘q	Yo‘q	Yo‘q	Yo‘q	Ha
GOST 28147-89 IMIT	Yo‘q	Yo‘q	Ha	Ha	Ha
GOST R 34.11-94	Yo‘q	Yo‘q	Ha	Ha	Ha

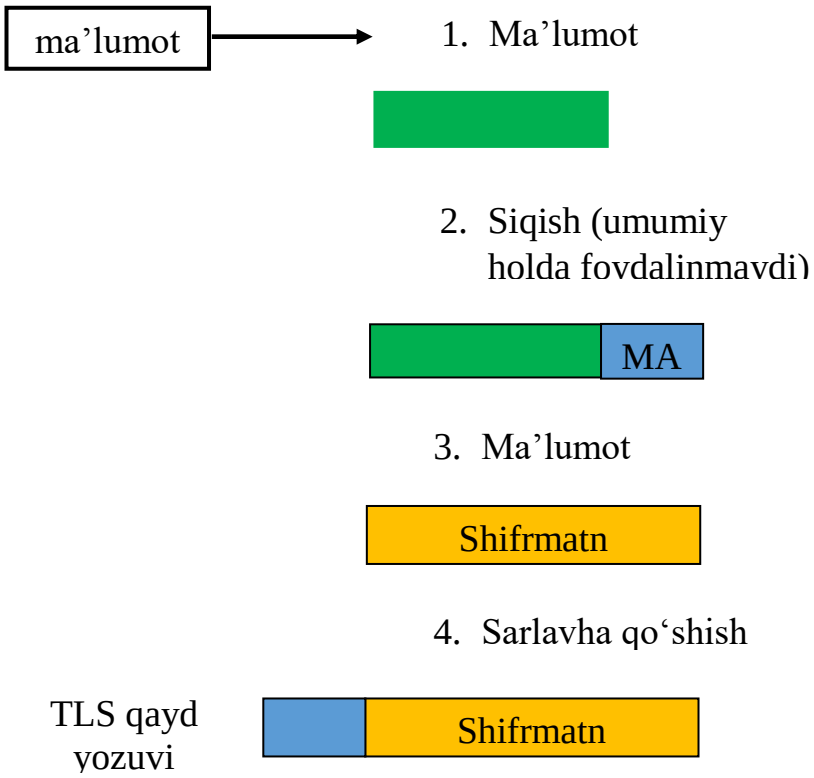
Hozirda yuqorida nomlari keltirilgan SSL/TLS protokol versiyalari amalda foydalanilmoqda va quyidagi jadvalda ularning web cahifalarda foydalanish ko‘rsatkichlari va ularning xavfsizlik xususiyati keltirilgan.

Protokol versiyasi	Web cahifalarda qo‘llab quvatlanishi	Xavfsizlik ko‘rsatkichi
SSL 2.0	14.4% (−0.5%)	Xavfsiz emas
SSL 3.0	47.3% (−3.1%)	Xavfsiz emas
TLS 1.0	99.7% (±0.0%)	Algoritm turiga bog‘liq
TLS 1.1	51.5% (+1.6%)	Algoritm turiga bog‘liq
TLS 1.2	54.5% (+1.8%)	Algoritm turiga bog‘liq

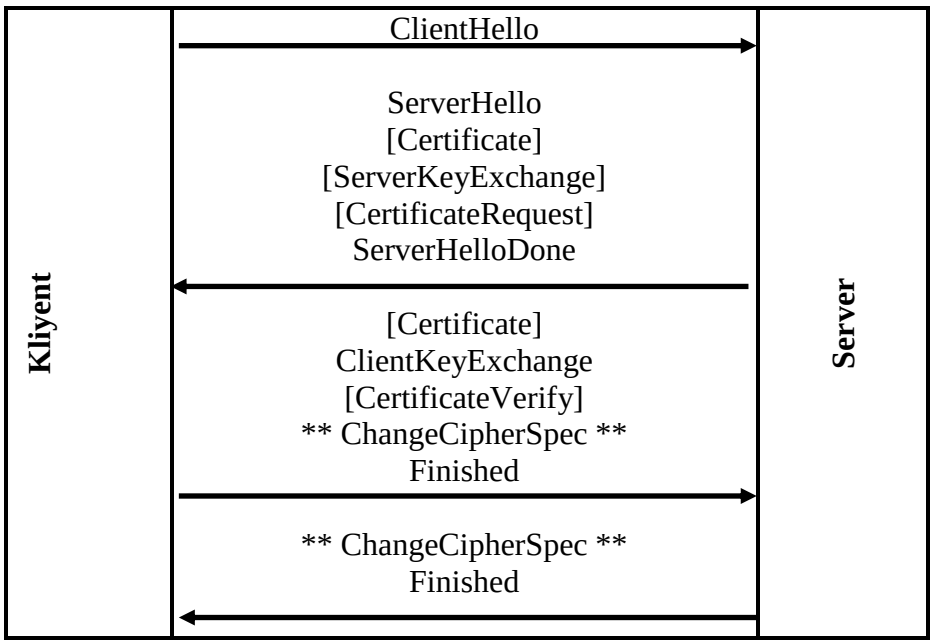


SSL/TLS sathining quyi tashkil etuvchi protokoli (TLS qayd yozuvi protokoli), dastlabki ma'lumotni fragmentlarga ajratish, sozlanishga ko'ra fragment ma'lumotni siqish, siqilgan ma'lumotga uning MAC qiymatini qo'shish, hosil bo'lgan ma'lumot juftini shifrlash algoritmi yordamida shifrlash va unga TLS qayd yozuvi sarlavhasini qo'shish amallaridan hosil bo'ladi.

Yuqori darajali protokol. Ushbu protokol TLS qayd yozuvi protokoli ustida joylashtirilgan bo'lib, u to'rtta protokoldan iborat. Har bir protokol o'zining maxsus vazifasiga ega bo'lib, ular alohida yoki birgalikda ham foydalanilishi mumkin.



Handshake protokoli. Ushbu protokol har ikki tomonda bir-birini autentifikatsiyalash, foydalaniladigan kriptografik algoritmlarni kelishish va boshqa bog'lanish parametlarini almashish imkonini beradi. Ushbu protokol kliyent va server orasida almashinuvchi to'rtta xabarlar majmuasidan iborat. Har bir xabarlar majmuasi alohida paket bo'lib yuboriladi. Quyida ushbu protokolning umumiy ko'rinishi keltirilgan.



Certificate (server)	Barcha kalit almashinish algoritmlari uchun zarur
ServerKeyExchange	Ba’zi hollarda, masalan Diffi-Xelman algoritmida zarur
CertificateRequest	Kliyent autentifikatsiyasi talab etilganda zarur.
Certificate (client)	CertificateRequest so‘roviga javob berishda zarur.
CertificateVerify	Kliyent Certificate yuborilganda zarur.

ChangeCipherSpec Protocol: ushbu protokol asosida aloqa kanali himoyalanadi.

Alert Protocol: ushbu xabar berish protokoli, barcha protokol natijalarini e’lon qilishda foydalaniladi.

Application Data Protocol: ushbu protokol ilova sathidan ma’lumotni olib, uni maxfiy kanal orqali yuborishni ta’minlaydi.

TLS qayd yozuvi formati.

Ushbu format uchta maydondan iborat bo‘lib, uning asosida yuqori darajali protokol quriladi.

- Byte 0: TLS qayd yozuvi turi.
- Bytes 1-2:TLS protokol versiyasi (major/minor).
- Bytes 3-4: qayd yozuvidagi ma’lumot uzunligi (o‘zidan tashqari). Maksimal qiymati 16384 bit yoki 16Kbit.

TLS qayd yozuvi turi	**Versiyasi**		**Ma’lumot uzunligi**		**yuqori darajali protokol**
	major	**minor**	**(bits 15..8)**	**(bits 7..0)**	

TLS qayd yozuvi turi

Hex	**Dec**	**Tyri**
0x14	20	ChangeCipherSpec
0x15	21	Alert
0x16	22	Handshake
0x17	23	Application
0x18	24	Heartbeat

Protokol versiyasi

Hex	Dec	Protokol versiyasi
0x0300	3,0	SSL 3.0
0x0301	3,1	TLS 1.0
0x0302	3,2	TLS 1.1
0x0303	3,3	TLS 1.2

Handshake protokol formati. Ushbu protokol TLS protokolida asosiy protokollarda biri sanalib, bu protokol orqali xavfsizlik parametrlari uzatiladi. Ushbu protokol orqali o‘n bir turdagi xabar uzatilashi mumkin.

Byte +0	Byte +1	Byte +2	Byte +3
22			
Versiya		Uzunlik	
Minor	Major	(bits 15..8)	(bits 7..0)
Xabar turi	Handshake ma’lumoti uzunligi		
	(bits 23..16)	(bits 15..8)	(bits 7..0)
Handshake ma’lumoti			
Xabar turi	Handshake ma’lumoti uzunligi		
	(bits 23..16)	(bits 15..8)	(bits 7..0)
Handshake ma’lumoti			

Handshake ma’lumoti uzunligi. Ushbu maydon uzunligi 3 bayt bo‘lib, faqat Handshake ma’lumoti uzunligini bildiradi, sarlavhani o‘z ichiga olmagan holda. Bitta TLS yozishmasida bir nechta Handshake ma’lumoti bo‘lishi mumkin.

Handshake protokolida xabar turi quyidagicha bo‘lishi mumkin:

Xabar turi		
Dec	Hex	Tasnif
0	0x00	HelloRequest
1	0x01	ClientHello
2	0x02	ServerHello
4	0x04	NewSessionTicke
11	0x0b	Certificate
12	0x0c	ServerKeyExchange
13	0x0d	CertificateRequest
14	0x0e	ServerHelloDone
15	0x0f	CertificateVerify
16	0x10	ClientKeyExchange
20	0x14	Finished

HelloRequest: ushbu xabar orqali server handshake protokolini qayta yuklaydi. Ushbu xabar ko‘p foydalanilmasada, agar bog‘linish uzoq vaqt davom ettirilsa, kalitni zaifligi ortadi. Shunday vaziyatda seans kalitini qaytadan hosil qilish va bog‘lanishni qayta qurish uchun foydalaniladi.

Byte +0	Byte +1	Byte +2	Byte +3
22			
Versiya		Uzunlik	
Minor	Major	4 bit	
0	Handshake ma’lumoti uzunligi		
	0 bit		

ClientHello: Handshake protokoli odatda ushbu xabardan boshlanib, ushbu xabar orqali kriptografik algoritmlar ro‘yxati, siqish usullari va kengaytmalar ro‘yxati yuboriladi. Bundan tashqari ushbu xabar, sessiyani qaytadan yuklash uchun ham ishlatiladi.

Byte +0	Byte +1	Byte +2	Byte +3
22			
Versiya		Uzunlik	
Major	Minor	Ma’lumotga bog‘liq holda	
01	Handshake ma’lumoti uzunligi		
	3 bayt		
SSL/TLS versiyasi (major/minor)		32 bitli tasodifiy kalit	
SessionId uzunligi	Max 32 bitli SessionId		
Kriptografik algoritmlar ro‘yxati			
Siqish usullari		Kengaytmalar	

ServerHello: ushbu xabar xam **ClientHello** ga o‘xshash bo‘lib, farqli tomoni kriptografik algoritmlar ro‘yxati va siqish usullari maydonidadir. Agar SessionId>0 bo‘lsa, u holda kliyent ushbu parametrdan kelajakda foydalanadi.

Byte +0	Byte +1	Byte +2	Byte +3
22			
Versiya		Uzunlik	
Major	Minor	Ma’lumotga bog‘liq holda	
02	Handshake ma’lumoti uzunligi		
	3 bayt		
SSL/TLS versiyasi (major/minor)		32 bitli tasodifiy kalit	
SessionId uzunligi	Max 32 bitli SessionId		
Kriptografik algoritmlar ro‘yxati			
Siqish usullari		Kengaytmalar	

Certificate: ushbu xabar ochiq kalit sertifikatidan tashkil topgan. Ushbu sertifikat TLS protokolida sertifikat iyerarxiyasi va PKI dan foydalanish imkonini beradi.

Byte +0	Byte +1	Byte +2	Byte +3
22			
Versiya		Uzunlik	
Major	Minor	Ma’lumotga bog‘liq holda	
11	Handshake ma’lumoti uzunligi		
	3 bayt		
Sertifikatlar ketma-ketligi uzunligi		Sertifikat uzunligi	
sertifikat	...bir nechta sertifikatlar		

ServerKeyExchange. Ushbu xabar o‘zida kalit almashinish protokollari parametrlarini saqlab, ushbu protokol asosida almashingan kalit asosida semmetrik shifrlash algoritmlari uchun kalit hosil qilinadi. Ushbu xabar tanlovga ko‘ra amalga oshiriladi. Odatda sertificate xabari orqali umumiy kalit asosida qilinadi.

Byte +0	Byte +1	Byte +2	Byte +3
22			
Versiya		Uzunlik	
Major	Minor	Ma’lumotga bog‘liq holda	
12	Handshake ma’lumoti uzunligi		
	3 bayt		
Algoritm parametrlari			

CertificateRequest. Ushbu xabar server kliyentdan autentifikatsiyadan o‘tishni talab qilganda bajariladi. Ushbu xabar umuman olganda, veb xizmatlar uchun ko‘p talab etilmasada, ba’zida talab etiladi. Bundan tashqari, ushbu xabar orqali kliyent o‘zining sertifikat versiyasi va qaysi tashkilot tomonidan berilgani haqidagi ma’lumot yuboriladi.

Byte +0	Byte +1	Byte +2	Byte +3
22			
Versiya		Uzunlik	
Major	Minor	Ma’lumotga bog‘liq holda	
13	Handshake ma’lumoti uzunligi		
	3 bayt		
Sertifikat tipi uzunligi	Sertifikat avtorlari uzunligi	Sertifikat avtorining nomi	.....

ServerHelloDone. Ushbu xabar server tomon handshake protokilini ishini tugatganda yuboriladi. Ushbu xabar o‘zida ortiqcha ma’lumotni olmaydi.

Byte +0	Byte +1	Byte +2	Byte +3
22			
Versiya		Uzunlik	
Major	Minor	Ma’lumotga bog‘liq holda	
14	Handshake ma’lumoti uzunligi		
	0 bayt		

ClientKeyExchange. Ushbu xabar serverni hosil qilinayotgan simmetrik kalit bilan ta’minlash uchun parametrlarni yuborishda foydalaniladi. Ushbu xabardani algoritmlar ServerKeyExchange xabariga yuborilgan algoritmlar bilan mos keladi.

Byte +0	Byte +1	Byte +2	Byte +3
22			
Versiya		Uzunlik	
Major	Minor	Ma’lumotga bog‘liq holda	
16	Handshake ma’lumoti uzunligi		
	3 bayt		
Algoritm parametrlari			

CertificateVerify. Ushbu xabar orqali kliyent server kalitini uning sertifikatiga mosligini tekshiradi. Ushbu xabar imzolangan xeshdan tashkil topadi. Ushbu xabar CertificateRequest so‘rovi yuborilgan holda va Certificate xabarini tasdiqlash uchun foydalaniladi.

Byte +0	Byte +1	Byte +2	Byte +3
22			
Versiya		Uzunlik	
Major	Minor	Ma’lumotga bog‘liq holda	
15	Handshake ma’lumoti uzunligi		
	3 bayt		
Imzolangan xesh			

Finished. Ushbu xabar orqali TLS muolajasi tugallanganligi va tanlangan shifrlash algoritmlari aktivlashganligini ko‘rsatiladi. Ushbu xabar oldingi barcha handshake protokol xabarlarini o‘z ichiga oladi.

Byte +0	Byte +1	Byte +2	Byte +3
22			
Versiya		Uzunlik	
Major	Minor	Ma’lumotga bog‘liq holda	

20	Handshake ma’lumoti uzunligi
	3 bayt
Imzolangan xesh	

ChangeCipherSpec protokol formati. Ushbu protokol bitta xabardan iborat bo‘lib, paketning shifrlanganligini bildiradi. TLS protokoli butun TLS qayd yozuvi ma’lumotini inkapsutsiyalaydi.

Byte +0	Byte +1	Byte +2	Byte +3
20			
Versiya		Uzunlik	
Major	Minor	1 bit	
1			

Alert protokoli. Handshaking va application turidagi protokol o‘z ishini normal holatda tugatmagan holda Alert protokoli orqali xabar beriladi. Shunga qaramasdan, ushbu xabar har bir turlagi protokol bilan birgalikda yuboriladi. Agar ushbu xabar ma’lumoti “fatal error” bo‘lsa, u holda sessiya zudlik bilan yopiladi. Agar xabar ma’lumoti “warning” bo‘lsa, u holda masofadagi foydalanuvchi talabiga ko‘ra sessiyani tugatish yoki tugatmaslik tanlanadi.

Byte +0	Byte +1	Byte +2	Byte +3
21			
Versiyasi		Uzunligi	
Major	Minor	0	2
Daraja	Tasnif		

Daraja. Ushbu maydon Alert ni darajasini ko‘rsatadi. Yuqorida aytib o‘tilganidek, ikki turdagi Alert mavjud.

Kodi	Daraja turi	Bog‘lanish holati
1	warning	Bog‘lanish yoki xavfsizlik o‘zgaruvchan bo‘lishi mumkin.
2	fatal	Bog‘lanish yoki xavfsizlik xavfli bo‘lishi mumkin, tiklib bo‘lmas xatolik yuz bergan.

Agar jarayon normal holatda o‘z ishini tugatgan taqdirda ham, biror bir daraja turi qaytariladi. Jarayonning qanday tugaganligi esa tasnif asosida aniqlanadi. Quyida tasnif jadvali keltirilgan.

Kodi	Tasnif	Daraja
0	Close notify	warning/fatal
10	Unexpected message	fatal
20	Bad record MAC	fatal
21	Decryption failed	fatal
22	Record overflow	fatal
30	Decompression failure	fatal
40	Handshake failure	fatal
41	No certificate	warning/fatal
42	Bad certificate	warning/fatal
43	Unsupported certificate	warning/fatal
44	Certificate revoked	warning/fatal
45	Certificate expired	warning/fatal

46	Certificate unknown	warning/fatal
47	Illegal parameter	fatal
48	Unknown CA (Certificate authority)	fatal
49	Access denied	fatal
50	Decode error	fatal
51	Decrypt error	warning/fatal
60	Export restriction	fatal
70	Protocol version	Fatal
71	Insufficient security	Fatal
80	Internal error	Fatal
90	User canceled	fatal
100	No renegotiation	warning
110	Unsupported extension	warning
111	Certificate unobtainable	warning
112	Unrecognized name	warning/fatal
113	Bad certificate status response	Fatal
114	Bad certificate hash value	Fatal
115	Unknown PSK identity (used in TLS-PSK and TLS-SRP)	Fatal
120	No Application Protocol	Fatal

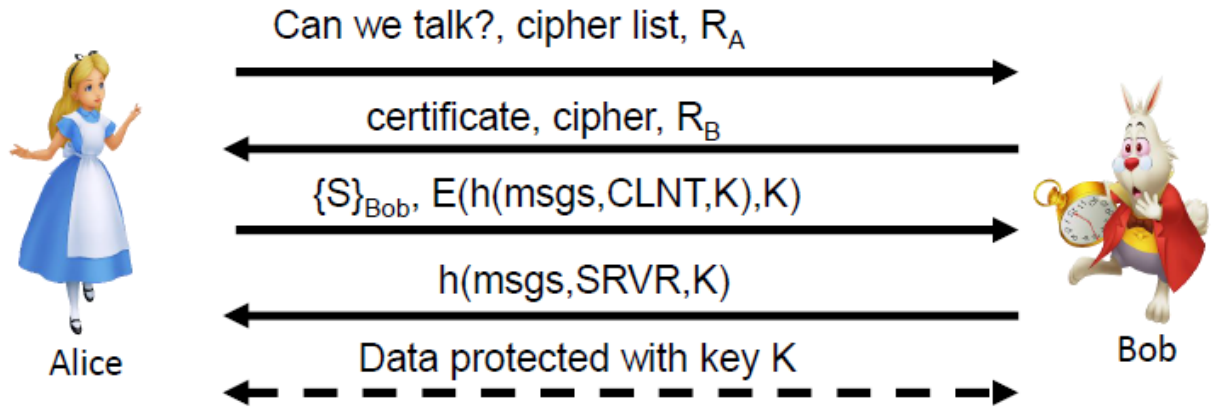
ApplicationData protokoli. Ushbu protokol ma'lumotni shifrlab jo'natuvchi protokol sanalib, ma'lumot va uning MAS qiymati birgalikda shifrlanib yuboriladi.

Byte +0	**Byte +1**	**Byte +2**	**Byte +3**
23			
Versiyasi		**Uzunligi**	
Major	**Minor**	**16 kb gacha**	
Ma'lumot			**MAS qiymati**

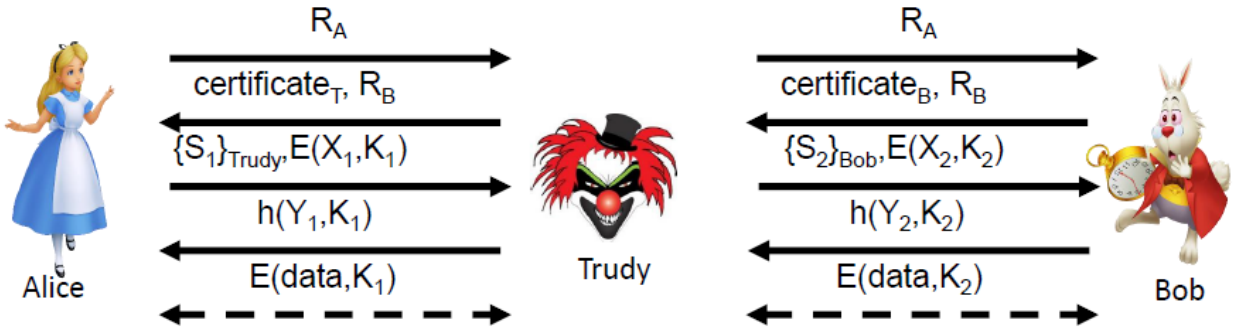
```
0000   02 00 00 00 45 00 00 98 13 ed 40 00 40 06 00 00  ....E.....@.@...
0010   7f 00 00 01 7f 00 00 01 ec 26 01 bb 43 7c ee 74  .........&..C|.t
0020   60 b5 50 0a 80 18 31 d7 fe 8c 00 00 01 01 08 0a  `.P...1.........
0030   21 62 1e 1e 21 62 1e 1e 16 03 01 00 5f 01 00 00  !b..!b......_...
0040   5b 03 01 54 9a ab 72 98 65 11 2f da 9e cf c9 db  [..T..r.e./.....
0050   6c bd 4b 4c 56 4b 0c a5 68 2b aa 60 1f 38 66 e7  l.KLVK..h+.`.8f.
0060   87 46 b2 00 00 2e 00 39 00 38 00 35 00 16 00 13  .F.....9.8.5....
0070   00 0a 00 33 00 32 00 2f 00 9a 00 99 00 96 00 05  ...3.2./........
0080   00 04 00 15 00 12 00 09 00 14 00 11 00 08 00 06  ................
0090   00 03 00 ff 01 00 00 04 00 23 00 00              .........#..

------------

Family: IP

0000   02 00 00 00                                      ....

IP protocol

0000   45 00 00 98 13 ed 40 00 40 06 00 00 7f 00 00 01  E.....@.@.......
0010   7f 00 00 01                                      ....

TCP protocol

0000   ec 26 01 bb 43 7c ee 74 60 b5 50 0a 80 18 31 d7  .&..C|.t`.P...1.
0010   fe 8c 00 00 01 01 08 0a 21 62 1e 1e 21 62 1e 1e  ........!b..!b..

------------

TLSv1 protocol

0000   16 03 01 00 5f 01 00 00 5b 03 01 54 9a ab 72 98  ...._...[..T..r.
0010   65 11 2f da 9e cf c9 db 6c bd 4b 4c 56 4b 0c a5  e./.....l.KLVK..
0020   68 2b aa 60 1f 38 66 e7 87 46 b2 00 00 2e 00 39  h+.`.8f..F.....9
0030   00 38 00 35 00 16 00 13 00 0a 00 33 00 32 00 2f  .8.5.......3.2./
0040   00 9a 00 99 00 96 00 05 00 04 00 15 00 12 00 09  ................
0050   00 14 00 11 00 08 00 06 00 03 00 ff 01 00 00 04  ................
0060   00 23 00 00                                      .#..

TLSv1 Record protocol

0000   16 03 01 00 5f                                    ....

       16             Handshake protocol type
       03 01          SSL version (TLS 1.0)
       5f             Record length (95 bytes)
```

```
TLSv1 Handshake protocol

0000   01 00 00 5b 03 01 54 9a ab 72 98 65 11 2f da 9e  ...[..T..r.e./..
0010   cf c9 db 6c bd 4b 4c 56 4b 0c a5 68 2b aa 60 1f  ...l.KLVK..h+.`.
0020   38 66 e7 87 46 b2 00 00 2e 00 39 00 38 00 35 00  8f..F.....9.8.5.
0030   16 00 13 00 0a 00 33 00 32 00 2f 00 9a 00 99 00  ......3.2./.....
0040   96 00 05 00 04 00 15 00 12 00 09 00 14 00 11 00  ................
0050   08 00 06 00 03 00 ff 01 00 00 04 00 23 00 00     ............#..

       01             ClientHello message type
       00 00 5b       Message length
       03 01          SSL version (TLS 1.0)
       54 .. b2       32-bytes random number
       00             Session Id length
       00 2e          Cipher Suites length (46 bytes, 23 suites)
       00 39 .. ff    23 2-byte Cipher Suite Id numbers
       01             Compression methods length (1 byte)
       00             Compression method (null)
       00 04          Extensions length (4 bytes)
       00 23          SessionTicket TLS extension Id
       00 00          Extension data length (0)
```

Soddalashgan holda SSL/TLS protokoli ishlash prinsipi quyidagicha:



S – dastlabki maxfiy kalit.
$K = h(S,R_A,R_B)$
“msgs” – oldingi barcha xabarlar.
CLNTva SRVR lar o‘zgarmaslar.



Secure Shell protokoli. SSH protokoli aloqa tarmog‘ida, masofadan turib amal bajarish, ikki tarmoq foydalanuvchisi orasida xavfsiz kanal hosil qilish uchun foydalaniladigan kriptografik tarmoq protokolidir. Ushbu algoritm xavfsiz tarmoq orqali maxfiy aloqani tashkil etish uchun foydalaniladi va bunda SSH kliyent va SSH server orasida xavfsiz kanal hosil qilinadi. Ushbu protokolning ikki SSH-1 va SSH-2 variantlari mavjud.

Ushbu protokol Unix yoxud LINUX sistemalariga resurslarga murojaatni amalga oshirishda foydalaniladigan asosiy yutilitalardan sanalib, WINDOWS operatsion tizimi foydalanuvchilari uchun ham moslashtirilgan. Ushbu protokol Telnet yoki boshqa xavfsiz bo‘lmagan protokollar (Bekreley rsh, rexec, rlogin) o‘rnini bosish maqsadida ishlab chiqilgan. Ushbu protokolda shifrlashdan foydalanish orqali ma’lumotning butunligi va konfidensialligini ta’minlash amalga oshirilgan (Lekin, Edvard Snovden tomonidan bazida NSA (National Security Agency) tomonidan SSHni deshifrlash orqali ma’lumotdan yashirincha foydalanilgan deb ham aytilgan).

SSH protokoli quyidagi imkoniyatlarni beradi:

- Xavfsiz login bilan bog‘lanishni;
- Xavfsiz ma’lumot almashishni ochiq (ishonchsiz) kanal orqali amalga oshirishni ta’minlaydi.

SSH protokollari quyidagilarga asoslanadi:

- Ochiq kalitli shifrlash algoritmlariga yoki
- Raqamli sertifikatlarga yoki
- Parollarga.

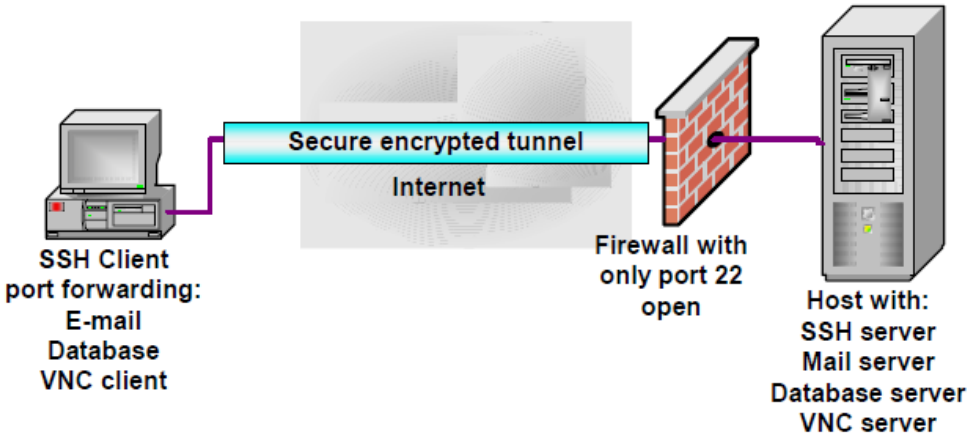
Ushbu protokolning ikki turdagi varianti, pullik va bupul turlari mavjud.

SSH vazifalari

- Xavfsiz buyruq-oynasi (command-shell)
- Xavfsiz fayl transferi
- Port forwarding

Xavfsiz Command-shell. Command shell tizimi Linux, Unix, Windows operatsion tizimlarda mavjud bo‘lib, asosan dasturiy vositalarni yuklashda va boshqa buyruqlarni bajarishda foydalaniladi. Xavfsiz command-shell ilovasi masofadan turib, buyruqlarni bajarishda, fayllarni tahrir qilishda, katalog tarkibini ko‘rishda va ma’lumot bazasini boshqarishda foydalanilishi mumkin. Ushbu tizimdan tarmoq administratori masofadan turib, o‘z vazifalarini bajarishda, xizmatlarni boshqarishda va boshqa amallarni bajarishda foydalanishi mumkin. Bunda barcha buyruqlar xavfsiz kanal orqali yuboriladi.

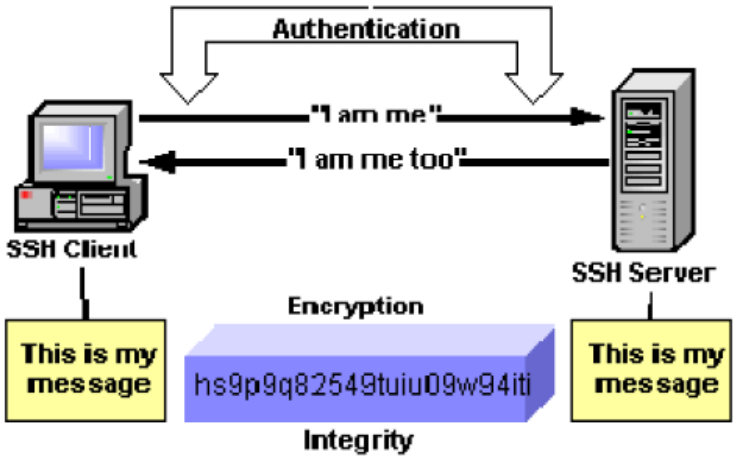
Port forwarding. SSH ning ushbu imkoniyati, TCP/IP xizmati orqali amalga oshiriluvchi, e-mail, istemolchi ma’lumoti bazasi va hak. ilovalardan xavfsiz kanal orqali foydalanish uchun zamin yaratadi. Ushbu xizmat ba’zida tunellash kabi xizmatni amalga oshirib, TCP/IP ilovalarini xavfsiz kanal orqali amalga oshiradi. Port forwarding xizmati o‘rnatilgandan so‘ng, himoyalangan kanal orqali bir tomondan (foydalanuvchi qism) ikkinchi tomonga (server tomonga) ma’lumot jo‘natiladi. Bunda hosil qilingan yagona himoyalangan kanal orqali ko‘plab ilovalar ma’lumotlari yuborilishi mumkin. Ba’zi ilovalarni boshqarishda buyruqlar oynasini o‘zi yetarli sanalmaydi, grafik interfeys orqali boshqarish ta’lab etiladi. Ushbu holda SSH ushbu xizmati orqali masofadagi ilova bilan kriptografik himoyalangan kanal hosil qilinadi. Bunga misol qilib, Virtual Network Client (VNC) ni misol qilib olish mumkin.



Xavfsiz fayl transferi. Secure File Transfer Protocol (SFTP) protokoli SSH protokoli asosida ishlab chiqilgan bo‘lib, bunda FTP protokolida mavjud ko‘plab zaifliklar oldi olingan. Birinchidan SFTP foydalanuvchi login/parolini va yuborilayotgan ma’lumotini shifrlab jo‘natadi. Ikkinchidan ushbu protokol SSH ning porti (22 port) orqali ishlaydi. Bundan tashqari FTP protokolida mavjud bo‘lgan Network Address Translations (NAT) muammosi uchramaydi.

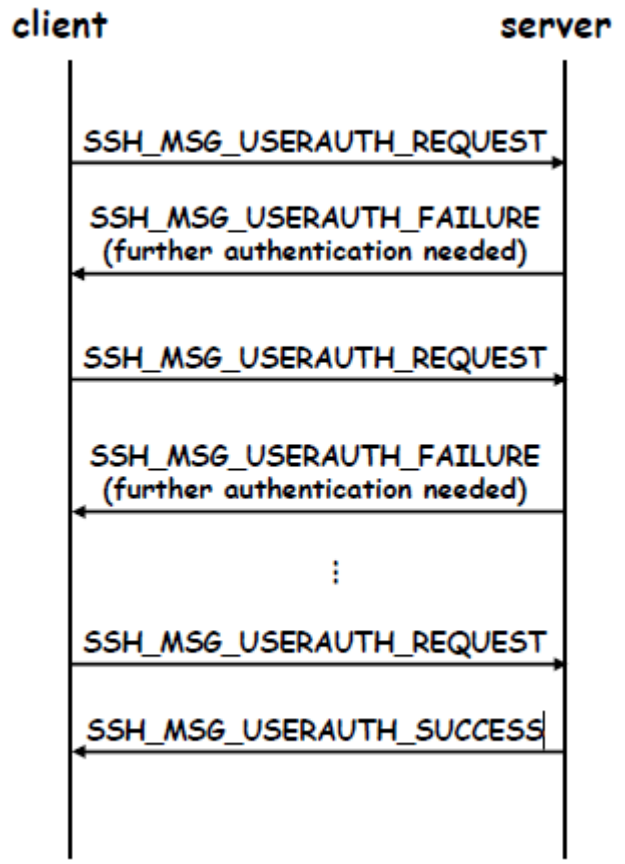
SSH ning protokol asosi

- Foydalanuvchi autentifikatsiyasi (User authentication);
- Hostga asoslangan autentifikatsiyasilash (Host authentication);
- Ma’lumotni shifrlash;
- Ma’lumot butunligi.



Foydalanuvchi autentifikatsiyasi (User authentication). Foydalanuvchini haqiqiyligini ta’minlashda SSH tizimi quyidagi turdagi autentifikatsiyalash vositalaridan foydalaniladi:

- Parol asosida;
- Ochiq kalitli shifrlash algoritmlariga asoslangan autentifikatsiyalash usullari;
- Kerberos, NTLM va boshqalar.



USERAUTH_REQUEST
- Foydalanuvchi nomi
- Xizmat nomi
- Metod nomi

USERAUTH_FAILURE
- Foydalanilayotgan autentifikatsiyalash metodlari nomlari
- bayroq holati
  - True: dastlabki autentifikatsiya muvofaqiyatli ammo yaxshiroq autentiyaikaflash zarur
  - False: oldingi autentifikatsiyalash muofaqiyatsiz

USERAUTH_SUCCESS
- Autentifikatsiya tugadi va server xizmat so‘rovini kutishni boshlaydi

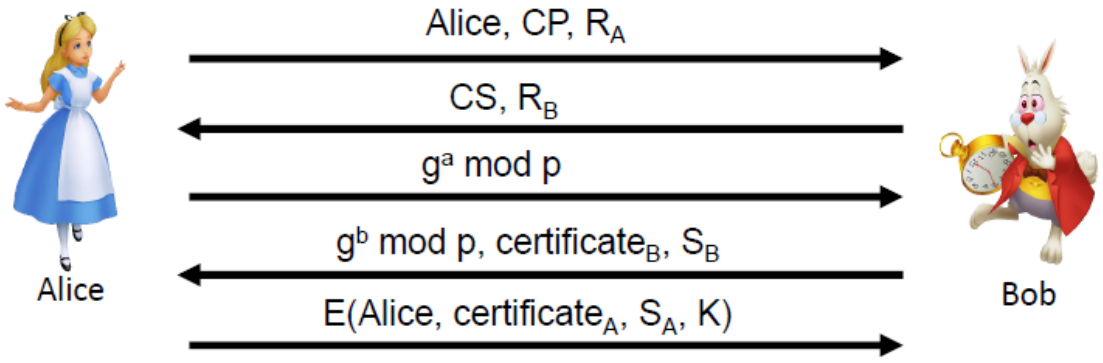
Parol asosida autentifikatsiyalash. Ushbu usul boshqa autentifikatsiyalash usullariga qaraganda ko‘p uchrab, bunda parol va logini asosida foydalanuvchi haqiqiyligi ta’minlanadi. Ba’zi protokollar, FTP, Telnet protokollari login va parolni kanalda ochiq holatda yuboradi. Bu esa buzg‘unchiga tarmoqni tinglash va ularni qo‘lga kiritish imkonini beradi. Bundan farqli ravishda SSH protokolida login va parol tarmoqda shifrlangan holatda yuboriladi.

SSH_MSG_USERAUTH_REQUEST
- Foydalanuvchi ismi
- Xizmat nomi
- Parol
- FALSE (bayroq holati FALSE)
- Parol

Ushbu so‘rovga server quyidagicha javob berishi mumkin:
SSH_MSG_USERAUTH_FAILURE,
SSH_MSG_USERAUTH_SUCCESS, yoki
SSH_MSG_USERAUTH_PASSWD_CHANGEREQ

Ochiq kalitli shifrlash algoritmlariga asoslangan autentifikatsiyalash usullari. Ushbu usul SSH tizimida keng foydalaniladigan autentifikatsiyalash usullaridan biridir. Bunda kalit uzunligi 1024 bitdan 2048 bit oralig‘ida bo‘ladi. Ushbu usulda foydalanuvchi ochiq kalitlari serverda saqlanadi. Bundan tashqari foydalanuvchi maxfiy kalitga mos parolga ega bo‘lib, buzg‘unchi maxfiy kalitni bilganda ham parolsiz tizimni boshqara olmaydi.

Quyida sertifikatlarga asoslangan soddalashtirilgan SSH protokoli keltirilgan:



Bu yerda:

CP = “crypto proposed”, and CS = “crypto selected”

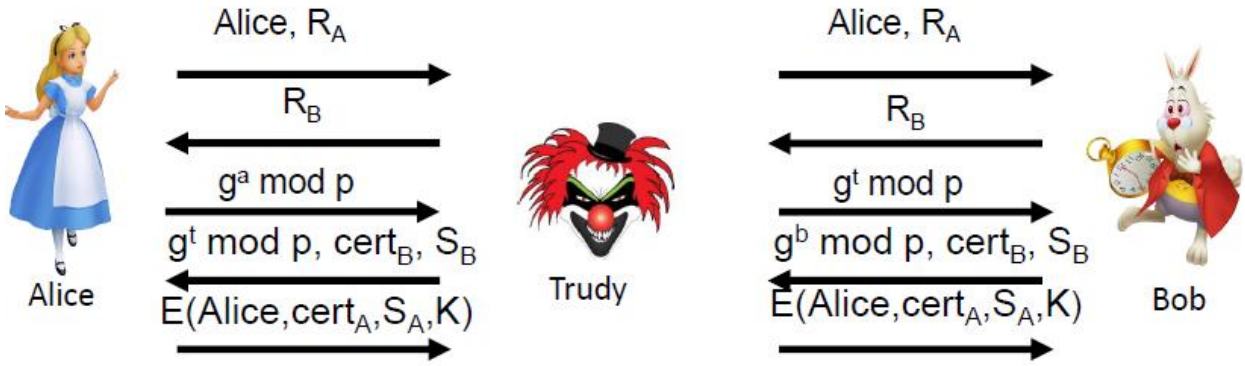
$H = h(Alice,Bob,CP,CS,R_A,R_B,g^a \bmod p,g^b \bmod p,g^{ab} \bmod p)$

$S_B = [H]_{Bob}$

$S_A = [H, Alice, certificate_A]_{Alice}$

$K = g^{ab} \bmod p$

SSH da MIM hujumi



Alisa quyidagini hisoblaydi:

$H_a = h(Alice,Bob,CP,CS,R_A,R_B,g^a \bmod p,g^t \bmod p,g^{at} \bmod p)$

Ammo Bob quyidagiga imzo chekadi:

$H_b = h(Alice,Bob,CP,CS,R_A,R_B,g^t \bmod p,g^b \bmod p,g^{bt} \bmod p)$

Host asoslangan autentifikatsiyalash (Host authentication)

Ushbu usulda foydalanuvchi hostiga asoslangan holda autentifikatsiyalash amalga oshiriladi. Agar bir nechta foydalanuvchilar bir mashinada bo‘lsa u holda ular uchun yagona host kaliti mavjud bo‘lib, autentifikatsiyalash aynan shu kalitga asoslangan holda amalga oshiriladi. Ushbu holda foydalanuvchi o‘zining shaxsiy kaliti va shaxsini ta’minlash uchun sertifikatini yuboradi. Server esa ochiq kalitni aynan shu foydalanuvchiga tegishli yoki tegishli emasligini va imzoni haqiqiyligini tekshiradi.

SSH_MSG_USERAUTH_REQUEST

- Foydalanuvchi ismi;
- Xizmat nomi;
- “hostbased”;
- Ochiq kalitli algoritm nomi;
- Mijoz hosti uchun sertifikat va ochiq kalit;
- Mijoz hosti nomi;

- Mijoz hostida foydalanuvchi ismi;
- Imzo (sessiya raqami, va hak).

Ma’lumotni shifrlash

Yuborilayotgan ma’lumot boshqalar tushuna olmasligi uchun shifrlash algoritmlari yordamida shifrlanadi. Bunda SSH protokoli blokli shifrlash algoritmlari sanalgan (DES, 3DES, Blowfish, AES, va Twofish) lardan foydalanadi. Ma’lumot almashinishdan oldin ikki tomon orasida foydalanilanishi kerak bo‘lgan kripografik algoritmlar kelishib olinadi. Autentifikatsiya jarayonidan so‘ng, umumiy kalit tanlanib, ushbu kalit asosida foydalanuvichilar ma’lumotni shifrlab yuborishadi.

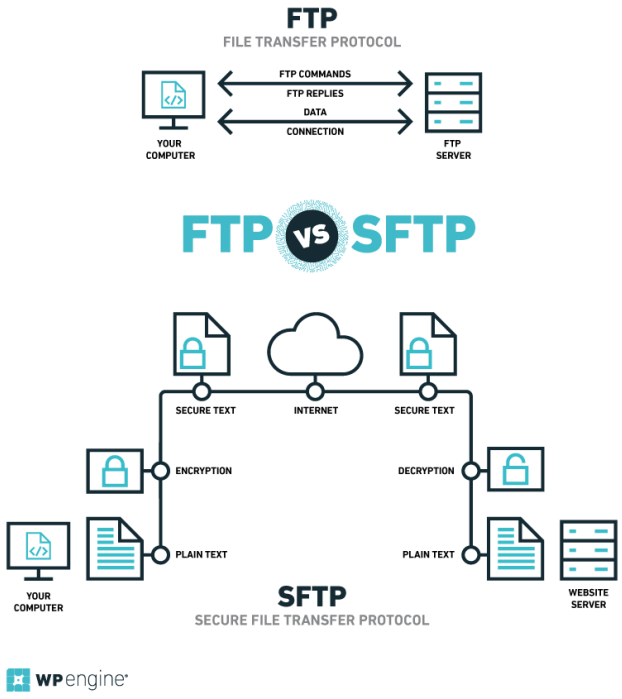
Ma’lumot butunligi.

Ma’lumot uzatilish jarayonida buzg‘unchi tomonidan ma’lumotni yo‘q qilinishga urinish yoki ma’lumotni o‘zgartirish holatlari kuzatiladi. Ushbu holatlarni oldini olish va tekshirish uchun SSH tizimlarida ma’lumot butunligini ta’minlash algoritmlari foydalaniladi. SSH1 protokolida ma’lumotni butunligini tekshirishda oddiy 32 bitli CRC ma’lumotni tekshirish tizimidan foydalanilgan bo‘lsa, SSH2 tizimida esa MAS (Message Authentication Code) tizimlaridan foydalanilgan.

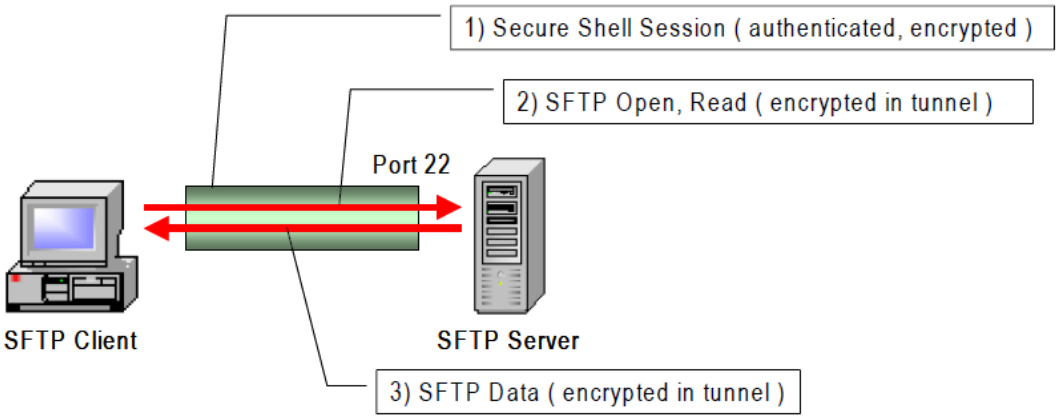
CRC (Cyclic Redundancy Check). Ushbu tizim ma’lumotni butunligini tekshirishda xatolikni tekshiruvchi kodlardan foydalanadi. Ushbu tizim W. Wesley Peterson tomonidan 1961-yilda ixtiro qilingan bo‘lib, 32 bitli CRC tizim Ethernet uchun foydalaniladi.

SFTP va FTPS protokollari

SFTP: SSH File Transfer protocol



SFTP: SSH File Transfer protocol



IPSec va FTP

- IPSec tarmoq sathida autentifikatsiya, ma’lumot butunligi va maxfiyligini ta’minlaydi.
- Barcha tizimlarda ham IPSec mavjud emas.
- Mavjud bo‘lsa ham ularni sozlash murakkab vazifa.
- Tomonlar orasida o‘rnatilgan IPSec aloqa orqali uzatilgan paketlar maxfiyligi va butunligi ta’minlangan holatda uzatiladi (FTP ga o‘xshash).

TLS (SSL) orqali FTP (FTPS)

- TLS protokoli odatda veb brauzer va server orasida ma’lumotlarni xavfsiz uzutish uchun ishlatiladi.
- FTPS protokoli FTP protokoli ustiga parol maxfiyligi va serverni tekshirish imkoniiyatini qo‘shadi.
- Shuningdek, uzatiluvchi ma’lumot maxfiyligini ham ta’minlaydi.

Ma’lumotlarni xavfsiz uzatish usullari tahlili

Usul	Parol maxfiy-ligi	Serverni tekshirish	Ma’lumot maxfiy-ligi	Ma’lumot butunligi	Avtomat-lashganligi
Alohida shifrlash	Yo‘q	Yo‘q	Bor	Bo‘lishi mukin	Yo‘q
SFTP	Bor	Bor, shaxsiy kalit	Bor	Bor	Bor
SSH orqali FTP	Bor	Bor, shaxsiy kalit	Bor	Bor	Yo‘q
IPSec orqali FTP	Bor	Bor, shaxsiy kalit, yoki SA	Bor	Bor	Bor, sozlash murakkab
TLS orqali FTP (FTPS)	Bor	Bor, shaxsiy kalit, yoki SA	Bor	Bor	Yo‘q

IPSec (Internet Protocol Security) protokoli

Internet Protocol Security (IPsec) protokol tizimlari Internet Protocol (IP) bog‘lanishlarini xavfsizligini jo‘natilayotgan har bir paketni shifrlash orqali ta’minlaydi. Ushbu protokol sessiya boshlanishida ikki tomonlama autentifikatsiyani amalga oshiradi va keyinchalik umumiy kriptografik kalitga ega bo‘linadi. IPSec protokoli ma’lumot oqimlari himoyasini ikki host orasida (host-to-host), ikki tarmoq orasida (network-to-network) va host va tarmoq orasida (network-to-host).

Ushbu protokol OSI modelining tarmoq sathida ishlaydi va tarmoq sathida autentifikatsiyalashni, ma’lumot autentifikatsiyasini, ma’lumot butunligini, ma’lumot maxfiyligini va ma’lumotni takrorlashdan himoyalashni qo‘llab quvatlaydi.

Ipsec protokollar to‘plami ochiq standart sanalib, u quyidagi turli vazifalarni bajaruvchi quyidagi funksiyalardan iborat:

Sarlavha autentifikatsiyasi (Authentication Headers (AH)) funksiya vazifasi bog‘lanish yaxlitligi, IP datagrammasi uchun ma’lumot autentifikatsiyasi va takrorlashga asoslangan hujamlarni oldini olishdan iborat. Umuman olganda IPsec ning ushbu vazifasi orqali faqat ma’lumot butunligi ta’minlanadi.

Xavfsizlik yuki inkapsulyatsiyasi (Encapsulating Security Payloads (ESP)) funksiyasi esa ma’lumotni konfidensialligini, ma’lumot butunligini va boshqa bir qator vazifalarni bajaradi.

Xavfsizlik to‘plami (Security Associations (SA)) funksiyasi vazifasi yuqoridagi ikkita funksiyaga kerakli bo‘lgan barcha kriptografik algoritmlar to‘plamini o‘zida saqlaydi. Internet Security Association and Key Management Protocol (ISAKMP) tizimi esa autentifikatsiyalash va kalit almashinish jarayoning amalga oshiradi.

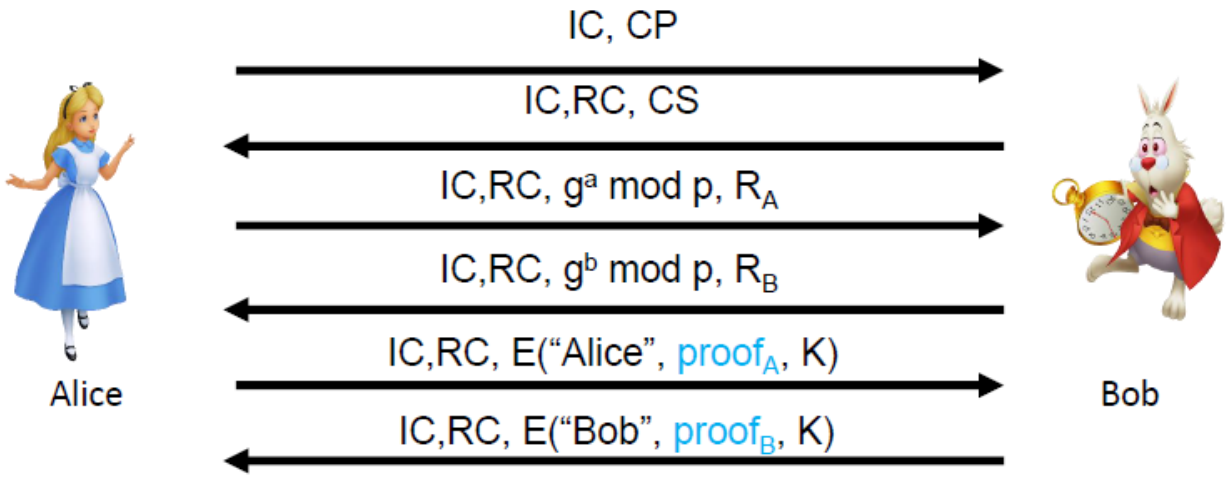
Sarlavha autentifikatsiyasi (Authentication Headers (AH)). Ushbu funksiya vazifasi bog‘lanish butunligini va IP paketi ma’lumotlari haqiqiyligini ta’minlashda foydalaniladi. Bundan tashqari u ma’lumotni qayta yuborish hujumiga qarshi himoyalashda foydalaniladi.

Xavfsizlik to‘plami (Security Associations (SA)). IPsec ning ushbu vazifasi orqali dastlab ikki tomonlama autentifikatsiya va sessiya kalitlarini almashinish amalga oshiriladi va shundan so‘ng asosiy amallar bajariladi. Bu bosqichni amalga oshirishda quyidagi usullardan foydalaniladi:

- ochiq kalitli shifrlash (haqiqiy) asosida
    - asosiy rejim
    - agressiv rejim
- ochiq kalitli shifrlash (shakllantirilgan) asosida
    - asosiy rejim
    - agressiv rejim
- ochiq kalitli imzolash asosida
    - asosiy rejim
    - agressiv rejim
- simmetrik shifrlash asosida
    - asosiy rejim
    - agressiv rejim

Ushbu bosqichda ochiq kalitli shifrlash va ularga asoslangan imzolash algoritmlaridan foydalanishdan asosiy maqsad, kalitni maxfiy saqlanishidir. Bunda maxfiy kalit faqat bir tomonda maxfiy holatda saqlanadi. Sessiya kalitini hosil qilishda esa Diffi-Xelman kalitlarni ochiq taqsimlash algoritmidan foydalanadi.

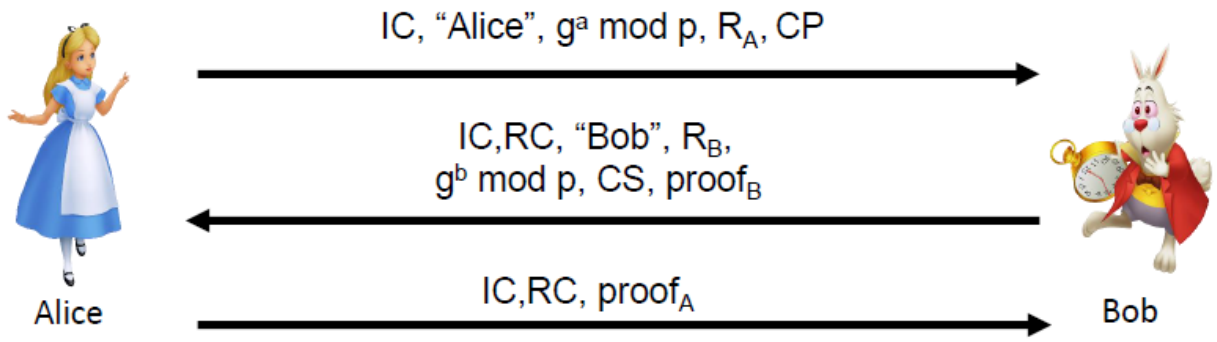
Ochiq kalitli shifrlash algoritmlari asosida imzolashga asoslangan (asosiy rejim):



Bu yerda:

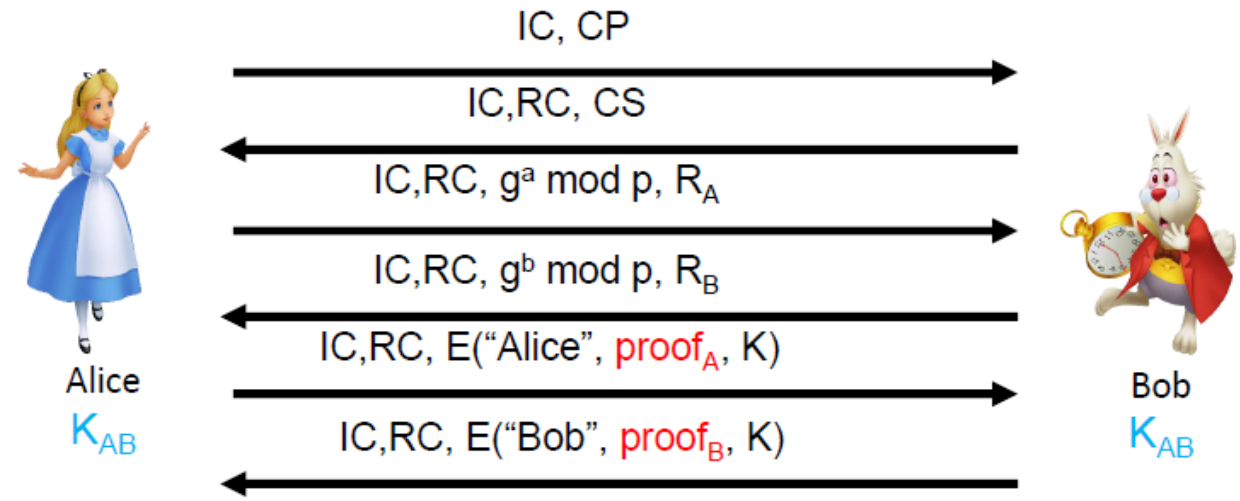
- CP = taklif etilgan kriptografik algoritmlar ro‘yxati (crypto proposed), CS = tanlangan kriptografik algoritmlar ro‘yxati (crypto selected).
- IC = initiator “cookie”, RC = responder “cookie”, Ushbu ikki parametr DDOS hujumni oldini olishda foydalaniladi.
- $K = h(IC,RC,g^{ab} \bmod p, R_A, R_B)$
- $SKEYID = h(R_A, R_B, g^{ab} \bmod p)$
- $proofA = [h(SKEYID, g^a \bmod p, g^b \bmod p, IC, RC, CP, \text{“Alice”})]_{Alice}$

Ochiq kalitli shifrlash algoritmlari asosida imzolashga asoslangan (agressiv rejim):



Asosiy rejimdan farqi foydalanuvchi nomlari yashirinmaganligidir.

Simmetrik kalitli shifrlash algoritmlari asosida imzolashga asoslangan (asosiy rejim):



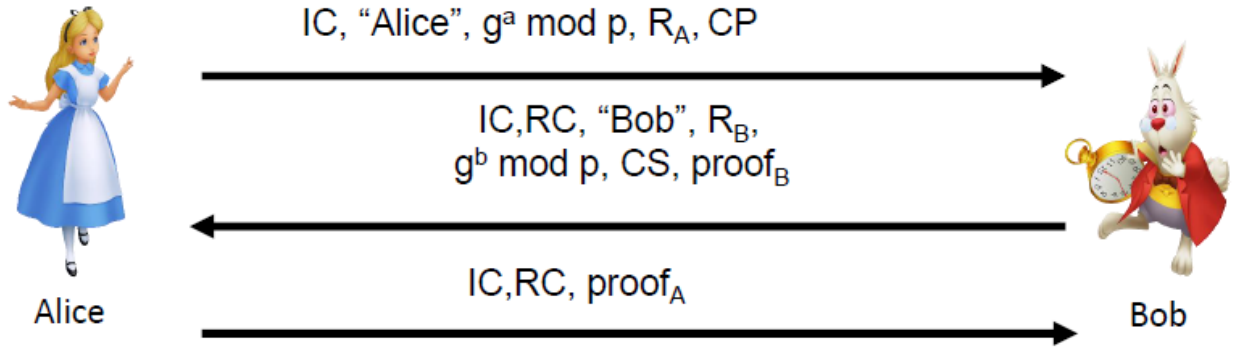
Bu yerda:

- K_{AB}= simmetrik almashingan kalit
- $K = h(IC,RC,g^{ab} \bmod p, R_A, R_B, K_{AB})$

- SKEYID = h(K, g^{ab}mod p)
- proofA= h(SKEYID,g^{a}mod p,g^{b}mod p,IC,RC,CP,“Alice”)

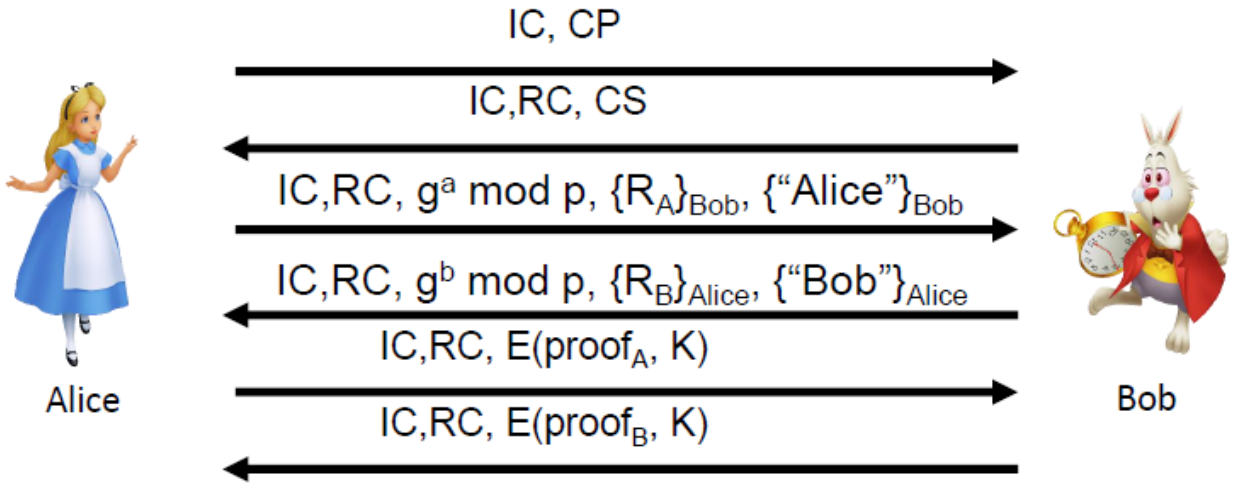
Ushbu protokolda Alis o‘zining ismini 5 chi xabarda jo‘natmoqda. Alisning identifikatori K kalit bilan shifrlanmoqda. Bob ushbu K kalitni topish uchun esa K_{AV} ni bilishi kerak. K_{AV} bilish uchun esa u Alis bilan gaplashayotganini bilishi shart.

Simmetrik kalitli shifrlash algoritmlari asosida imzolashga asoslangan (agressiv rejim):



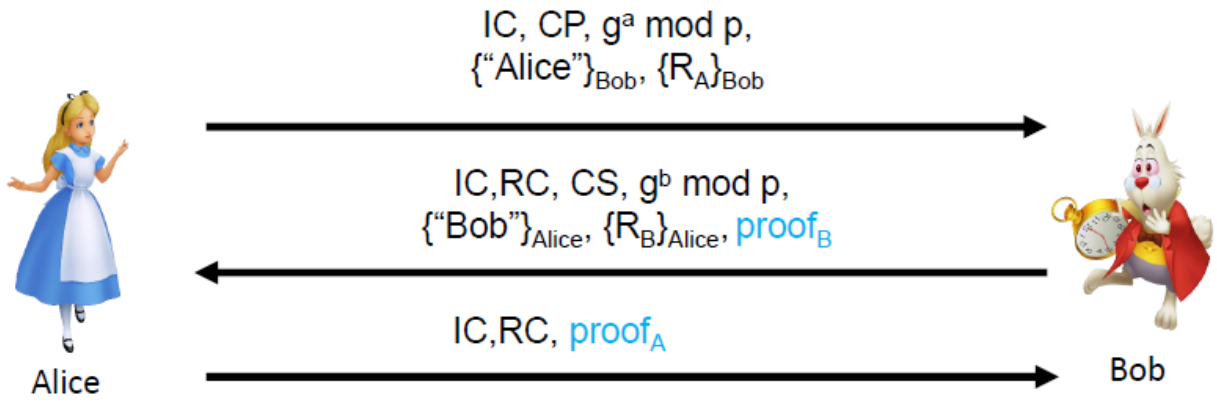
Ushbu protokolda ham foydalanuvchini nomini yashirish amalga oshirilmagan ammo asosiy rejimda uchraydigan muammo mavjud emas.

Ochiq kalitli shifrlash algoritmlariga asoslangan (asosiy rejim):



CP = crypto proposed, CS = crypto selected
IC = initiator “cookie”, RC = responder “cookie”
K = h(IC,RC,g^{ab}mod p,R_A,R_B)
SKEYID = h(R_A, R_B, g^{ab}mod p)
proofA= h(SKEYID,g^{a}mod p,g^{b}mod p,IC,RC,CP,“Alice”)

Ochiq kalitli shifrlash algoritmlari asosida imzolashga asoslangan (agressiv rejim):



K, proofA, proofB asosiy rejimdagi kabi hisoblanadi. Barcha nomlar yashiringan.

Nazorat savollari

1. SSH protokolida foydalanilgan autentifikatsiya usullari.
2. SSH protokolning vazifasi.
3. SSH protokoli xizmatidan foydalanuvchi protokollar.
4. SSH protokolida maxfiylik va butunlikni ta'minlash usullari.
5. SSH protokolida MITM xujumining ahamiyati.
6. SSL protokolida O'rtaga turgan odam hujumi.
7. SSL protokolida foydalanilgan autentifikatsiya va kalitlarni taqsimlash protokollari.
8. SFTP va FTPS orasidagi farq.
9. FTP protokolidagi asosiy kamchiliklar.
10. IPSec asosida FTP protokoli.
11. TCP/IP protokolida mavjud muammolar.
12. IPSec protokolining imkoniyatlari va vazifasi.
13. IPSec protokolida foydalanilgan autentifikatsiya usullari.

5-ma'ruza. Xavfsiz pochta protokollari (2 soat)

Reja:

1. PGP protokoli.
2. S/MIME protokoli.
3. POP protokoli.
4. POP3 protokoli.

S/MIME protokoli

Hozirgi kunda *e-mail* aloqa almashinishning eng samarali, foydali va tezkor xizmatlaridan biri. *E-mail* xabarlari shaxsiy axborot, biznesga oid axborotdan iborat bo'lib, Internet yoki intranet orqali yuboriladi. Internet tomonidan joriy holda ta'minlangan *e-mail* xizmatlarida odatda yuboriladigan pochta ma'lumotlari *ochiq holda* uzatiladi. Bu holda uzatilgan axborotni tutib olgan ixtiyoriy tahdidchi uni o'qishi mumkin bo'ladi. Bu holda muammolarni bartaraf etish uchun ochiq standart sanalgan *Secure Multipurpose Internet Mail Extension (S/MIME)* protokolidan foydalanish mumki.

S/MIME standarti *e-mail* xavfsizligiga bag'ishlangan bo'lib, *e-mail* pochta uzatilishidagi uzilishlarni oldini oladi. *S/MIME* ikkita muhim xususiyatga ega:

- *Autentifikatsiyalash:* S/MIME protokoli *e-mail* pochta yuboruvchisi va qabul qiluvchisini haqiqiyligini tekshirish uchun elektron raqamli imzodan foydalanadi.
- *Maxfiylik:* S/MIME protokoli *e-mail* xabarlarni shifrlash orqali maxfiyligini ta'minlaydi.

S/MIME zaruryati. 1982-yilda Internet standarti *RFC 822* da *e-mail* formati formallashtirildi. Masalan, u ochiq xabarlarni madadlaydi va ma'lumotlar uzunligini cheklangan holda qabul qiladi (1000 ta belgi). Internet tarmog'ining keng rivojlanish natijasi o'laroq ishonchliroq va kengaytirilgan *e-mail* formati kerak edi.

1992-yilda esa Internet Engineering Task Force (IETF) tomonidan yangi *e-mail* formati standarti ishlab chiqildi. Ushbu standart Multipurpose Internet Mail Extensions (MIME) nomi bilan ma'lum. Ushbu standart ASCII belgilari sanalmagan *e-mail* xabari formati tasnifini aniqlaydi. MIME formati odatiy *e-mail* da matnni, grafikni va audioni o'z ichiga oladi.

Bundan tashqari MIME da xabarni xavfsizligi kafolatlanmagan. *MIME* formatida *e-mail* xabar xavfsizligini ta'minlash uchun *RSA Data Security Inc.* tomonidan Secure Multipurpose Internet Mail Extensions (S/MIME)ning 2.0 versiyasi ishlab chiqildi. S/MIME protokoli o'zida kriptografik himoya usullarini qo'llash orqali xavfsizlik ta'minotini amalga oshiradi. Yuboruvchi xabarni raqamli imzolaydi. Qo'yilgan imzo xabarni autentifikatsiyalash vazifasini bajaradi. Yuboruvchi shuningdek X.509 sertifikatining 3 versiyasiga asosida mavjud ochiq kalit bilan xabarni shifrlaydi.

S/MIME bilan foydalanilgan kriptografik algoritmlar. Ushbu protokolda quyidagi simmetrik kriptografiy algoritmlardan foydalaniladi:

- DES;
- Triple−DES;
- RC2.

Ma'lumotni butunligini ta'minlash uchun SHA1 xesh funksiyasidan foydalaniladi.

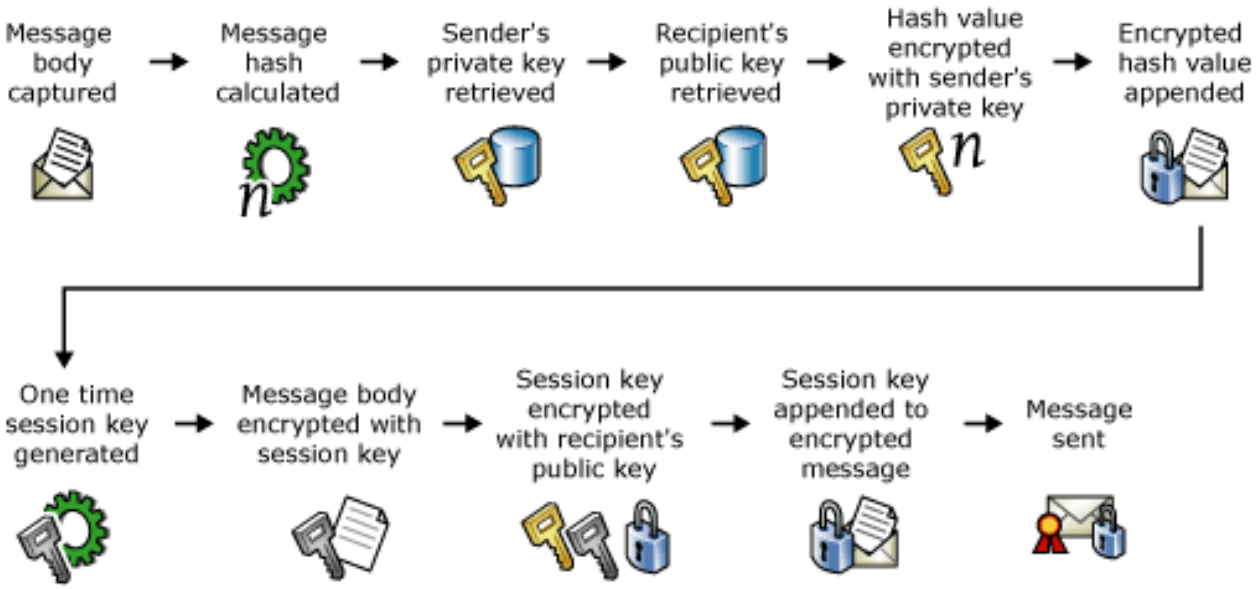
Raqamli imzoni shakllantirishda S/MIME protokoli PKCS#7 formatidan foydalaniladi. S/MIME protokolining afzalligi quyidagilardan iborat:

- S/MIME protokoli transport MIME ma'lumotlari bilan ham foydalanilishi mumkin, HTTP;
- S/MIME protokoli xavfsizlikning asos xususiyatlari bo'lgan, butunlik, maxfiylik va autentifikatsiyani ta'minlaydi;
- S/MIME protokoli moslashuvchan bo'lib, turli *e-mail* ilovalardan foydalanuvchi mijozlar orasida xavfsiz ma'lumot almashinuvini ta'minlaydi;
- S/MIME protokoli qolgan turli ishlab chiqaruvchilar, Microsoft, Lotus, tomonidan elektron pochtaning xavfsiz vositasi ekanligi tasdiqlangan.

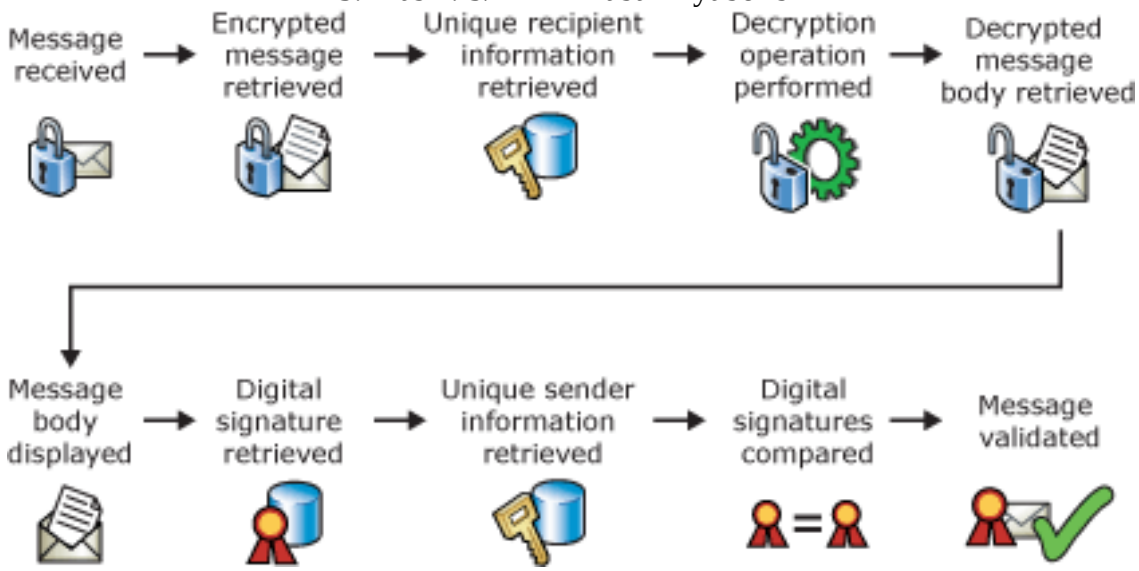
Ushbu protokol ko'plab posta mijozlari, Microsoft Outlook, Mozilla, The Bat! va h. lar tomonidan keng foydalaniladi. Quyidagi 11.1 – rasmda S/MIME protokolida imzo chekilgan xabarning fragmenti keltirilgan.

```
X-MS-Exchange-Organization-Network-Message-Id: 6cceb407-6979-4078-f56b-08d50a68cef4
X-MS-Exchange-Organization-AuthSource: xxxxxx.sec-consult.com
X-MS-Exchange-Organization-AuthAs: Internal
X-MS-Exchange-Organization-AuthMechanism: 07
X-Originating-IP: [xxxxxx]
X-MS-Exchange-Organization-AVStamp-Enterprise: 1.0
X-MS-Exchange-Transport-EndToEndLatency: 00:00:00.2931048
MIME-Version: 1.0

------=_NextPart_000_0066_01D33C62.701A7500
Content-Type: text/plain; smime-type=enveloped-data; name="smime.p7m"
Content-Transfer-Encoding: 7bit

SECRET

------=_NextPart_000_0066_01D33C62.701A7500
Content-Type: application/pkcs7-mime; name="smime.p7m"
Content-Transfer-Encoding: base64
Content-Disposition: attachment; filename="smime.p7m"

MIAGCSqGSIb3DQEHA6CAMIACAQAxggMOMIIBgwIBADBrMF0xEzARBgoJkiaJk/IsZAEZFgNjb20x
GzAZBgoJkiaJk/IsZAEZFgtzZWMtY29uc3VsdDETMBEGCgmSJomT8ixkARkWA3ZpZTEUMBIGA1UE
AxMLU0VDLVJvb3QtQ0ECCmD8o4wAAQAAEdswDQYJKoZIhvcNAQEBBQAEggEAAhdUMfGUdSvb2Egx
EnimK23adJDRt5kg0b93UImkqfXMNgkvLKphhRMs/sdPIOM/vjTwwqcD9vK9gU5OwUASZ42H6pmQ
A5WZv8a0mm2lHmQwNSYX8oo4W5FerCUNDb41zEDOtPAXP45+kErfQEZL2mFDno4oEQfi4lWnsa3P
1XdgKMeSxAMxNfLMhHhhYhlNzkTzrYaJXyCEH1iH/WarLklDNJsCdW6AZfRDNJiN6IIBzsDfkoPd
```

5.1 – rasm. S/MIME protokol ma'lumoti fragmenti



5.2-rasm. S/MIME xabarni yuborish

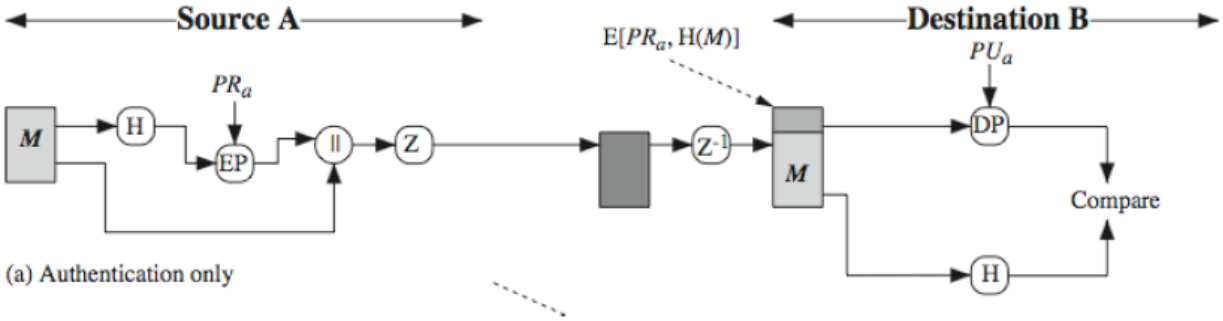


5.3-rasm. S/MIME xabarni qabul qilish

Pretty Good Privacy (PGP)

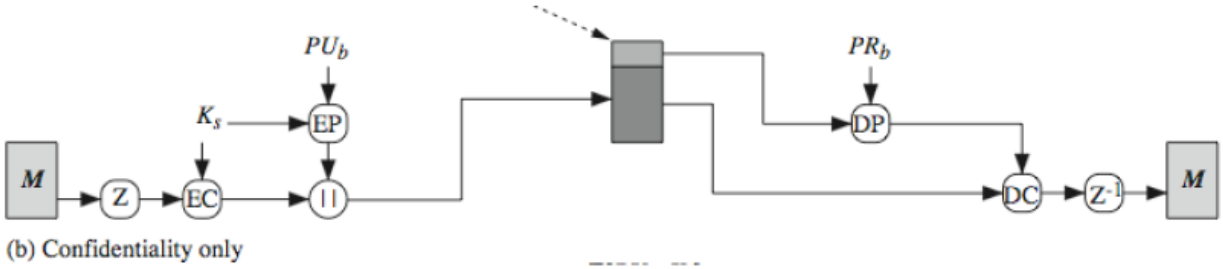
Email xavfsizligida keng foydalaniladi. Phil Zimmermann tomonidan ishlab chiqilgan. Yagona dasturiy vositaga birlashtirilgan. Umumiy holda ochiq, biroq hozirda kommersial ko‘rinishlari ham mavjud. IETF tomonidan OpenPGP standarti ishlab chiqilgan.

PGP amali – Autentifikatsiyalash. Yuboruvchi xabarni yaratadi. Xabar xeshini SHA1 asosida hosil qiladi. Xabarni imzolashda RSAdan foydalaniladi. Qabul qiluvchi deshifrlaydi, xesh kodni tiklaydi. Qabul qilingan xabar xeshi bilan tekshiradi.



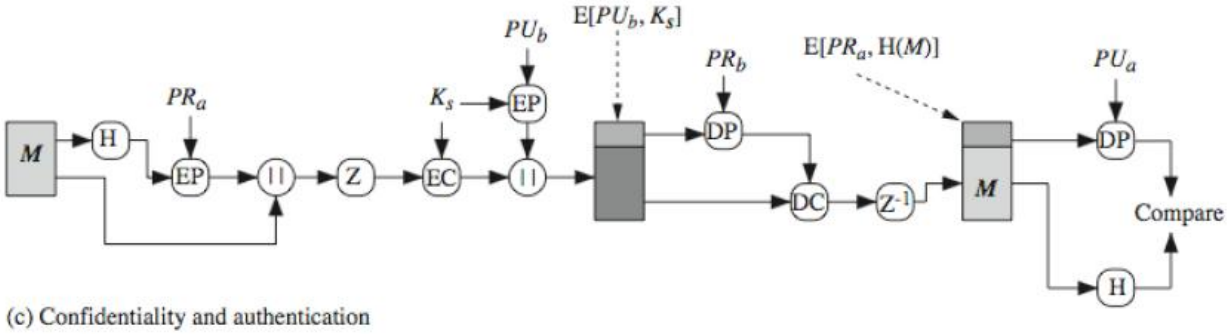
PGP amali – maxfiylik. Yuboruvchi 128-bitli sessiya kalitini generatsiya qiladi. Sessiya kaliti bilan xabarni shifrlaydi. Sessiya kaliti RSA algoritmida shifrlanadi. Qabul qiluvchi

deshifrlaydi va sessiya kalitini oladi. Sessiya kaliti xabarni deshifrlash uchun xizmat qiladi.



(b) Confidentiality only

Maxfiylik va autentifikatsiya. Xabarga imzo qo‘yiladi. Xabar va imzo shifrlanadi. Sessiya kaliti RSA asosida shifrlanadi.



(c) Confidentiality and authentication

PGP amali – EMAILga sozlash. PGP xabarlarni segmentlarga ajratadi agar u katta bo‘lsa. PGP binar ma’lumotlarni hosil qiladi. EMAIL xabar matn uchun mo‘ljallangan:

- Binar ma’lumotni ASCII belgisiga o‘tkazish kerak.

Radix-64 yoki Base-64 dan foydalaniladi.

- 3 baytni 4 ta pechat qilinuvchi belgiga o‘tkazish.

Text content	M								a								n							
ASCII	77								97								110							
Bit pattern	0	1	0	0	1	1	0	1	0	1	1	0	0	0	0	1	0	1	1	0	1	1	1	0
Index	19								22								5							
Base64-encoded	T								W								F							

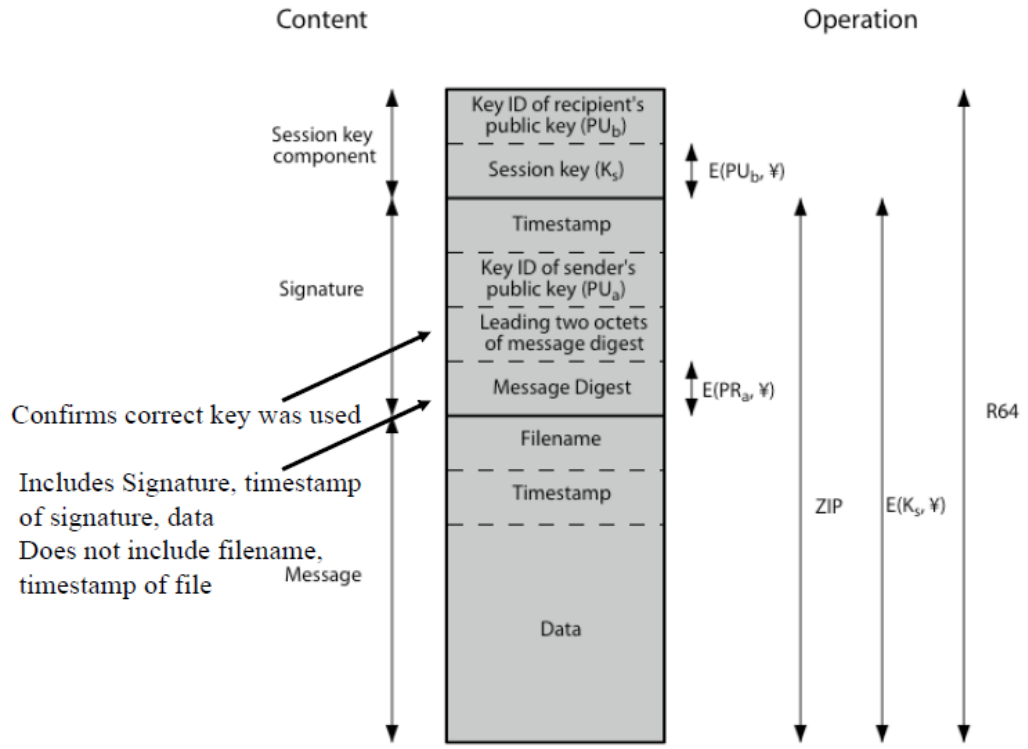
PGP sessiya kalitlari. Xar bir xabar uchun turli uzunlikdagi sessiya kalitlari zarur

- 56-bit DES
- 128-bit 3DES
- 128-bit CAST
- IDEA (International Data Encryption Algorithm)

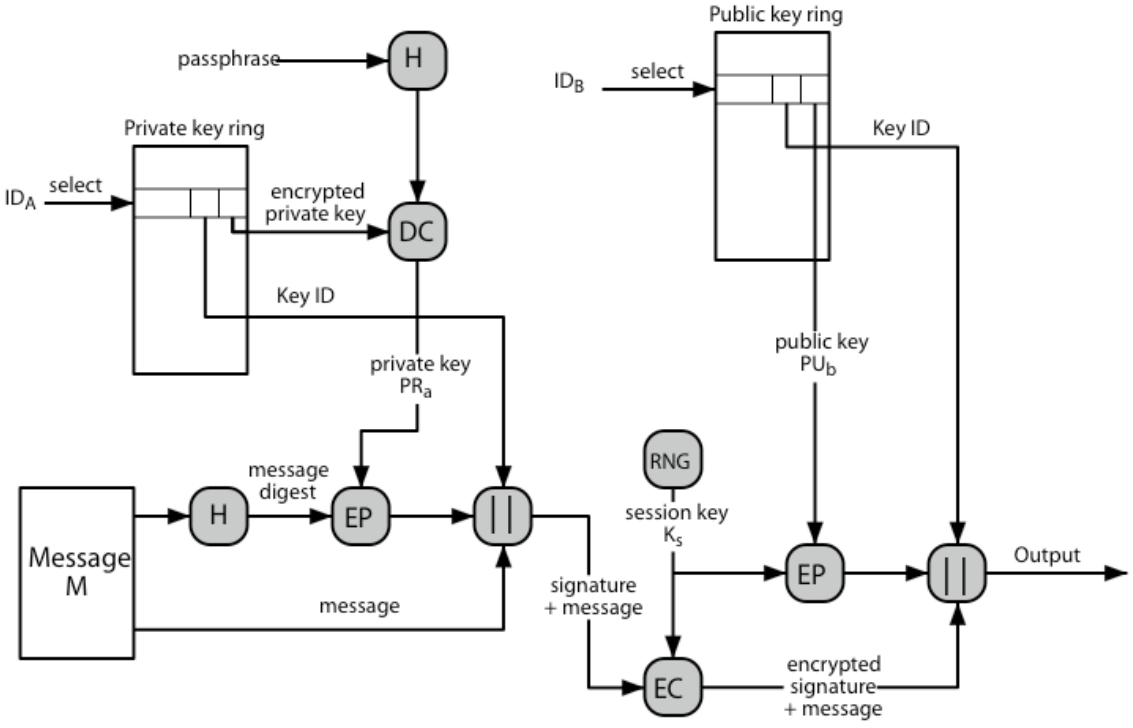
Sessiya kalitlari tasodifiy sonlar generatori tomonidan hosil qilinadi.

PGP ochiq va maxfiy kalitlari. Foydalanuvchilarga bir nechta ochiq/maxfiy kalitlardan foydalanishga ruxsat beriladi. Qaysi kalit foydalanilganini aniqlash zarur. Kalit identifikatori= kalitning 64-bit LSB dan iborat, foydalaniladi. Imzolash kaliti shifrlash kalitidan farq qilishi zarur.

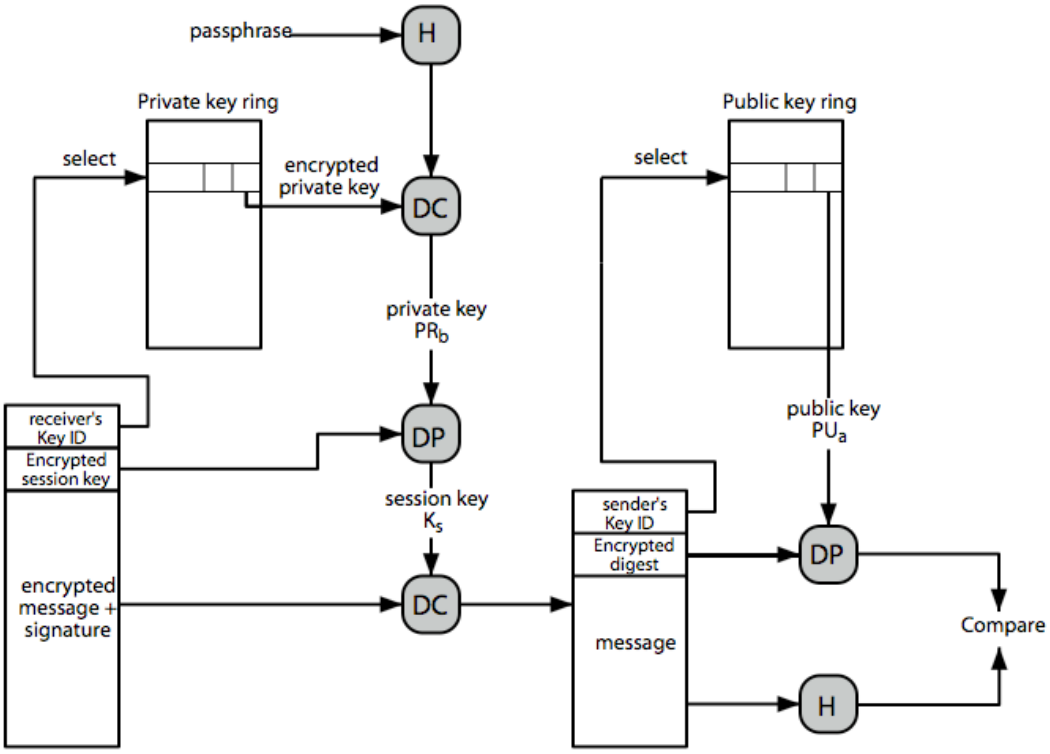
PGP xabar formati



PGP xabarini hosil qilish



PGP xabarni qabul qilish



Ishonch zanjiri

Biror tashkilotdan sertifikat sotib olish zarur emas. Bir foydalanuvchi boshqasiga imzo qo‘yishi mumkin. Agar biror kishiga ishonsangiz, unga imzo qo‘yganlarga ham ishonishingiz zarur. Siz har bir foydalanuvchi uchun ishonch darajasini belgilashingiz mumkin. Masalan: To‘liq ishonchli foydalanuvchi imzolagan sertifikat to‘liq ishonchli. Ikki yarim ishonchli foydalanuvchilar imzolagan sertifikat to‘liq ishonchli. Bir yarim ishonchli foydalanuvchi imzolagan sertifikat yarim ishonchli. Ba’zi sertifikatlar ishonchsiz bo‘lishi mumkin.

Nazorat savollari

1. PGP protokoli va uning vazifasi.
2. PGP protokolida mavjud kriptografik algoritmlar.
3. S/MIME protokoli va uning vazifasi.
4. MIME protokolida mavjud muammolar.

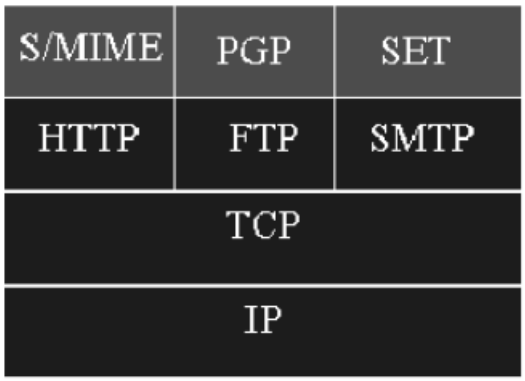
6-ma’ruza. Xavfsiz tranzaksiya protokollari

Reja:

1. SET (Secure Electronic Transactions) protokoli.
2. SET protokoli tashkil etuvchilari.
3. Ikki marta imzolash.

SET (Secure Electronic Transactions) protokoli

SET (Secure Electronic Transactions) protokoli OSI modelining ilova sathida ishlaydi.



6.1-rasm. Ilova sathida ishlovchi xavfsizlik protokollari

SET protokoli ikkita asosiy qismdan iborat:

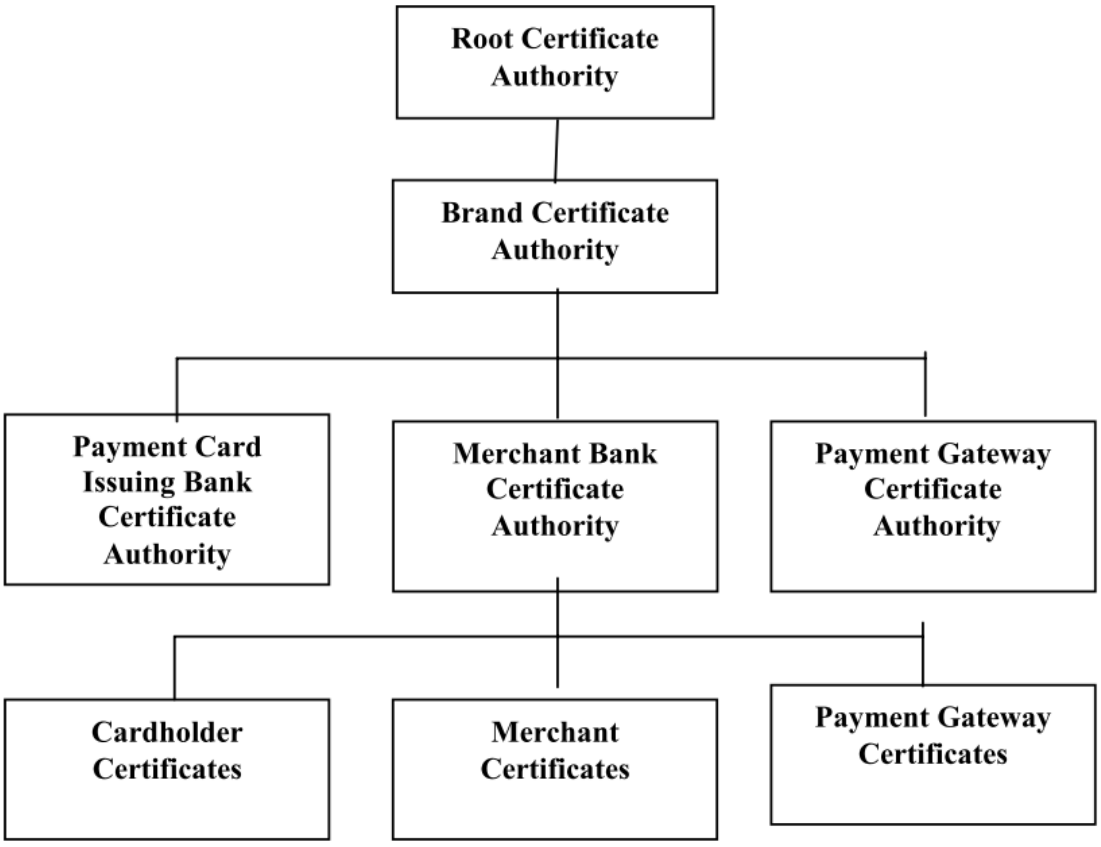
1. Ro‘yxatdan o‘tkazish muolajasi.
2. Tranzaksiya muolajasi.

Ro‘yxatdan o‘tkazish muolajasi

SET protokoli PKI xizmatiga asoslangan bo‘lib, tranzaksiyada uchta turdagi ishtirokchi mavjud bo‘ladi:

1. Mijoz (karta egasi).
2. Sotuvchi.
3. To‘lovni amalga oshiruvchi marshrutizator.

SETda har bir ishtrokchilarga sertifikatlarni tarqatish iyerarxik tarzda amalga oshiriladi. Ushbu holat quyidagi 13.2-rasmda keltirilgan. Eng yuqori darajada *root certificate authority* joylashgan bo‘lib, u Secure Electronic Transaction LLC (SETCo) (1997-yilda Visa va MasterCard tomonidan tashkil etilgan) tomonidan yuritiladi. Root Certificate Authority turli to‘lov brendlari uchun ochiq kalitlar sertifikatini taqdim etadi. Ular o‘z navbatida Certificate Authorities (SA) kabi bo‘lib, o‘zlariga tegishli banklarga sertifikat tarqatadi.



6.2-rasm. SET sertifikati iyerarxiyasi

Payment card issuing certificate authority tashkiloti mijozlarga ochiq kalitlar sertifikatini tarqatsa, *merchant bank* yoki *acquirer certificate authority* tashkiloti esa sotuvchilar uchun sertifikatlarni tarqatsa, *Payment Gateways* lar uchun o‘zining CA markazi mavjud bo‘ladi.

Ushbu arxitekturada ixtiyoriy iyerarxiyada *sertifikatlar zanjiri* asosida tekshirish mumkin. Masalan, biror sotuvchi uchun sertifikat zanjiri o‘zining SA tomonidan berilgan ochiq kaliti sertifikati, o‘zining SA ga brend SA tomonidan berilgan sertifikat va brend sertifikatga o‘zak SA tomonidan berilgan sertifikatdan iborat bo‘ladi. Uning ko‘rinishi quyidagi 13.3-rasmda keltirilgan.

Merchant’s Certificate (from Merchant’s Bank CA)	**Merchant’s Bank CA Certificate (from Brand CA)**	**Brand’s CA Certificate (from Root CA)**

6.3-rasm. Sertifikat zanjiri

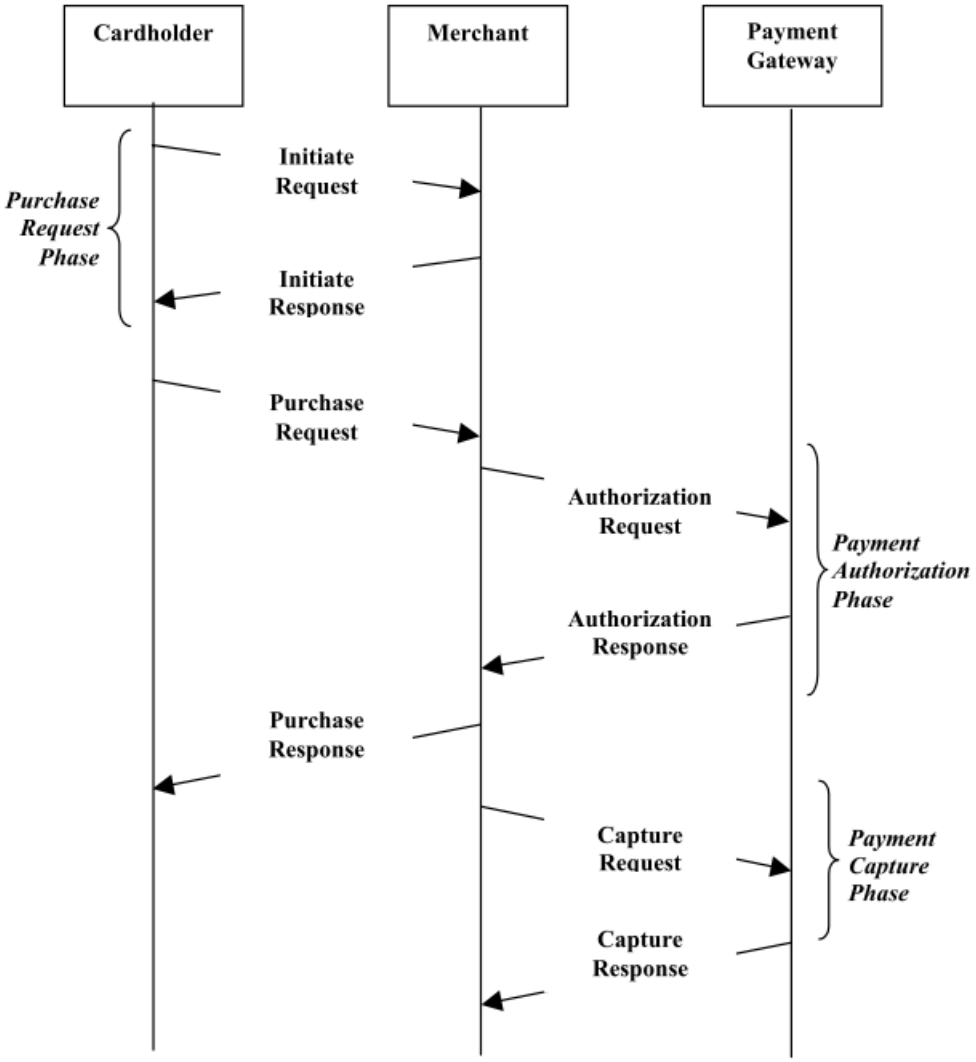
Foydalanuvchi sertifikatni olish uchun CAga unique identification information (ID) va ochiq kalitini taqdim etadi va unga mos sertifikatni qabul qiladi.

Tranzaksiya muolajasi

Xavfsiz elektron tranzaksiyalarda quyidagi uchta asosiy bosqich mavjud:

1. Buyurtma so‘rovi.
2. To‘lov avtorizatsiyasi.
3. To‘lovni olish.

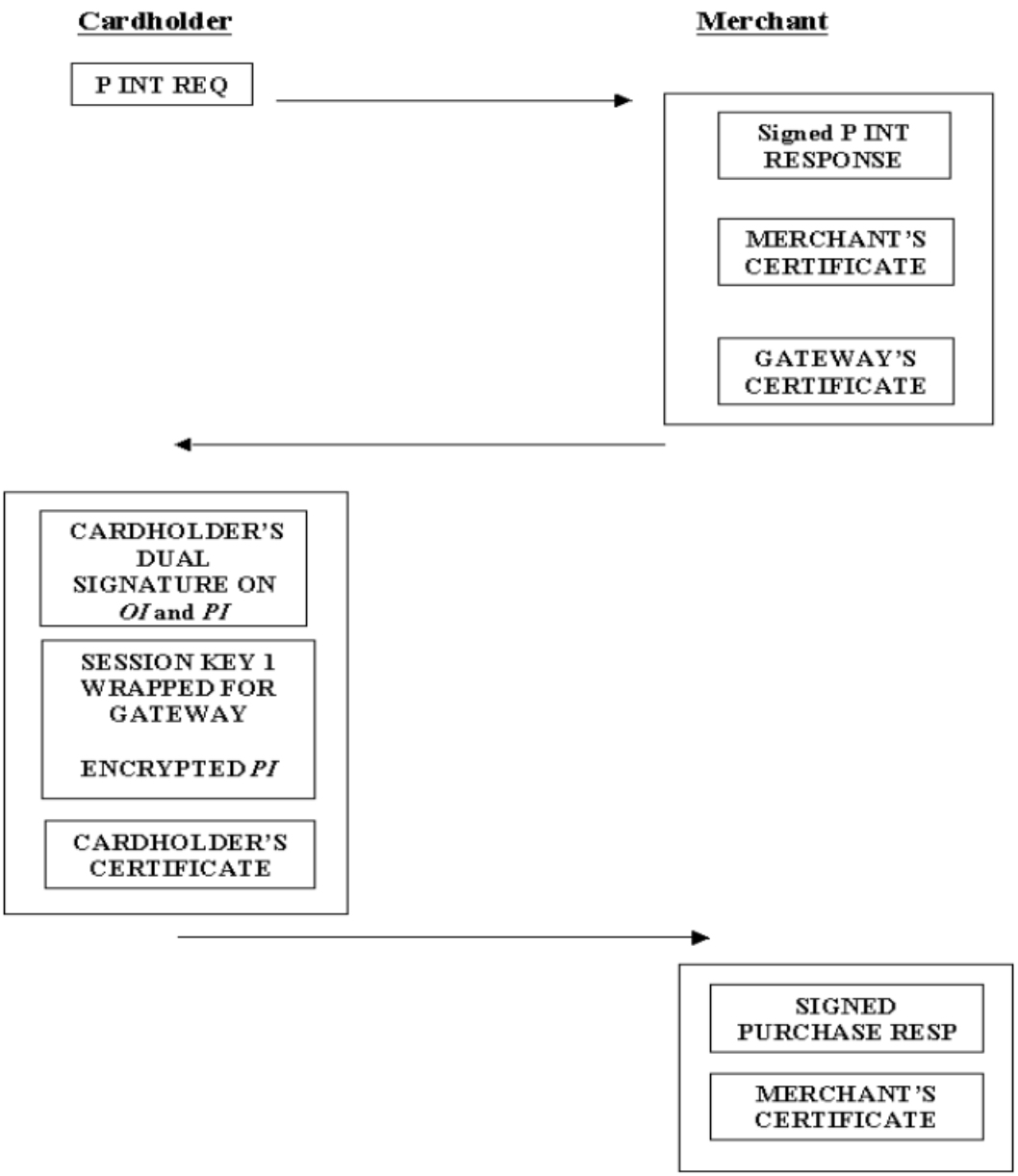
Umumiy holda ishtirokchilarni tranzatsiyadagi ishtiroki quyida keltirilgan 13.4-rasmdagi kabi bo‘ladi.



6.4-rasm. SET protokolining umumiy ko‘rinishi

Buyurtma so‘rovi bosqichi

Buyurtma so‘rovini amalga oshirish quyidagi 13.5-rasmda keltirilgan.



6.5-rasm. Buyurtma so‘rovi bosqichi

Buyurtma so‘rovini ichida quyidagi 5 ta qadam mavjud:

Initiate Request. Jarayon mijozni bir yoki bir nechta narsalarni sotib olishdan boshlanadi. Barcha kerakli narsalarni tanlagandan so‘ng biror to‘lov kartani tanlaydi. Karta egasi kompyuterida ishlovchi dasturiy vosita dastlabki so‘rov (*Initiate Request, P INIT REQ*) ni payment gateway ochiq kalitini so‘rovchi sotuvchiga yuboradi.

Initiate Response. Sotuvchi *Initiate Request* ni qabul qilingandan so‘ng, u xabarga unique transaction IDni tayanlaydi va imzolangan ID ni, o‘zining sertifikatini va to‘lov marshrutizatori sertifikatini qaytaradi.

Cardholder Purchase Request. Javob qabul qilingandan so‘ng, karta egasi sotuvchi va marshrutizator sertifikatini tekshiradi va xabarga qo‘yilgan imzoni tekshiradi. Bu jarayon tugagandan so‘ng, karta egasi ikkita xabarni hosil qiladi: sotuv oluvchi uchun *axborot tartibi (order information (OI))* va to‘lov marshrutizatori uchun mo‘ljallangan *to‘lov axborot (payment information (PI))*. PI xabar ma’lumoti masalan, karta raqami sotuvchidan yashiriladi. Ushbu ikkala xabar ham sotuvchi tayinlagan yagona ID ga ega bo‘ladi. Bu nuqtada ikkita xabarni bir-biriga bog‘lash uchun ajoyib usul qo‘llaniladi. Karta egasi OI va PI ning xesh qiymatini shakllantiradi. Ushbu ikki xesh qiymat birlashtiriladi va uchinchi xesh qiymat shakllantiriladi. Uchinchi xesh qiymat karta egasi tomonidan imzolanadi va u OI va PI uchun *ikki marta imzolangan* deyiladi.

Keyingi qadamda PI axborot sotuv oluvchidan himoyalanadi. Karta egasi tasodifiy sessiya kalitini generatsiya qiladi va PI ni shifrlaydi. Bu axborotni to‘lov marshrutizatoriga yuborish uchun karta egasi sessiya kaliti va o‘zining qayd ma’lumotlarini birlashtiradi va uni to‘lov marshrutizatorini ochiq kaliti bilan shifrlaydi.

Ushbu ma’lumotlar keyingi qadamda sotuvchi orqali to‘lov marshrutizatoriga yuboriladi. Bu yerda ikki marta imzolashdan asosiy maqsad quyidagilar: to‘lov marshrutizatorida OI ning xesh qiymati, uning o‘zi emas, saqlanadi. To‘lov marshrutizatori bu axborotdan savdo ma’lumotini hisoblay olmaydi. Agar sotuvchi va mijoz orasida nizo kelib chiqqanda to‘lov marshrutizatori uni tekshirish imkoniyatiga ega bo‘ladi. Bu SET protokolining muhim xususiyatlaridan biri sanaladi.

Merchant‘s Purchase Request Processing. Sotuv haqidagi xabar sotuvchiga qabul qilinganda u karta egasini sertifikatini tekshiradi. Bu keyinchalik ikki marta imzolangan imzoni tekshirish uchun ishlatiladi.

Tekshiruv tugagandan so‘ng sotuvchi imzolangan *purchase response* ni generatsiya qiladi va uni karta egasiga uzatadi.

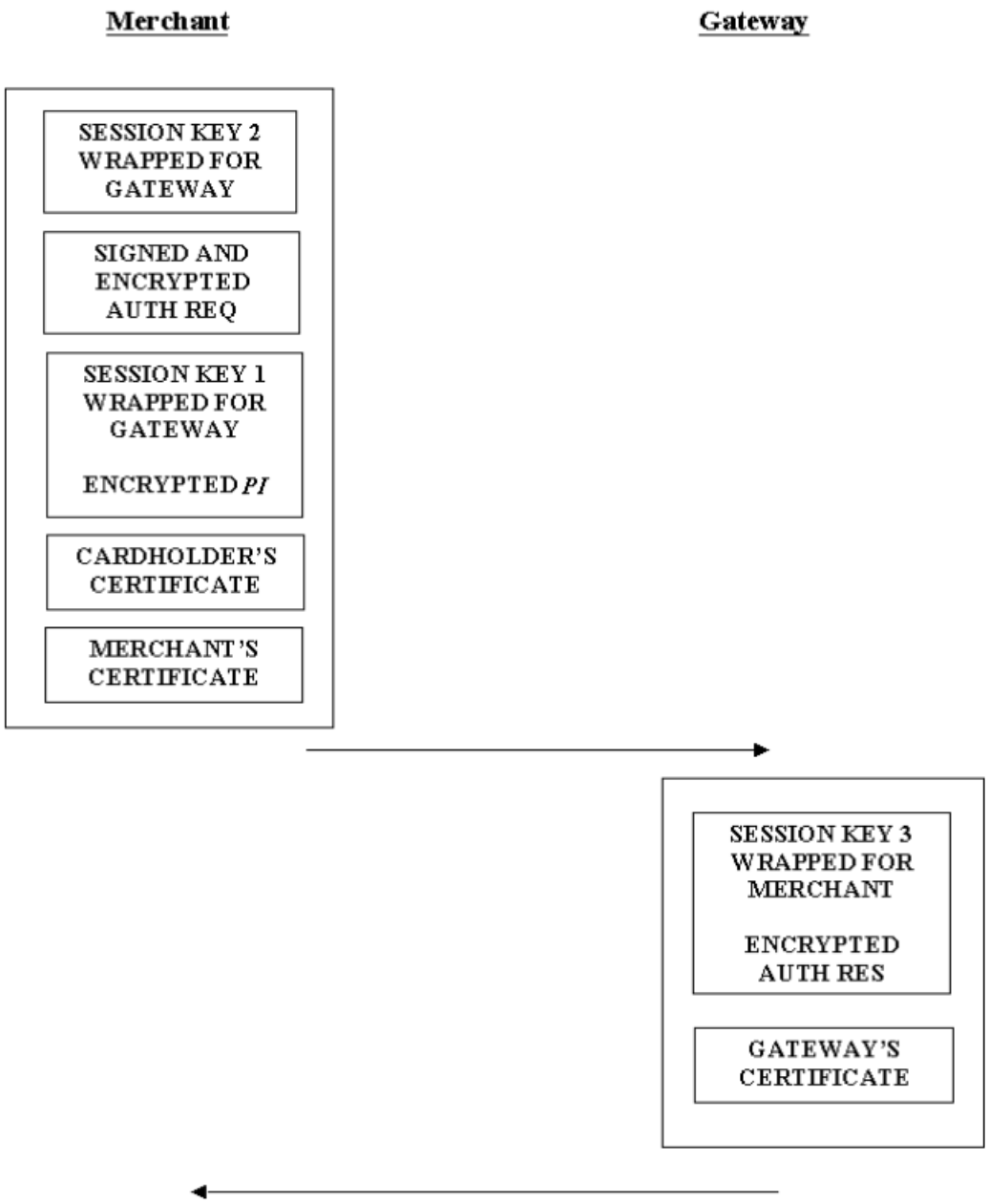
Purchase Response. Oxirgi qadamda karta egasi sotuvchining ochiq kaliti yordamida purchase response ni tekshiradi va keyingi murojaatlar uchun saqlanadi.

To‘lov avtorizatsiyasi

Protokolning bu qismi sotuvchi va to‘lov marshrutizatorini o‘z ichiga oladi. Bunda asosiy maqsad sotuvchini tranzaksiyani amalga oshirish uchun avtorizatsiyadan o‘tishi hisoblanadi. Bunda 3 ta asosiy qadam mavjud (13.6-rasm).

Merchant Authorization Request. Sotuvchi raqamli imozalangan avtorizatsiya so‘rovini generatsiya qiladi. Avtorizatsiya so‘rovi avtorizatsiyalanishi kerak bo‘lgan summa, tranzaksiya ID si va tranzaksiya haqidagi boshqa axborotdan iborat bo‘ladi.

Sotuvchi tasodifiy sessiya kalitini generatsiya qiladi va u asosida avtorizatsiya so‘rovini shifrlaydi. Sessiya kaliti esa to‘lov marshrutizatori ochiq kaliti bilan shifrlanadi. Ushbu ma’lumotlar karta egasining PI axboroti va shifrlangan sessiya kaliti, karta egasining ochiq kalit sertifikati va sotuvchining ochiq kalit sertifikati bilan birgalikda to‘lov marshrutizatorga yuboriladi.



6.6-rasm. To‘lov avtorizatsiyasi bosqichi

Payment Gateway Processing. To‘lov marshrutizatori avtorizatsiya so‘rovini qabul qilganda, o‘zining maxfiy kaliti bilan shifrlangan sessiya kalitlarini deshifrlaydi. Sessiya kaliti bilan esa so‘rovni deshifrlaydi. Sotuvchi sertifikati tekshiriladi va u asosida so‘rovni haqiqiyligi tekshiriladi.

Shundan so‘ng, ikkinchi sessiya kaliti va mijozning axborotni tiklanadi. Sessiya kalitidan foydalanib PI tiklanadi. Karta egasining ochiq kaliti tekshiriladi va u asosida OI va PI tekshiriladi. Xabarlarni ikkala qismidagi tranzaksiya ID larini solishtirish orqali ularning bir xilligi tekshiriladi.

Keyingi bosqichda to‘lov marshrutizatori joriy bank uchun xabar generatsiya qiladi. Bu shaxsiy moliyaviy tarmoq orqali amalga oshiriladi.

Agar savdo avtorizatsiyadan o‘tkazilsa, to‘lov marshrutizatori tomonidan imzolangan javob qaytariladi. Bu javob yangi sessiya kaliti bilan shifrlangan bo‘lib, sessiya kaliti esa sotuvchining ochiq kaliti bilan shifrlangan bo‘ladi.

Merchant Response Processing. Sotuvchiga javob qabul qilinganda to‘lov avtorizatsiyasi tiklanib, tekshiriladi. Avtorizatsiyaning nusxasi sotuvchiga saqlanadi.

To‘lovni olish

SET protokolidagi oxirgi bosqich to‘lovni olish bosqichi bo‘lib, unda sotuvchi to‘lov marshrutizatoridan to‘lovni talab etadi. Bu bosqich tranzatsiya amalga oshirilgandan so‘ng

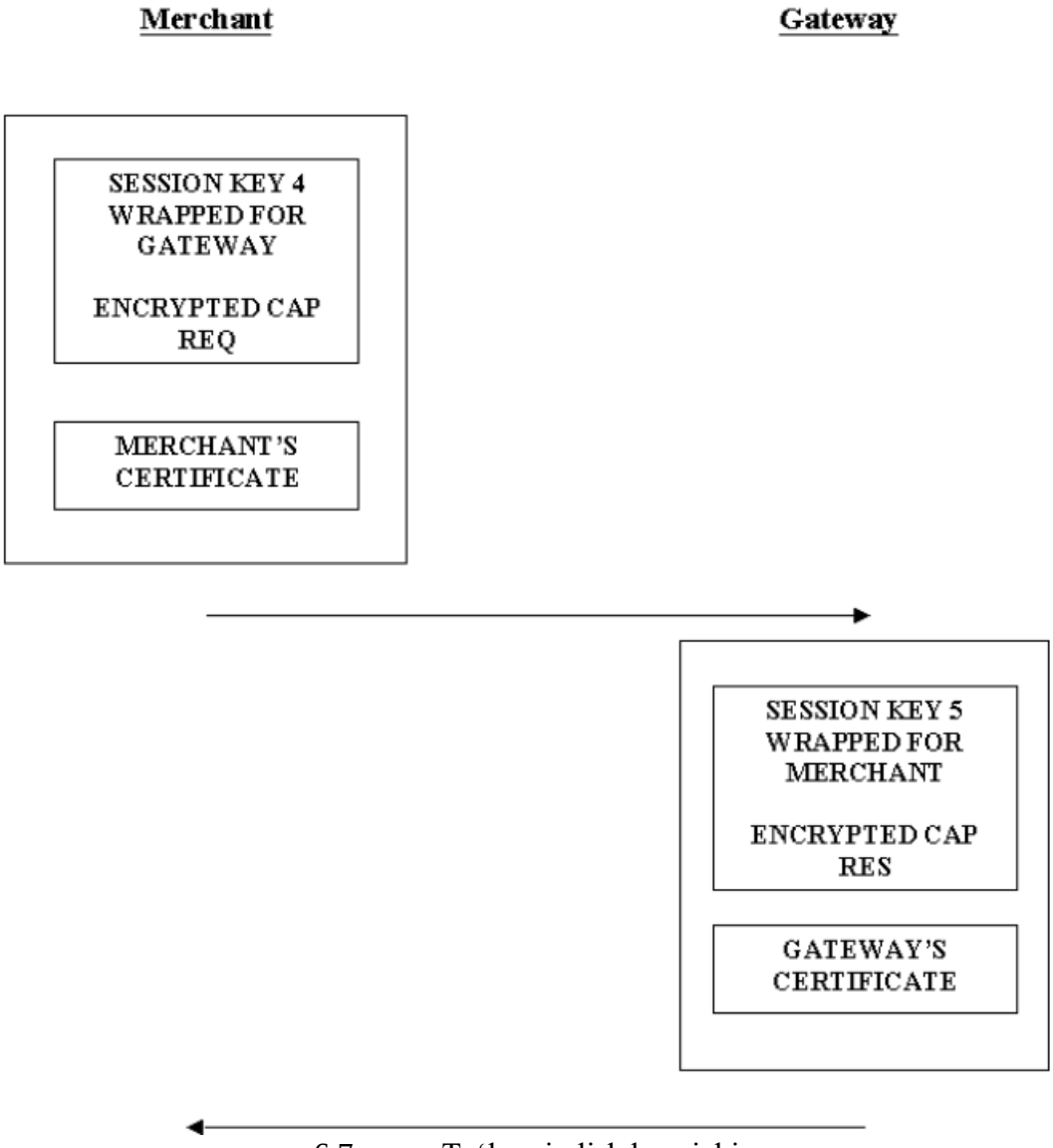
ma’lum vaqtdan so‘ng amalga oshirilishi mumkin va u quyidagi uchta qadamdan iborat bo‘ladi (13.7-rasm).

Merchant Payment Capture Request. Sotuvchi oxirgi tranzaksiya summasini, tranzatsiya ID sini va boshqa tranzatsiya ma’lumotlarini o‘z ichiga olgan to‘lov summasini hosil qiladi va imzolaydi. Bu yangi sessiya kaliti bilan shifrlanadi va sessiya kaliti to‘lov marshrutizatorining ochiq kaliti bilan shifrlanadi. Shifrlangan so‘rov va sessiya kaliti sotuvchining ochiq kalit sertifikati bilan birga to‘lov marshrutizatoriga yuboriladi.

Payment Gateway Capture Processing. To‘lov marshrutizatori sessiya kalitini tiklaydi va to‘lovni olish so‘rovini tiklaydi. Shundan so‘ng sotuvchi sertifikatini tekshirish orqali so‘rovga qo‘yilgan imzoni tekshiradi.

Shundan so‘ng to‘lov marshrutizatori imzolangan va shifrlangan javob so‘rovini generatsiya qiladi va uni o‘zining ochiq kalit sertifikati bilan birga sotuvchiga yuboradi.

Merchant Processing of Response. Bu protokolning oxirgi qadami bo‘lib, sotuvchi sessiya kaliti va to‘lovni olish xabarini tiklaydi hamda xabarga qo‘yilgan imzoni haqiqiyligini tekshiradi. Bu xabar to‘lov marshrutizatorida bo‘lishi mumkin bo‘lgan fors-major holatlar uchun saqlanadi.



6.7-rasm. To‘lovni olish bosqichi

SETning 1.0 versiyasida ochiq kalitli kriptogarfik algoritm sifatida RSA algoritmi (kamida 768 - bit) foydalanilgan. Bundan tashqari elliptik egri chiziqga asoslangan ko‘rinishlari ham mavjud. Hozirgi kunda IBM, Verisign, CyberTrust, Verifone, Sterling Commerce, Terisa, Netpay va GlobeSet kompaniyalari ushbu xizmatni taklif etadi.

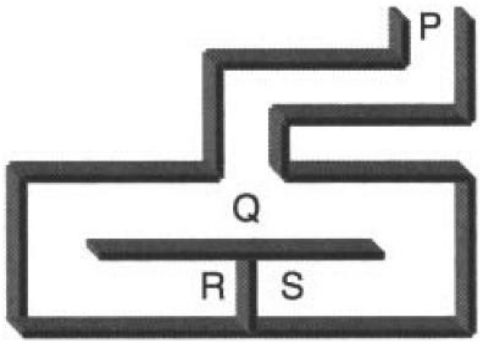
Nazorat savollari

1. SET protokoli OSI modelining qaysi sathida ishlaydi.
2. SET protokolida autentifikatsiyalashda nimaga asoslanadi?
3. SET protokolida ochiq kalitli kriptografik algoritmlar sifatida qaysi algoritmlar foydalanilgan?

7-mavzu: E'lon qilinganligi nolga teng protokollar (2 soat)

ZKP da Alisa Bobga maxfiy kattalikni bilsada u haqida axborot bermasdan o‘zini haqiqiyligini tasdiqlashga harakat qiladi. Bu holda Bob Alisani maxfiy xabarni bilganda va hattoki u haqida axborotga ega bo‘lmasa ham uni tekshirish imkoniyatiga ega bo‘ladi. Bir ko‘rinishdan uning imkoni yo‘qdek. Biroq, interekaktiv ehtimoliy jarayon mavjud bo‘lib, unda Bob Alisani maxfiy kattalikni yuqori ehtimol bilan bilishini tekshiradi. Bu interaktiv tasdiqlash tizimiga misol bo‘la oladi.

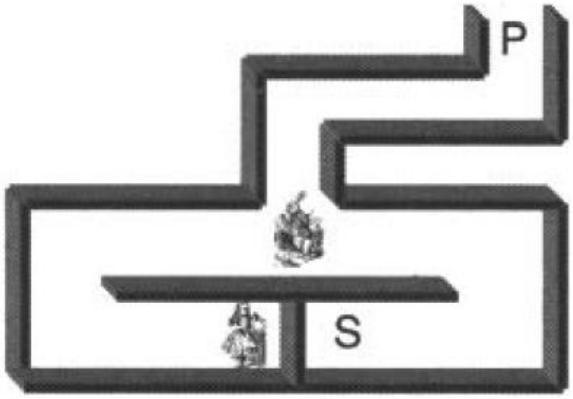
Bu toifadagi protokollarni tahlil qilishdan oldin, dastlab “Bobning g‘ori” bilan tanishib chiqsak (29-rasm).



29-rasm. Bobning g‘ori

Faraz qilaylik Alisa R va S orasidagi eshikni ochish uchun kerakli maxfiy iborani (“ochil simsim”) bilaman deb iddao qiladi. Alisa Bobni ushbu iborani aytmasdan turib ishontira oladimi?

Quyidagicha protokolni faraz qilaylik. Alisa Bobning g‘origa kirdi va tanga tashlab R yoki S tomonga borishini hal qildi. Shundan so‘ng g‘orga Bob kirdi va Q nuqtaga keldi. Faraz qilaylik Alisa R nuqtada (30-rasm).



30-rasm. Bobning g‘ori protokoli

Bob tangani tashlaydi va Alisaga ushbu tomonga bo‘lishini talab etadi. 30-rasmda ifolangani kabi, agar Bob R tomonni tanlagan bo‘lsa, u holda Alisa maxfiy iborani bilish yoki bilmasligidan qat’iy nazar R tomonda bo‘ladi. Ammo Bob S tomonni tanlagan vaqtda, Alisa faqat maxfiy iborani bilgandagina bo‘la oladi. Boshqa so‘z bilan aytganda, agar Alisa maxfiy iborani bilmasa Bobni aldab ishontirish ehtimoli ½ ga teng. Bu ko‘rinishdan foydali sanalmasada, biroq, agar protokol n marta takrorlanganda ushbu ehtimollik $(1/2)^n$ ga teng bo‘ladi. Shuning uchun, Alisa va Bob ushbu protokolni n marta takrorlaganda, Alisa Bobni maxfiy iborani bilishga

ishontirishi har safar ammalga oshirilishi mumkin.

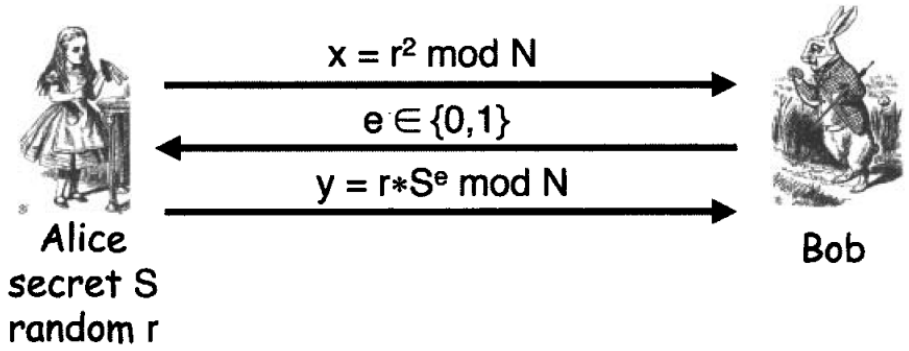
Agar Alisa (yoki Tridi) maxfiy iborani bilmagan taqdirda ham Bobni ishontirish imkoniyatiga ega bo‘lar ekan. Biroq, Bob ushbu ehtimolni *n* ni mos tanlash orqali yetarlicha kamaytirishi mumkin. Masalan, *n=20* ga teng bo‘lsa, 1000000 dan 1 urinishda Alisa maxfiy iborani bilmasdan turib haqiqiyligini isbotlashi mumkin bo‘ladi. Shuningdek, ushbu protokolda Bob hech narsa o‘rgana olmaydi. Nihoyat, ushbu protokolda muhim jixatlardan biri bu Bobni qaysi tomonga paydo bo‘lishni so‘rashi tasodifiy bo‘lishi shart. Agar Alisa yoki Tridi Bobni qaysi tomonni so‘rashini oldindan bilsa, yanada yaxshiroq imkoniyat paydo bo‘ladi.

Bobning g‘ori shuni ko‘rsatadiki ZKP holati mavjud ekan. Biroq ushbu turga tegishli protkollar amalda mashxur emas. Buni amalda ko‘rish mumkinmi? Ha, Fiat-Shamir protokoli bilan tanishib chiqiladi?

Fiat-Shamir protokoli faktorlash muammosidagi modul *N* asosida kvadrat ildizni topishga qaratilgan. Faraz qilaylik $N=pq$ ga teng va bu yerda *p* va *q* lar tub sonlar. Alisa maxfiy *S* ni biladi va uni maxfiy saqlashi shart. *N* va $v=S^2 mod N$ kattaliklar ochiq. Bunda Alisa S haqidagi bilimlarni oshkor etmasdan Bobni ishontirishi zarur.

31-rasmda Fiat-Shamir protokoli keltirilgan bo‘lib, quyidagi asnoda ishlaydi. Alisa *r* ni tasodifiy tanlaydi va u asosida $x=r^2 mod N$ ni hisoblaydi. Birinchi xabarda, Alisa *x* ni Bobga yuboradi. Ikkinchi xabarda Bob tasodifiy qiymat $e \in \{0,1\}$ ni tanlaydi va uni Alisaga yuboradi. O‘z navbatida Alisa quyidagini hisoblaydi: $y=rS^e mod N$. Uchinchi xabarda Alisa *y* ni Bobga yuboradi. Shundan so‘ng, Bob quyidagini tekshiradi: $y^2=xv^e mod N$.

Ya’ni, $y^2=r^2S^{2e}=r^2(S^2)^e=xv^e \ mod \ N$. (1)



31-rasm. Fiat-Shamir protokoli

Ikkinchi xabarda, Bob *e=0* yoki *e=1* ni yuboradi. Har bir holatni alohida alohida qarab chiqaylik. Agar Bob *e=1* ni yuborsa, u holda Alisa uchinchi xabarda *y=rSmodN* bilan javob beradi va (1) tenglik quyidagicha bo‘ladi:

$$y^2=r^2S^2=r^2(S^2)=xv \ mod \ N$$

Bu holda Alisandan S ni bilish talab etiladi.

Boshqa holatda, agar Bob *e=0* ni ikkinchi habarda yuborsa, u holda uchinchi xabarda Alisadan $y = r \ mod \ N$ ni oladi va (1) tenglik quyidagicha bo‘ladi:

$$y^2=r^2=x \ mod \ N$$

Ushbu holatda Alisadan S ni bilish talab etilmaydi. Bu esa jiddiy muammo bo‘lib, Bobning g‘oridagi holatga o‘xshaydi.

Ushbu protokolning xavfsizligi tanlanadigan tasodifiy qiymat *e* ni tanlanishiga bog‘liq. Faraz qilaylik Tridi Bob ikkinchi xabarja *e=0* ni yuborishini biladi. U holda Tridi birinchi xabarda $x=r^2 \ mod \ N$ ni yuboradi va uchinchi xabarda $y=r \ mod \ N$ ni yuboradi. Ya’ni, Tridi maxfiy *S* ni bilishi talab etilmaydi.

Boshqa tomondan, agar Tridi ikkinchi xabarda Bobni *e=1* ni yuborishini oldingan bilsa, u holda birinchi xabarda $x=r^2v^{-1} \ mod \ N$ ni yuboradi va uchinchi xabarda $y= r \ mod \ N$ ni qabul qiladi. Protokolga asosan, Bob $y^2=r^2$ va $xv^e=r^2v^{-1}v=r^2$ ni hisoblaydi va natijani to‘g‘ri deb qabul qiladi.

Bundan xulosa shuki, Bob $e \in \{0,1\}$ ni tasodifiy ravishda tanlashi zarur. Agar shunday bo‘lsa, Tridi ½ ehtimollik bilan Bobni aldashi mumkin bo‘ladi. Bob iteratsiyani *n* marta takrorlashi natijasida ushbu ehtimollikni $(1/2)^n$ gacha tushiradi.

Ochiq kalitli kriptogarfik tizimlardan ushbu muammolarni bartaraf etishda foydalanish har ikkala tomonni anonim bo‘lish ehtimolini yo‘qqa chiqaradi. ZKP asoslangan protokollar esa tomonlarni anonim bo‘lishini ta’minlaydi.

Quyida esa **Shamir protokoli** deb ataluvchi (kalitsiz) umumiy mahfiy ma’lumotdan foydalanmagan holda kalitni uzatish protokolini ko‘rib chiqiladi. Bu protokol qadamlariga muvofiq kalitning mahfiylik masalasi ta’minlanadi.

Shunday shifrlash va deshifrlash o‘zgartirishlari mavjudki barcha x -ma’lumotlar, k_1 va k_2 -kalitlar uchun quyidagi shart bajariladi:

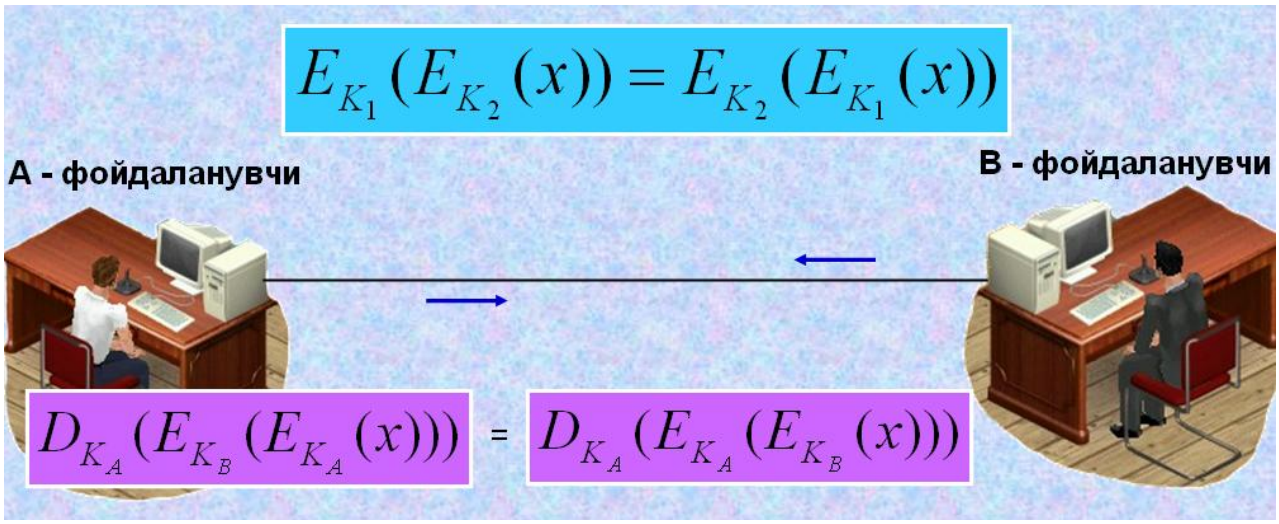
$$E_{k_1}(E_{k_2}(x)) = E_{k_2}(E_{k_1}(x)).$$

U holda A va V -foydalanuvchilar k -seans kalitini uzatuvchi quyidagi 3 - bosqichli protkoldan foydalanishlari mumkin:

(1) $\mathrm{A} \to \mathrm{V}$: $E_{k_A}(k)$,

(2) $\mathrm{V} \to \mathrm{A}$: $E_{k_B}(E_{k_A}(k))$,

(3) $\mathrm{A} \to \mathrm{V}$: $D_{k_A}(E_{k_B}(E_{k_A}(k)))$.



Xususan, Shamir protokolida modul bo‘yicha darajaga ko‘tarish amalidan foydalanish taklif etilgan, ya’ni $E_{k_A}(k) = k^{k_A} \bmod p$. Shunday qilib, bu protokolning kriptobardoshligi diskret logarifmlash masalasining murakkabligiga asoslangan. Shamir protokolining kamchiligi shundaki, bu protokolda autentifikatsiya masalasi hal etilmagan.

Nazorat savollari

1. Nollik bilimga asoslangan autentifikatsiyalash.
2. O‘rtaga turgan odam hujumini tushunting.
3. Shamir protokoli.

IV-BO‘LIM

AMALIY MASHG‘ULOT MATERIALLARI

IV. AMALIY MASHG‘ULOT MATERIALLARI

1-amaliy ish. HLPSL sodda protokollarni ifodalash va SPAN +AVISPA vositasini yuklash (2 soat)

Protokol HLPSL tilida ko‘rsatilishi protokolning o‘zini (xabar almashinuvi), tahlil qilinadigan senariyni va tekshirish uchun xavfsizlik xususiyatlarini belgilashni anglatadi. Shunday qilib, spetsifikatsiya quyidagi elementlardan iborat:

• Protokolda ishtirok etuvchi har bir ro‘lning spetsifikatsiyasi.

• Protokolni ifodalash uchun ushbu ro‘llar tarkibining spetsifikatsiyasi.

• Bajarish muhiti va o‘rganiladigan protokol misollarining spetsifikatsiyasi.

• Tahlil qilish uchun xavfsizlik xususiyatlarining deklaratsiyasi.

• Senariyning bajarilishi.

HLPSL spetsifikatsiyasi tilidan foydalanish quyida NSPK Key Server protokolining spetsifikatsiyasi bilan ko‘rsatilgan, bunda A, agar kerak bo‘lsa, umumiy kalitlarni tarqatish bo‘lgan S serveridan foydalanib, B autentifikatsiya qilishni xohlaydi.

Bu misolda $\boldsymbol{P\,K_x}$ X ning ochiq kaliti, $\boldsymbol{inv(PK_x)}$ esa mos keladigan shaxsiy kalit; $\boldsymbol{N_x}$ protokolni ishga tushirish vaqtida X tomonidan yaratilgan nonces hisoblanadi.

$$\boldsymbol{if\ A\ does\ not\ know\ PK_b}$$

$$\boldsymbol{A \rightarrow S : A, B}$$

$$\boldsymbol{S \rightarrow A : \{B, PK_b\}_{inv(PKa)}}$$

$$\boldsymbol{A \rightarrow B : \{N_a, A\}_{PK_b}}$$

$$\boldsymbol{if\ B\ does\ not\ know\ PK_a}$$

$$\boldsymbol{B \rightarrow S : B, A}$$

$$\boldsymbol{S \rightarrow B : \{A, PK_a\}_{inv(PKa)}}$$

$$\boldsymbol{B \rightarrow A : \{N_a, N_b\}_{PK_a}}$$

$$\boldsymbol{A \rightarrow B : \{N_b\}_{PK_b}}$$

Xabarlarni belgilash:Xabar bir nechta birlashtirilgan ma'lumotlardan iborat. Elementar ma'lumotlar quyidagilardir: ishtirokchilar (agent turi), nonces (matn turi), nosimmetrik kalitlar (simmetrik_kalit turi), ochiq kalitlar (ochiq_kalit turi), xesh funktsiyalari (hash_func turi), raqamlar (nat turi), mantiqiy (bool turi, doimiylar turi true va false), teglar (protokolli turdagi) va aloqa kanallari (kanal turi (dy)).

Bir nechta ma'lumotni birlashtirishga ruxsat beruvchi funktsiyalar quyidagilardir: birlashtirish (masalan: $\boldsymbol{PK_a}$ dan keyin $\boldsymbol{A}.\boldsymbol{PK_a}$ yoziladi), shifrlash (masalan: $\boldsymbol{K}$ bilan shifrlangan $\boldsymbol{M}$ $\{\boldsymbol{M}\}\_\{\boldsymbol{K}\}$ yoziladi), xesh funksiyasini qo‘llash (masalan: $\boldsymbol{H}$ ning $\boldsymbol{M}$ ga qo‘llanilishi, $\boldsymbol{H}$ $(\boldsymbol{M})$ yozilgan).

Boshlang‘ich yoki tuzilgan har qanday ma'lumot uchun ishlatilishi mumkin bo‘lgan umumiy tur (xabar) mavjud.

Bu ma'lumotlar konstantalar ($[a-z][A-Za-z\_]*$) bilan ifodalanadi va o‘zgaruvchilarda saqlanadi. ($[a-z][A-Za-z\_]*$) bu yerda X o‘zgaruvchisi uchun uning eski qiymatini (X) va yangi qiymatini (X') ko‘rsatish mumkin, bu o‘tishdan oldin ushbu o‘zgaruvchining qiymatini yoki uning yangi olingan qiymatidan foydalansak, bu o‘tish aniqlikka o‘tishga imkon beradi.

Ishtirokchining ro‘li. Protokolning har bir ishtirokchisi o‘z ro‘lini o‘ynaydi. Bu ro‘l HLPSLda mustaqil jarayon sifatida ko‘rsatilgan, ma'lumotni parametr sifatida qabul qiladi va aloqa kanallari orqali boshqa ro‘llar bilan bog‘lanadi.

ro‘le ro‘le-name (typed-parameters) played by player def=
local
declaration-of-local-variables
const
declaration-of-constants
init
initialization-of-variables
transition
list-of-transitions
end role

Har bir bo‘limning aniq sintaksisi quyidagicha:

• o‘zgaruvchilar deklaratsiyasi:

o‘zgaruvchi [, o‘zgaruvchi]* : turi [, o‘zgaruvchi [, o‘zgaruvchi]* : turi]*

• Konstantalar deklaratsiyasi:

doimiy [, doimiy]* : turi [, doimiy [, doimiy]* : turi]*

• o‘zgaruvchilarni initsializatsiya qilish:

o‘zgaruvchi: = ifoda [/\o‘zgaruvchi: = ifoda]*

• o‘tish:

teg. shart [/\ shart]* =|> harakat [/\ harakat]*

Masalan, NSPK Key Server protokolida ochiq kalitlar serverining ro‘li quyidagicha ko‘rsatilgan:

$$\begin{aligned}&\boldsymbol{role\ server}(\boldsymbol{S}:\boldsymbol{agent},\\&\quad \boldsymbol{PKs}:\boldsymbol{public_{key}},\\&\quad \boldsymbol{KeyMap}:(\boldsymbol{agent}.\boldsymbol{public_{key}})\boldsymbol{set},\\&\quad \boldsymbol{Snd},\boldsymbol{Rcv}:\boldsymbol{channel}(\boldsymbol{dy}))\\&\boldsymbol{played_{by}S}\ \mathbf{def}=\\&\ \boldsymbol{local}\ \boldsymbol{X},\boldsymbol{Y}:\boldsymbol{agent},\\&\quad \boldsymbol{PKy}:\boldsymbol{public_{key}}\end{aligned}$$

$$transition$$

$$req1.Rcv(X'.Y')/\backslash in(Y'.PKy', KeyMap) = | > Snd(\{Y'.PKy'\}\_inv(PKs))$$

$$end\ role$$

S-ni o‘ynaydigan ro‘llar server uning asosiy identifikatori va juftlarni (agent nomi, umumiy kalit) bilan birga, uning aktyorining nomini oladi. Shuningdek, u ikkita aloqa kanalini oladi: biri xabarlarni yuborish (Snd) va xabarlarni qabul qilish uchun (Rcv).

Ushbu ro‘l uchun mahalliy o‘zgaruvchilar kerak: ikkita agent nomi va bitta ochiq kalit.
U faqat bitta o‘tishni o‘z ichiga oladi, lekin uni bir necha marta bajarish mumkin, chunki uning aloqa holati ikkita agent nomidan iborat xabarni olishi va kalit kartasida ikkinchi agentning ochiq kalitiga ega bo‘lishdir. Ushbu o‘tishning noyob harakati agentning nomidan va uning ochiq kalitidan tashkil topgan xabarini jo‘natishidir, bularning barchasi serverning maxsus kaliti bilan shifrlangan.
Ushbu protokolda Alice ismli A ro‘lini bajarish juda qiyin, chunki uni qayta ishlash dastlabki bilimlarga bog‘liq.

role alice (A, B: agent,
PKa, PKs: public_key,
KeyRing: (agent.public_key) set,
Snd, Rcv: channel(dy))
played_by A def=
local State : nat,
Na, Nb: text,
PKb: public_key
init State := 0
transition
% Beginning, if A does not know yet the public key of B ask. State = 0 ∧ Rcv(start) ∧ not(in(B.PKb', KeyRing)) =|> State':= 1 ∧ Snd(A.B)
% Receiving the answer of the server learn. State = 1 ∧ Rcv({B.PKb'}_inv(PKs)) =|> State':= 2 ∧ KeyRing‘:=cons(B.PKb', KeyRing) ∧ Na':=new() ∧ Snd({Na'.A}_PKb') ∧ secret(Na',sna,{A,B}) ∧ witness(A,B,bob_alice_na,Na')
% Beginning/continuation, when A knows the public key of B knows. State = 0 ∧ Rcv(start) ∧ in(B.PKb', KeyRing) =|> State':= 2 ∧ Na':=new() ∧ Snd({Na'.A}_PKb') ∧ secret(Na',sna,{A,B}) ∧ witness(A,B,bob_alice_na,Na')
cont. State = 2 ∧ Rcv({Na.Nb'}_PKa) =|> State':= 3 ∧ Snd({Nb'}_PKb) ∧ request(A,B,alice_bob_nb,Nb')
end role

E'tibor berilsa xabarni qabul qilish boshlanishi protokolning bajarilishining boshlang‘ich nuqtasidir,

shuning uchun u faqat bitta ro‘lda ko‘rinishi kerak.

To‘plamdagi elementning tegishli va mansub bo‘lmaslik shartlari so‘radi va yorlig‘i bilan belgilangan o‘tishlarda qo‘llaniladi. To‘plamga element qo‘shish funksiyasi (minuslari) o‘rganish jarayonida qo‘llaniladi. To‘plamdan elementni olib tashlash funksiyasi (o‘chirish) ishlatilmaydi.

Maxfiylik, guvohlik va so‘rov predikatlari xavfsizlik xususiyatlarini aniqlash uchun ishlatiladi va ular mulkni deklaratsiyalashga bag‘ishlangan bo‘limda tushuntiriladi.

NSPK Key Server protokolida bu erda B ning ro‘li bob deb nomlangan:

ro‘le bob(A, B: agent,

PKb, PKs: public_key,

KeyRing: (agent.public_key) set,

Snd, Rcv: channel(dy))

played_by B def=

local State: nat,

Na, Nb: text,

PKa: public_key

init State := 0

transition

% Beginning if B does not know the public key of A ask. State = 0 ∧ Rcv({Na’.A}_PKb) ∧ not(in(A.PKa’, KeyRing)) =|> State’:= 1 ∧ Snd(B.A)

% Answer of the server learn. State = 1 ∧ Rcv({A.PKa’}_inv(PKs)) =|> State’:= 2 ∧ KeyRing‘:=cons(A.PKa’, KeyRing) ∧ Nb’:=new() ∧ Snd({Na.Nb’}_PKa’) ∧ secret(Nb’,snb,{A,B}) ∧ witness(B,A,alice_bob_nb,Nb’)

% Beginning/continuation when B knows the public key of A knows. State = 0 ∧ Rcv({Na’.A}_PKb) ∧ in(A.PKa’, KeyRing) =|> State’:= 2 ∧ Nb’:=new() ∧ Snd({Na’.Nb’}_PKa’) ∧ secret(Nb’,snb,{A,B}) ∧ witness(B,A,alice_bob_nb,Nb’)

cont. State = 2 ∧ Rcv({Nb}_PKb)

=|> State’:= 3 ∧ request(B,A,bob_alice_na,Na)

end role

Protokol ishtirokchilarining ro‘llarining ushbu misollarida o‘tish to‘g‘risidagi ma'lumotlarni umumlashtirish uchun quyidagi shartlar mavjud:

• comparison: expression = expression

taqqoslash : ifoda = ifoda,

• receiving a message: Rcv(message)

qabul qiluvchi xabar: Rcv(xabar)

• test of membership to a set: in(element,set) or non membership: not(in(element,set)) and the actions

are

o‘rnatish uchun aksessuarlar tekshiruvi: ichida (element, to‘siq) yoki yo‘q a'zolik: yo‘q (ichida (element, to‘siq)) va harakatlar quyidagilardir:

• assignment: variable’ := expression

maqsad: o‘zgaruvchi': = ifoda

• creation of fresh information: variable’ := new()

yangi ma'lumot yaratish: o‘zgaruvchi': = yangi()

• sending a message: Snd(message)

xabar yuborish: Snd (xabar)

• adding an element to a set: set’ := cons(element,set)

to‘plamga elementni qo‘shish: qo‘shish ': = cons (element, to‘siq)

Protokol sessiyasi

Ishtirokchilarning ro‘li aniqlanganda, ularni protokol sessiyasini qurish uchun ularni qanday qilib birlashtirishni tasvirlashimiz kerak.

Nspk kalit serveriga misol uchun, bu ro‘l aniqlanadi:

```
ro‘le nspk(A, B: agent,
          PKa, PKb, PKs: public_key,
          KeyMapA, KeyMapB: (agent.public_key) set,
          Snd, Rcv: channel(dy))
def=
         composition
           alice(A,B,PKa,PKs,KeyMapA,Snd,Rcv)
           /\ bob(A,B,PKb,PKs,KeyMapB,Snd,Rcv)
end role
```

Ushbu misolda sessiya Alice ro‘li va Bobning ro‘li tarkibiga kiradi. Ro‘llarda server bu darajada boshlamaydi, chunki u barcha protokol seanslari uchun keng tarqalgan.

Atrof-muhit va Senariyning tavsifi

Protokol to‘liq aniqlangan bo‘lsa, hali ham ushbu protokol tahlil qilinadigan muhitni (shu jumladan, buzg‘unchining dastlabki ma'lumotlarini) va bajarilishi kerak bo‘lgan senariyni, ya'ni parallel ravishda o‘tkaziladigan sessiyalar misollarini aniqlash kerak.

Parametr sifatida ro‘llarga uzatiladigan ma'lumotlar doimiydir (aloqa kanallaridan tashqari).

Misol uchun NSPK Key Server uchun muhitni quyidagicha tasvirlash mumkin:

```
ro‘le environment() def=
  local KeyMapS,KeyMapA,KeyMapB,KeyMapI: (agent.public_key) set,
          Snd, Rcv: channel(dy)
```

```
  const a, b, s, i: agent,
        pka, pkb, pki, pks: public_key,
        sna, snb, alice_bob_nb, bob_alice_na: protocol_id
init KeyMapS := {a.pka, b.pkb, i.pki}
        /\ KeyMapA := {a.pka, b.pkb}
        /\ KeyMapB := {b.pkb}
        /\ KeyMapI := {i.pki}
intruder_knowledge = {a, b, s, pks, pka, pkb, pki, inv(pki)}
  composition
        server(s,pks,KeyMapS,Snd,Rcv)
        /\ nspk(a,b,pka,pkb,pks,KeyMapA,KeyMapB,Snd,Rcv)
        /\ nspk(a,i,pka,pki,pks,KeyMapA,KeyMapI,Snd,Rcv)
        /\ nspk(i,b,pki,pkb,pks,KeyMapI,KeyMapB,Snd,Rcv)
 end role
```

Buzg`inchining oldingi nomi (i) bor va uning dastlabki bilimlari agentlarning nomlari, ushbu agentlarning ochiq kalitlari va a o‘zlari uchun ochiq kalitlari ($\boldsymbol{pki}$) va xususiy kalitlar ($\boldsymbol{inv(pki)}$). Ta'riflangan senariy uchta protokol seanslari bo‘lgan serverning bir qismidir: birinchi sessiyada futbolchilar a va b, ikkinchisida futbolchilar a va buzg`inchi, uchinchi sessiyada esa o‘yinchilar buzg`inchi va b.

Xavfsizlik xususiyatlari deklaratsiyasi

Xavfsizlik xususiyatlari maqsad deb nomlangan bo‘limda tasvirlangan bo‘lishi kerak. Agar egalik aniqlanmagan bo‘lsa, bu bo‘lim mavjud bo‘lmasligi kerak.

Xavfsizlik xususiyatlari uch xil:

Axborotning maxfiyligi: e'lon maxfiy kalit so‘z bilan amalga oshiriladi,doimiy identifikator; ushbu identifikator axborotni ishlab chiqaradigan (yoki birinchi marta uzatiladigan) ro‘lda ishlatiladi.

secret(information,identifier,agents-set)

Maxsus maxfiylikda:yashirin (axborot, identifikator, agentlar to‘plami)

Agentlar to‘plami ma'lumotni bilishga ruxsat berilgan agentlar to‘plamidir.

Agent X ning boshqa Y agenti tomonidan ba'zi ma'lumotlar uchun qattiq autentifikatsiya qilish: paro‘lni hal qilish kalit so‘z bilan amalga oshiriladi, autentifikatsiya qilish on, keyin doimiy identifikator; ushbu identifikator autentifikatsiya bilan bog‘liq ro‘llarda ishlatiladi; birinchidan, autentifikatsiya qilingan X agenti sifatida, sintaksisi bilan predikat guvohi:

witness(X,Y,identifier,information)

ikkinchidan, autentifikatsiya qilishni amalga oshiruvchi Y agenti sifatida, sintaksisi bilan oldindan so‘rashda:

request(Y,X,identifier,information)

Ushbu autentifikatsiya printsipi shundan iboratki, so‘rov yuborilganda, xuddi shu ma'lumot uchun yuborilgan tegishli guvoh bo‘lishi kerak.

Agent X ning zaif autentifikatsiyasi boshqa Y agenti tomonidan ba'zi ma'lumotlar uchun: reklama kalit so‘z bilan amalga oshiriladi zaif autentifikatsiya on, keyin doimiy identifikator ishlatiladi, masalan, witness va wrequest predmetlari bilan agentlarning ro‘llarida kuchli autentifikatsiya qilish uchun (iltimos, ushbu oxirgi predmetning mavjudligi

boshqa ism).

Zaif autentifikatsiya qilish bilan, predikat guvohi bir nechta wrequest predmetlari uchun ishlatilishi mumkin, bu erda faqat bitta so‘rovni oldindan belgilash uchun ishlatilishi mumkin bo‘lgan qattiq autentifikatsiyadan farqli o‘laroq.

NSPK kalit serveri misolida to‘rtta xavfsizlik xususiyatlari e'lon qilindi: ikkita maxfiylik xususiyatlari va ikkita qattiq autentifikatsiya xususiyatlari.

goal

secrecy_of sna, snb

authentication_on alice_bob_nb

authentication_on bob_alice_na

end goal

Senariyni bajarish

Muhitda tasvirlangan skriptning bajarilishini ifodalovchi asosiy ro‘lni bajarish spetsifikatsiya faylining oxirida atrof-muhit ro‘lini chaqirish orqali amalga oshiriladi.

environment()

ASBOBLARDAN FOYDALANISH

hlpsl2if

Hlpsl2if vositasi kengaytmali faylda parametr sifatida belgilangan protokolning spetsifikatsiyasini tuzadi .hlpsl va spetsifikatsiyalarda aniqlangan xatolar ro‘yxati yoki bir xil nomdagi faylni yaratadi, lekin kengaytma bilan keyinchalik tahlil qilinadigan spetsifikatsiya mavjud bo‘lsa.

Usage: hlpsl2if [option] file.hlpsl

Faqat foydali variant - ajratish. Bu ishlab chiqarish bo‘limida ko‘rsatilgan har bir xavfsizlik xususiyati uchun turli IF fayllari imkonini beradi. Bu har bir xavfsizlik xususiyatini alohida tahlil qiladi. Unutmang, bu derleyici xususiyatlari barcha xatolarni topish mumkin emas, ayniqsa, ba'zi semantik xatolar, boshqa tahlil vositasi bilan aniqlash kerak bo‘ladi.

ATSE

ATSE vositasi IF fayli tomonidan berilgan protokol senaryosida hujumlarni izlaydi. Agar hujum aniqlanmasa, bu xavfsizlik xususiyatlari har doim kafolatlangan degani emas, balki faqat ma'lum bir skript

uchun mo‘ljallangan.

Usage: atse [options] file.if

Asosiy foydali variantlar quyidagilardir:

-- if kerak bo‘lsa, bu fayl IF formatidagi spetsifikatsiyadir.

-- of if tavsiya etilsa, displey IFC formatiga mos kAlicehi kerak degan ma'noni anglatadi.

-- noexec xavfsizlik xususiyatlarini tahlil qilmaydi, lekin ijro skriptini aniq shaklda ko‘rsatadi; ushbu parametr semantik xatolarni aniqlash uchun juda foydalidir, masalan, hech qachon ishga tushirilmagan o‘zgaruvchini ishlatish (xayoliy sobit bilan ifodalanadi)...), yoki uning yangi qiymati o‘rniga o‘zgaruvchining eski qiymatini foydalanish (ramzi 0 hlpsl xususiyatlari unutilgan edi).

-- typed--untyped tahlilning tipiklangan rejimda (standart) bajarilganligini yoki yo‘qligini aniqlashga imkon beradi; atipik rejim ba'zan turlarning tartibligiga asoslangan ko‘proq hujumlarni aniqlash uchun foydalidir.

-- out file.atk , ushbu faylda aniqlangan hujumning izini (agar mavjud bo‘lsa) saqlaydi.

-- nb max it nb ,agar spetsifikatsiya bir xil o‘tishni bir necha marta qo‘llashga ruxsat etilsa, sukut bo‘yicha bu iteratsiya soni 3 bilan cheklangan; ushbu parametr bu maksimal iteratsiya sonini o‘zgartirish imkonini beradi.

-- short, qadamlar soni bo‘yicha eng qisqa hujumni qidirmoqda.

-- ns sukut bo‘yicha hujumni kuzatish soddalashtirilishi mumkin va ba'zi qadamlar yo‘q bo‘lishi mumkin; ushbu parametr buni oldini oladi va to‘liq izni ko‘rsatadi.

Quyidagi misolda a va b o‘rtasida faqat bitta protokol sessiyasi mavjud bo‘lganda Alice ro‘lini talqin qilish ko‘rsatilgan.

-- noexec.

```
Role alice played by (a,4):
|       Choice Point
|        | start => a.b
|        |       & Test b.PKb(8) not in set_95;
|        |       & Built from step_1
|        | {b.PKb(9)}_(inv(pks)) => {n9(Na).a}_PKb(9)
|        |       & Secret(n9(Na),(),set_106); Witness(a,b,bob_alice_na,n9(Na));
|        |       & Add b.PKb(9) to set_95; Add a to set_106; Add b to set_106;
|        |       & Built from step_2
|        | {n9(Na).Nb(10)}_pka => {Nb(10)}_PKb(9)
|        |       & Request(a,b,alice_bob_nb,Nb(10));
|        |       & Built from step_4
|       Or
|        | start => {n14(Na).a}_PKb(14)
|        |       & Secret(n14(Na),(),set_107); Witness(a,b,bob_alice_na,n14(Na));
```

| | & Test b.PKb(14) in set_95; Add a to set_107; Add b to set_107;

| | & Built from step_3

| | {n14(Na).Nb(15)}_pka => {Nb(15)}_PKb(14)

| | & Request(a,b,alice_bob_nb,Nb(15));

| | & Built from step_4

Ta'kidlash kerak bo‘lgan muhim eslatmalar quyidagilar:

Ushbu ro‘l 4 raqami ostida shaxs tomonidan ijro etiladi.

Tanlov nuqtasi boshidan belgilanadi, chunki vaziyatga qarab, birinchi ikkita o‘tish qizil bo‘lishi mumkin.

Yangi ma'lumotni aniqlash o‘zgaruvchining nomi, keyin identifikatsiya raqami bilan yoziladi. Masalan: $\boldsymbol{PKb}\ (\mathbf{8})\ \boldsymbol{yoki}\ \boldsymbol{Nb}(\mathbf{10})$.

Yangi axborotni yaratish o‘zgaruvchining nomiga yangi funktsiyani qo‘llash orqali yoziladi. Masalan: $\boldsymbol{n9}\ (\boldsymbol{Na})$.

Agar protokolning talqini dummy kabi qiymatlarni o‘z ichiga olgan bo‘lsa,bu o‘zgaruvchining ishlatilganligini anglatadi, hech qanday qiymatga ega bo‘lmasalikni ko‘rsatadi. Ko‘rsatilgan qiymat-o‘zgaruvchining turi nomi, undan oldin dummy so‘zi (masalan: dummy nonce matn turi uchun, xayoliy agent, xayoliy to‘siq, turi xabar uchun soxta xabar va boshqalar).

Ushbu belgilar hujumlarning izlarida ham qo‘llaniladi. Shuning uchun, nspk protokoli kalitlari server ushbu qo‘llanmada ko‘rsatilganidek tahlil qilinganda, natija quyidagicha:

SUMMARY

UNSAFE

DETAILS

TYPED MODEL

BOUNDED SPEC. READING DEPTH

PROTOCOL

nspk-ks.if

GOAL

Secrecy goal () on n19(Nb))

BACKEND

CL-AtSe

STATISTICS

Analysed : 16394 states

Reachable : 10174 states

Translation: 0.03 seconds

Computation: 1.48 seconds

ATTACK TRACE

```
i -> (a,7): start
        & Test i.PKb(28) not in set_95;
(a,7) -> i: a.i
        & Built from step_1
i -> (s,2): X(1).i
        & Test i.pki in set_94;
(s,2) -> i: {i.pki}_(inv(pks))
        & Built from step_0
i -> (s,2): X(2).a
        & Test a.pka in set_94;
(s,2) -> i: {a.pka}_(inv(pks))
        & Built from step_0
i -> (a,7): {i.pki}_(inv(pks))
(a,7) -> i: {n29(Na).a}_pki
        & Secret(n29(Na),(),set_125); Add i.pki to set_95;
        & Add a to set_125; Add i to set_125;
        & Built from step_2 i ->
(b,5): {n29(Na).a}_pkb
        & Test a.PKa(18) not in set_96;
(b,5) -> i: b.a
        & Built from step_5
i -> (b,5): {a.pka}_(inv(pks))
(b,5) -> i: {n29(Na).n19(Nb)}_pka
        & Secret(n19(Nb),(),set_118); Witness(b,a,alice_bob_nb,n19(Nb));
        & Add a.pka to set_96; Add a to set_118; Add b to set_118;
        & Built from step_6
i -> (a,7): {n29(Na).n19(Nb)}_pka
(a,7) -> i: {n19(Nb)}_pki
        & Built from step_4
```

E'tibor bering, tajovuzkor barcha xabarlar uchun vositachi sifatida ishlaydi. Buning sababi, protokolni bajarish ochiq deb hisoblanadi,ya'ni tajovuzkor xohlagan narsani bajarishi mumkin. Shunday qilib, tajovuzkor xabarni ushlab turadigan yoki ushlamagan hollarni ko‘rib chiqish o‘rniga, har bir xabar to‘g‘ridan-to‘g‘ri unga uzatiladi va xabarni o‘zgartirilgan versiyani tarqatadi yoki xabarni tarqatmaydi.

Topshiriq

HLPSL tilida sodda protokollarni ifodalang va SPAN+AVISPA vositasida yuklang.

Nazorat savollari

1. secret(information,identifier,agents-set) buyrug‘ining vazifasini tushuntirib bering
2. witness(X,Y,identifier,information) buyrug‘ining vazifasini tushuntirib bering
3. request(Y,X,identifier,information) buyrug‘ining vazifasini tushuntirib bering

2-amaliy ish. HLPSL tilida kalitlarni almashinish protokollarini ifodalash va SPAN+AVISPA vositasida yuklash (2 soat)

Faraz qilaylik, A va B maxfiy K kalitiga ega bo‘lsa(ya'ni K faqat A va B ga ma'lum bo‘lgan qiymat). Oldin almashilgan K1 kalitni olish uchun quyidagi protokolni ko‘rib chiqilgan.

$A \rightarrow B : \{Na\}_K$

$B \rightarrow A : \{Nb\}_K$

$A \rightarrow B : \{Na\}_{K1}$ qayerda $\mathrm{K1} = \mathrm{Hash}(\mathrm{Na}.\mathrm{Nb})$

Alice-Bob yozuvida bu shunday ko‘rinadi: A B ga K bilan shifrlangan nonces Na yuboradi va B keyin A ga K bilan shifrlangan boshqa nonce Nb yuboradi. Nihoyat, A birlashtirilgan Na va Nb qiymatini xeshlash orqali yangi K1 kalitini hisoblab chiqadi va K1 bilan shifrlangan Nb qiymatini B ga qaytarib yuboradi.

Protokol namunasining maqsadlari keyingi misolda xavfsizlik maqsadlarini aniq modellashtirishni muhokama qilamiz, lekin protokolining mo‘ljallangan xavfsizlik maqsadlarini umumlashtiriladi. Bu (o‘yinchoq) kalit almashish protokoli bo‘lib, unda dastlabki ikkita xabar asosiylaridan birini o‘rnatish uchun xizmat qiladi va oxirgisi A yangi kalitga ega ekanligini isbotlaydi. Bu misol uchun birinchi maqsad bir tomonlama autentifikatsiya: ya'ni B A ni Nbda (oxirgi xabarda) autentifikatsiya qiladi, boshqacha qilib aytganda: B uchinchi xabarni olganda, Nb A tomonidan yuborilganiga ishonch hosil qilishi mumkin. Bundan tashqari, kuchli autentifikatsiyani talab qilinadi, takroriy hujumlarni istisno qiladigan zaif autentifikatsiya deb ataladigan kengaytma. Shunday qilib, shuningdek, agar kuchli autentifikatsiyaga erishilsa, Nb ilgari B tomonidan qabul qilinmagan degan xulosaga kelib chiqilgan. Ikkinchi xavfsizlik maqsadi sifatida yangi K1 kaliti sir saqlanishi kerak.

```
role alice(.., K: symmetric_key,
        % K and Hash must be passed to each Hash: hash_func,
        % role, so that A and B agree on them. ..)
 ... def=
        local
                ... State : nat
```

% This variable is typically defined in all roles.

init

State := 1

transition

1. *State = 1 /\ RCV(start) =|>*
   State’:= 2 /\ Na’ := new() /\ SND({Na’}_K)
2. *State = 2 /\ RCV({Nb’}_K) =|>*
   State’:= 3 /\ SND({Nb’}_Hash(Na.Nb’))

Muhokama: Umumiy bilimlarni modellashtiriladi. A va B K ning qiymati bo‘yicha oldindan kelishibolgan deb taxmin qilinadi. Shuningdek, ular K1 ni yaratish uchun qaysi kriptografik xesh funksiyasidan foydalanishlari haqida kelishib olishlari kerak. Seansda ishtirok etishi kerak bo‘lgan alice va bob ro‘liga bir xil qiymatlarni o‘tkazish orqali oldingi bilim almashishni modellashtiriladi. Quyidagi to‘liq misolda ko‘rsatilganidek, kompozitsion rolni chaqirish: a va b o‘rtasidagi birinchi seans K qiymati uchun kab kalitidan, a va i hujumchi o‘rtasidagi ikkinchi seans esa kai dan foydalanadi. Har uch seans ham bir xil xesh funktsiyasidan foydalanilgan. Birgalikda ro‘llarda o‘zgaruvchilarning nomlari bir xil bo‘lishi shart bo‘lmasada, ularga bir xil nom berish amalga oshirilgan.

Muhokama: o‘tishlar birinchi o‘tish nisbatan aniq, lekin u HLPSL ning muhim xususiyatini ko‘rsatadi: ya'ni yangi ma'lumotlarni yaratishni qanday modellashtirishni boshlash bu Alice uchun protokolni ishga tushirish uchun signaldir. U nonce Na uchun uni new() ga belgilash orqali yangi qiymat yaratadi, bu intuitiv ravishda qiymat tasodifiy hosil bo‘lishini anglatadi. Shuningdek, new() ni ixtiyoriy turdagi ma'lumotlargada, masalan simmetrik_kalit turdagi yangi qiymatlarni yaratish uchun qo‘llash mumkinligini ta'kidlaymiz.

Shifrlangan qiymatni SND deb nomlangan kanalga kiritishdan oldin u bu qiymatni K kaliti yordamida shifrlaydi. Ushbu o‘tishdan keyin Alice 2-holatda.

Ikkinchi o‘tish qiyinroq. Birinchidan, Alice {Nb'}_K xabarini oladi. Agar Alice 2-holatda bo‘lsa va bu xabar {*}_K ko‘rinishida bo‘lsa, ba'zi bir qiymat * uchun alice Nb ni K ostida shifrlangan qabul qilingan qiymat sifatida belgilaydi. Xuddi shu o‘tishda yangi olingan qiymat yangi sifatida saqlanadi va Nb ning yangi qiymati yana yuboriladi, Hash(Na.Nb') kaliti bilan shifrlangan bo‘lib, u Na va Nb' ikki qiymatlarining birlash yo‘li bilan hisoblanadi.

Misol uchun to‘liq yechim quyida keltirilgan. Ushbu spetsifikatsiya sir, guvoh va so‘rov atamalarini o‘z ichiga oladi (bularning barchasi xavfsizlik maqsadlarini tavsiflash bilan bog‘liq).

1-misol:

ro‘le alice(

A,B : agent,

```
        K : symmetric_key,
        Hash : hash_func,
        SND,RCV : channel(dy))
         played_by A def=
        local
         State : nat,
        Na,Nb : text,
        K1 : message
         init
        State := 0
        o‘tish
        1. State = 0 /\ RCV(start) =|>
           State’:= 2 /\ Na’ := new()
           /\ SND({Na’}_K)
        2. State = 2 /\ RCV({Nb’}_K) =|>
           State’:= 4 /\ K1’ := Hash(Na.Nb’)
           /\ SND({Nb’}_K1’)
           /\ witness(A,B,bob_alice_nb,Nb’)
    end role
```

ro‘le bob(

```
    A,B : agent,
     K : symmetric_key,
     Hash : hash_func,
    SND,RCV : channel(dy))
    played_by B def=
local
    State : nat,
    Nb,Na : text,
    K1 : message
    Init
     State := 1
    o‘tish
    1. State = 1 /\ RCV({Na’}_K) =|>
       State’:= 3 /\ Nb’ := new()
       /\ SND({Nb’}_K)
```

/\ K1':= Hash(Na'.Nb')

/\ secret(K1',k1,{A,B})

2. *State = 3 /\ RCV({Nb}_K1) =|>*

   State':= 5 /\ request(B,A,bob_alice_nb,Nb)

   end role

   ——————————————————————————————— ro‘le

   session(

   A,B : agent,

   K : symmetric_key,

   Hash : hash_func)

   def= local SA, SB, RA, RB : channel (dy)

   Kampozitsiya

   alice(A,B,K,Hash,SA,RA)

   /\ bob (A,B,K,Hash,SB,RB)

end role

——————————————————————————————— ro‘le

environment()

def=

const

bob_alice_nb,

k1 : protocol_id,

kab,kai,kib : symmetric_key,

a,b : agent,

h : hash_func

intruder_knowledge = {a,b,h,kai,kib}

Kampozitsiya

session(a,b,kab,h)

/\ session(a,i,kai,h)

/\ session(i,b,kib,h)

end role

——————————————————————————————————goal

secrecy_of k1

authentication_on bob_alice_nb

end goal

——————————————————————————————— environment()

Ushbu misolda AVISPA Toolni ishga tushirish quyidagi natijani beradi:

```
% avispa ex1.hlpsl
 % ofmc
 % version of 2005/06/07 (versiyasi)
summary (xulosa)
        safe (xavfsiz)
details (tafsilotlar)
        bounded_number_of_sessions

PROTOCOL
./ex1.if
GOAL
        as_specified
BACKEND
        OFMC
COMMENTS
STATISTICS
        parseTime: 0.00s
        searchTime: 0.16s
        visitedNodes: 105 nodes
        depth: 8 plies
```

Yuqorida aytib o‘tgan asbob standart parametrlar bilan chaqirilganda OFMC backendni chaqiradi.

OFMC hech qanday hujum topmaganligini ko‘rish mumkin. Boshqacha qilib aytganda, belgilangan xavfsizlik maqsadlari atrof-muhit ro‘li bilan belgilangan cheklangan miqdordagi sessiyalar uchun bajarilgan. AVISPA asbobi cheklangan stsenariyga alternativalarni qo‘llab-quvvatlaydi (ya’ni cheklangan miqdordagi aniq seanslar)

Topshiriq

Biror kalit almashish protokolini HLPSL tilida SPAN+AVISPA vositasida yuklang, tekshiring va xavfsizlik talablariga tekshiring.

Nazorat savollari

1. Kalit almashish protokollariga misolllar keltiring?
2. HLPSL tilida SPAN+AVISPA vositasida kalit almashish protokolini yuklash jarayonini tushuntiting

3-amaliy ish. HLPSL tilida turli zamonaviy protokollarni ifodalash va SPAN+AVISPA vositasida yuklash (2 soat)

Ushbu misol Endryu Secure RPC protokolini ko‘rib chiqilgan.

$A -> B : A.\{Na\}\_Kab$

$B -> A : \{Na + 1.Nb\}\_Kab$

$A -> B : \{Nb + 1\}\_Kab$

$B -> A : \{K1ab.N1b\}\_Kab$

Protokol ikkala tomonni bir-biriga autentifikatsiya qilish va keyingi aloqa uchun ishlatilishi mumkin bo‘lgan yangi umumiy K1ab kalitini o‘rnatish uchun ishlatiladi. N1b qiymati kelajakda foydalanish uchun yuboriladi. K1ab ham, N1b ham B tomonidan yaratilgan.

Ushbu protokolni modellashtirish oldingi misollarimizga qaraganda ancha qiyin. Misol uchun, operator + turini o‘z ichiga olgan holda ilgari duch kelmagan narsadir.

HLPSL ixtiyoriy arifmetik operatorlarni qo‘llab-quvvatlamaydi;. Masalan, + kabi operatorning kiritilishi ilgari ko‘rmagan narsadir.

HLPSL ixtiyoriy arifmetik operatorlarni qo‘llab-quvvatlamaydi; ammo, xavfsizlik nuqtayi nazaridan eng muhim bo‘lgan xususiyatlarni aks ettiradigan qo‘shimchaning taxminiy modelini yaratishimiz mumkin.

Xususan, qo‘shishdan foydalanish o‘rtasidagi farq shundaki, tajovuzkor 1 ni ayirish orqali qo‘shishni osongina o‘zgartira oladi, esa kriptografik xeshni o‘zgartira olmaydi deb taxmin qilamiz. + kabi operator uchun inversiyani ahamiyat hisoblash murakkabligi bilan hisoblash mumkinligi aniq. Biroq, bu holda, natijalari hech qachon shifrlanmagan holda yuborilmaydi va buzg‘unchi K ni bilmasdan yoki kriptografiyani buzmasdan {Na} _K berilgan {Na+1}_K ni hisoblay olmaydi, degan oqilona taxmin. Shuning uchun modellashtirish taxminini keltiramizki, agentlar Na ning qandaydir funksiyasini oddiygina hisoblashi kerak, bu oson invertatsiya qilinishi shart emas. Bunday holda, HLPSL-da funktsiya belgisidan foydalanishimiz mumkin, ya'ni hash func tipidagi qiymat. Ushbu misol uchun Succ deb nomlangan voris funksiyani aniqlaymiz. Bu funksiya buzg‘unchiga ma'lum bo‘ladi, shuning uchun u o‘zi biladigan qiymatlarning vorislarini hisoblay oladi, lekin Succ(Na) ko‘rinishidagi qiymatlarni invert qilmaydi (agar u Na ni bilmasa).

role alice (A, B : agent,

Kab : symmetric_key,

Succ : hash_func,

SND, RCV : channel(dy))

played_by A

def=

local

State : nat,

Na,Nb : text,

K1ab : symmetric_key,

N1b : text

const alice_bob_k1ab, alice_bob_na, bob_alice_nb: protocol_id

init State := 0

transition

1. *State = 0 /\ RCV(start) =|>*
   State'= 2 /\ Na' := new()
   /\ SND(A.{Na'}_Kab)
1. *State = 2 /\ RCV({Succ(Na).Nb'}_Kab) =|>*
   State'= 4 /\ SND({Succ(Nb')}_Kab)
   /\ witness(A,B,bob_alice_nb,Nb')
2. *State = 4 /\ RCV({K1ab'.N1b'}_Kab) =|>*
   State'= 6 /\ request(A,B,alice_bob_k1ab,K1ab')
   /\ request(A,B,alice_bob_na,Na)

end role

———————————————————————————————————*ro'le bob (A, B : agent,*

Kab : symmetric_key,

Succ : hash_func,

SND, RCV : channel(dy))

played_by B

def=

local State : nat,

Nb,Na,N1b : text,

K1ab : symmetric_key

init State := 1

Transition

1. *State = 1 /\ RCV(A.{Na'}_Kab) =|>*
   State'= 3 /\ Nb' := new()
   /\ SND({Succ(Na').Nb'}_Kab)
   /\ witness(B,A,alice_bob_na,Na')
2. *State = 3 /\ RCV({Succ(Nb)}_Kab) =|>*
   State'= 5 /\ N1b' := new()

```
        /\ K1ab' := new()
            /\ SND({K1ab'.N1b'}_Kab)
            /\ witness(B,A,alice_bob_k1ab,K1ab')
            /\ request(B,A,bob_alice_nb,Nb)
            /\ secret(K1ab',k1ab,{A,B})
            /\ secret(N1b',n1b,{A,B})
end role
ro'le session(A, B : agent,
            Kab : symmetric_key,
            Succ : hash_func)
def=
    local SAB, RAB,
     SBA, RBA : channel (dy)
o'tish
    alice(A, B, Kab, Succ, SAB, RAB)
/\ bob (A, B, Kab, Succ, SBA, RBA)
end role
ro'le environment()
    def=
     const alice_bob_k1ab, alice_bob_na, bob_alice_nb,
        n1b, k1ab : protocol_id,
        a, b : agent,
        kab, kai, kib : symmetric_key,
        succ : hash_func
intruder_knowledge = {a, b, kai, kib, succ}
composition
goal
    secrecy_of n1b, k1ab
    authentication_on bob_alice_nb
    authentication_on alice_bob_na
    authentication_on alice_bob_k1ab
end goal
environment()
```

Munozara va tahlil natijalari

Xavfsizlik maqsadlari spetsifikatsiyasi xavfsizlik maqsadlari HLPSLda asosiy rol

o‘tishlarini maqsad faktlari deb ataladigan faktlar bilan to‘ldirish va keyin ularga qiymat berish orqali belgilanadi. HLPSL maqsad bo‘limidagi tavsiflar qanday shartlar - ya'ni bunday faktlarning qanday kombinatsiyasi - hujumni ko‘rsatadi. Ko‘rib turganimizdek, buning oddiy misoli maxfiylik bo‘lib, bu yerda maqsadli faktlar qaysi ma’lumotlar kimlar o‘rtasida sir bo‘lishi kerakligini ko‘rsatadi va maqsad bo‘limidagi maqsadli deklaratsiya qiladi, agar tajovuzkor maxfiy qiymatni o‘rgansa va u aniq emasligini tasvirlaydi, u va kimdir boshqalar o‘rtasidagi sir, bu hujum deb hisoblanishi kerak. Ichki tomondan, hujum shartlari vaqt mantig‘i nuqtayi nazaridan ko‘rsatilgan, ammo ikkita eng muhim shart uchun foydali va qisqacha makrolar taqdim etiladi.

Endryu Secure RPC protokoli ham maxfiylik, ham o‘zaro autentifikatsiyani ta'minlashi kerak. Almashtirilgan K1ab kaliti va hosil bo‘lmagan N1b A va B o‘rtasida sir saqlanishiga ishonch hosil qilishni kerak. Bunga birinchi navbatda spetsifikatsiyamizga maqsadli faktlarni qo‘shish orqali erishiladi: bu holda, qaysi qadriyatlar saqlanishi kerakligini ko‘rsatadigan maqsadli faktlar, sir va qaysi agentlarga bunday sirlarni bilishga ruxsat berilgan.

DA modellashtirishda, odatda, bunday maxfiy faktlarni maxfiy bo‘lishi kerak bo‘lgan qiymatni yaratadigan rolga qo‘yish tavsiya etiladi. Ushbu misolda bob roli ikkala qiymatni yaratadi, shuning uchun uning oxirgi o‘tishini (3 bilan belgilangan) quyidagi maxfiy faktlar bilan yakunlanadi. (bu erda yangi K1ab va N1b qiymatlarini ifodalash uchun tub raqamlar kerak bo‘ladi)

$$/\backslash\ secret(K1ab', k1ab, \{A, B\})$$

$$/\backslash\ secret(N1b', n1b, \{A, B\})$$

Bular B ikki qiymatni (faqat) A va B o‘rtasida taqsimlanishiga ruxsat berishini ko‘rsatadi. Agar bob buzg‘unchi bilan sessiyada (masalan, yuqoridagi muhit ro‘lida ko‘rsatilgan uchinchi seans) ishtirok etsada A = i bo‘ladi. Bunday hollarda, buzg‘unchiga maxfiy deb e'lon qilingan qiymatni bilishga ruxsat beriladi. Yashirin faktlarning doimiy ikkinchi argumenti protokol identifikatorlari deb ataladi va atrof-muhit ro‘lining const qismida protocol_id turida e'lon qilinishi kerak. Protokol identifikatori uchun maxfiylik fakti nazarda tutilgan o‘zgaruvchining nomini kichik harflarda ishlatish bo‘yicha modellashtirish konventsiyasini qabul qilingan. Maxfiylik faktlari bo‘lsa, protokol identifikatorlari faqat turli xil maxfiylik maqsadlarini farqlash uchun xizmat qiladi. Bu keyinchalik tahlilni osonlashtiradi, masalan, agar biror kishi ma'lum bir tajriba uchun K1ab ni e'tibor qoldirib, faqat N1b maxfiyligini tekshirmoqchi bo‘lsa, maqsad bo‘limida tegishli k1ab maxfiyligini izohlash mumkin.

Maqsad faktlari shunchaki protokol modelidagi voqealardir. Faktlarga qo‘shimcha ravishda, ushbu faktlarni qanday izohlash kerakligini aniqlab olishimiz kerak: maqsadli faktlarning qaysi kombinatsiyasi qonuniy va ma'qul va qaysi biri hujum deb hisoblanishi kerak?

Bu HLPSL spetsifikatsiyasining maqsad bo‘limida amalga oshiriladi. Maxfiylik uchun

shunchaki yozishimiz kerak.

$secrecy\_of\ k1ab, n1b$

Bu, yuqorida aytib o‘tilganidek, aslida vaqtinchalik mantiq formulasi uchun s o‘rinlidir. Intuitiv ravishda, u ikkinchi pozitsiyada k1ab yoki n1b ni o‘z ichiga olgan maxfiy faktlarning birinchi pozitsiyasida paydo bo‘ladigan har qanday qiymatlar maxfiy ekanligini bildiradi. Agar tajovuzkor bunday qiymatni o‘rgansa va u tegishli maxfiy faktning uchinchi pozitsiyasida berilgan to‘plamda bo‘lmasa, bu hujumni anglatadi.

Autentifikatsiya Guvoh va so‘rov hodisalari autentifikatsiya bilan bog‘liq maqsadli faktlardir. Ular direktorning o‘z tengdoshi mavjudligiga ishonishda haqligini tekshirish uchun ishlatiladi.

Ular har doim bir xil uchinchi parametrga ega bo‘lgan juftliklarda paydo bo‘ladi. Protokolning maqsad bo‘limida. Bu misolda, masalan, ikki ishtirokchi almashtirilgan K1ab kalitining qiymati haqida albatta kelishib olishlari kerak. Xususan, Alice bu qiymat haqiqatan ham Bob tomonidan yaratilganiga, u umumiy kalit sifatida foydalanish maqsadida yaratilganiga va avvalgi sessiyada takrorlanmaganiga ishonch hosil qilishni xohlaydi.

$$/\backslash\ request(A, B, alice\_bob\_k1ab, K1ab')$$

Alicening oxirgi o‘tishini quyidagicha o‘qiydi: "agent A K1ab' qiymatini qabul qiladi va endi B agenti mavjudligi kafolatiga tayanadi va u bilan ushbu qiymat bo‘yicha rozi bo‘ladi." Bundan tashqari, uchinchi argument alice_bob_k1ab turli autentifikatsiya juftlarini ajratish uchun ishlatiladi: ya'ni qiymat qanday maqsadda talqin qilinayotganini tasdiqlash uchun, modellashtirish konventsiyasi sifatida odatda autentifikatsiya qiluvchi ro‘lning nomlari, autentifikatsiya qilinadigan ro‘l va tekshirilayotgan o‘zgaruvchining nomi kichik harflarda birlashtiriladi. U yuqori darajadagi ro‘lda protocol_id tipidagi doimiy sifatida e'lon qilinishi kerak. Shunday qilib, so‘rovning talqini yanada kuchliroq bo‘ladi, chunki A nafaqat B mavjudligini va qiymatga rozi bo‘lishini, balki B uni alice_bob_k1ab protokoli identifikatori uchun ishlatishni ham talab qiladi.

Zaif autentifikatsiyaga mos keladigan so‘rov ham mavjud. Agar wrequest ishlatilsa, takroriy himoya o‘rnatilmaydi. Yuqoridagi misol so‘rov faktini oladigan bo‘lsak, agar bu so‘rov bo‘lsa, B ning mavjudligi talabi bo‘shatiladi. B ning o‘tmishda mavjud bo‘lganligi va o‘sha paytda K1ab' qiymati bo‘yicha kelib chiqib, uni protokol identifikatori alice_bob_k1ab deb talqin qilgan bo‘lishi kifoya.

Shuningdek, mos keladigan guvoh predikatini o‘tishning bir qismi sifatida Bob ro‘liga kiritiladi va K1ab qiymati Alicega yuboriladi.

$$/\backslash\ witness(B, A, alice\_bob\_k1ab, K1ab')$$

Maqsad guvohi (B,A,alice_bob_k1ab,K1ab') "agent B alice_bob_k1ab protokoli identifikatori bilan aniqlangan autentifikatsiya jarayonida K1ab' qiymatidan kelib chiqadi, A

agentining tengdoshi bo‘lish uchun tasdiqlash kerak.

Protokolning maqsad bo‘limida yoziladi.

$$authentication\_on\ bob\_alice\_nb$$

$$authentication\_on\ alice\_bob\_na$$

$$authentication\_on\ alice\_bob\_k1ab$$

Ushbu uchta protokol identifikatorini o‘z ichiga olgan guvoh va so‘rovning maqsadi faktlari hisobga olinishi kerakligini ko‘rsatiladi, Shuningdek, takroriy himoya autentifikatsiya maqsadini belgilash uchun zaif_authentication_on yozish mumkin.

Shunga qaramay, yuqoridagi maqsadlar aslida vaqtinchalik mantiq formulalari uchun makroro‘lardir. Intuitiv ravishda, autentifikatsiya maqsadi so‘rov hodisasidan oldin guvohlik beruvchi voqea sodir bo‘lganligi har doim haqiqat ekanligini tasdiqlaydi. Bu holda hamrohlik qilish, ikki faktning protokol identifikatori va qiymatidan kelib chiqqanligibi va ikkita agent nomi teskari ekanligini anglatadi. Bundan tashqari, kuchli autentifikatsiya uchun hech bir agent bir xil aloqa hamkoridan bir xil qiymatni ikki marta qabul qilmasligi kerak: ya’ni so‘rov hodisasidan bir vaqt oldin bir xil qiymat hech qachon so‘ralmagan.

Bu ta'rif Louning kelib chiqqadi.

• Alice bobni Na qiymatida tasdiqlaydi (bu amal qiladi, chunki faqat bob Na shifrini ochishi mumkin) shuning uchun, takroriy hujumlarni topishga yordam berish uchun OFMC -sessco (sessiyani tuzish) variantini quyidagicha taqdim etadi:

• Bob Aliceni Na qiymati bo‘yicha autentifikatsiya qiladi (bu amal qiladi, chunki faqat Alice shifrini hal qila oladi) va Succ(Na) ni Alicega qaytarib yuboradi.

% avispa ex3.hlpsl --ofmc -sessco Nb va Succ(Nb) ni bobga qaytarib yuboring.

• alice bobni K1ab qiymatida autentifikatsiya qiladi. K1ab-da kuchli autentifikatsiyani suiste'mol qilamiz. Bu erda K1ab yangi hosil bo‘lishi (va qayta o‘ynatilmasligi) kerakligini ifodalash uchun.

Takroriy hujumlarni aniqlash atrof-muhit ro‘li doirasida tegishli tahlil senariysini taqdim etishi kerak. Ko‘pincha, protokolning tuzilishi bunday senariyni taklif qiladi. Bu holatda, Endryu Secure RPC Protokoliga yaxshi ma'lum bo‘lgan hujum mavjud bo‘lib, buzg‘unchi B dan qonuniy to‘rtinchi xabar o‘rniga oldingi protokoldagi to‘rtinchi xabarni takrorlaydi. Bu A eski seans kalitidan foydalanishga majbur qiladi, vaqt o‘tishi bilan buzilgan bo‘lishi mumkin.

Umuman olganda, takroriy hujumlarni yuqoridagi muhit ro‘lida e'lon qilingan dastlabki ikki seansda bo‘lgani kabi bir xil agentlar o‘rtasida bir nechta parallel seanslarni belgilash orqali topish mumkin.

Shuning uchun, takroriy hujumlarni topishga yordam berish uchun OFMC -sessco (sessiyani tuzish) variantini quyidagicha taqdim etadi:

$$\%\ avispa\ ex3.hlpsl\ --ofmc\ -sessco$$

Seans kompilyatsiyasi bilan OFMC a va b o‘rtasidagi ikkinchi parallel seans ham takroriy hujumni topadi. Buning sababi shundaki, u birinchi navbatda butun tizimning ishlashini taqlid qiladi va ikkinchi ishga tushirishda bosqinchiga birinchi bosqichda o‘rganilgan bilimlardan foydalanish imkonini beradi.

AVISPA vositasi kutilgan takroriy hujumni topadi va quyidagi natijani beradi:

$\%\ avispa\ ex31.hlpsl$

% OFMC

% Version of 2005/06/07

SUMMARY

UNSAFE

DETAILS

ATTACK_FOUND

PROTOCOL

./ex31.if

GOAL

replay_protection_on_k1ab

BACKEND

OFMC

COMMENTS

STATISTICS

parseTime: 0.00s

searchTime: 6.59s

visitedNodes: 809 nodes

depth: 8 plies

ATTACK TRACE

i -> (a,3): start

(a,3) -> i: a.{Na(1)}_kab

i -> (a,6): start

(a,6) -> i: a.{Na(2)}_kab

i -> (b,3): a.{Na(2)}_kab

(b,3) -> i: {succ(Na(2)).Nb(3)}_kab

i -> (b,6): a.{Na(1)}_kab

(b,6) -> i: {succ(Na(1)).Nb(4)}_kab

i -> (a,3): {succ(Na(1)).Nb(4)}_kab

(a,3) -> i: {succ(Nb(4))}_kab

i -> (a,6): {succ(Na(2)).Nb(3)}_kab

(a,6) -> i: {succ(Nb(3))}_kab

i -> (b,3): {succ(Nb(3))}_kab

(b,3) -> i: {K1ab(7).N1b(7)}_kab

i -> (a,3): {K1ab(7).N1b(7)}_kab

i -> (a,6): {K1ab(7).N1b(7)}_kab

% Reached State:

% state_dummy(i,replay_protection_on_k1ab,i,0,17)

% request(a,b,alice_bob_k1ab,K1ab(7),6)

% request(a,b,alice_bob_k1ab,K1ab(7),3)

% request(a,b,alice_bob_na,Na(2),6)

% request(a,b,alice_bob_na,Na(1),3)

% witness(b,a,alice_bob_k1ab,K1ab(7),i)

% request(b,a,bob_alice_nb,Nb(3),3)

% secret(K1ab(7),k1ab,set_77)

% secret(N1b(7),n1b,set_78)

% contains(a,set_77)

% contains(b,set_77)

% contains(a,set_78)

% contains(b,set_78)

% witness(a,b,bob_alice_nb,Nb(3),i)

% witness(a,b,bob_alice_nb,Nb(4),i)

% witness(b,a,alice_bob_na,Na(1),i)

% witness(b,a,alice_bob_na,Na(2),i)

% state_bob(b,i,kib,succ,1,dummy_nonce,dummy_nonce,dummy_nonce,
dummy_sk,set_90,set_91,13)

% state_alice(a,i,kai,succ,0,dummy_nonce,dummy_nonce,dummy_sk,
dummy_nonce,9)

% state_alice(a,b,kab,succ,6,Na(2),Nb(3),K1ab(7),N1b(7),6)

% state_bob(b,a,kab,succ,3,Nb(4),Na(1),dummy_nonce,dummy_sk,set_84,set_85,6)

% state_alice(a,b,kab,succ,6,Na(1),Nb(4),K1ab(7),N1b(7),3)

% state_bob(b,a,kab,succ,5,Nb(3),Na(2),N1b(7),K1ab(7),set_77,set_78,3)

“Authentication_on alice_bob_k1ab” maqsadi buzilganligini ko‘rishimiz mumkin.

Topshiriq

Bitta zamonaviy protokolni HLPSL tilida ifodalang va SPAN+AVISPA vositasida yuklang.

Nazorat savollari

1. Endryu Secure RPC protokolining vazifasi va ishlash prinsipini tushuntirib bering
2. Endryu Secure RPC protokolini HLPSL tilida ifodalash bosqichlarini tushuntirib bering
3. O‘zingiz SPAN+AVISPA vositasida yuklagan protoklingizni va uning vazifasini tushuntirib bering.

V-BO‘LIM

KEYSLAR BANKI

V. KEYSLAR BANKI

1-keys mavzusi: “Tashkilot Wi-Fi qurulmasi xavfsizligini ta'minlash”

Vaziyat tavsifi: Tashkilot o‘z xodimlari va mijozlari uchun Wi-Fi xizmatini taqdim etadi. Hozirda tashkilot Wi-Fi xavfsizligi uchun faqat oddiy parolni qo‘llab kelmoqda. Biroq, yaqinda parolni buzish orqali tarmoqqa ruxsatsiz kirish holatlari qayd etildi. Shu sababli, korxona rahbariyati xavfsizlikni oshirish uchun qo‘shimcha chora-tadbirlarni joriy etmoqchi.

Tashkilot foydalanishi mumkin bo‘lgan variantlar:

1. WEP (Wired Equivalent Privacy) protokoli.
2. WPA (Wi-Fi Protected Access).
3. WPA2 yoki WPA3 (Wi-Fi Protected Accessning rivojlangan versiyalari).
4. MAC-manzillarni filtrlash.
5. VPN xizmatlaridan foydalanish.

Vazifa:

Siz IT xavfsizlik bo‘yicha mutaxassis sifatida quyidagi savollarga javob berishingiz kerak:

1. Wi-Fi xavfsizligini ta’minlash uchun qaysi protokoldan foydalanishni tavsiya etasiz?
2. Har bir protokolning afzalliklari va kamchiliklarini qisqacha tahlil qiling.
3. Wi-Fi tarmog‘ini buzish ehtimolini kamaytirish uchun yana qanday xavfsizlik choralarini qo‘shishni taklif qilasiz?
4. Qaysi protokollar va choralar kichik bizneslar uchun eng arzon va samarali bo‘lishi mumkinligini asoslab bering.

Qo‘shimcha shartlar:
Korxona maxsus IT bo‘yicha xodimga ega emas va murakkab xavfsizlik tizimlarini o‘rnatish uchun cheklangan budjetga ega.
Mijozlarning Wi-Fi xizmatiga qulay kirishini saqlab qolish muhim hisoblanadi.
Xodimlarning tarmoqdagi ma'lumotlari maxfiyligi himoya qilinishi kerak.

1) Keysdagi muammoni keltirib chiqargan asosiy sabablarni va ularning oqibatlarini belgilang.

№	Sabab	Oqibat
1		
2		

2) Maqsad, kutiladigan natijalar, vaqt oraliqlari, nazorat indikatorlari kabi jixatlarini aniqlab, tashkilot miqyosida WI-FI texnologiyasidan foydalanishdagi xavfsizlik chorlarini ishlab chiqing.

2-keys mavzusi: “Kriptografik kalit almashish protokollari: xavfsizlik va samaradorlik muammosi”

Vaziyat tavsifi: Bir bank o‘zining ichki tizimlari uchun xavfsiz aloqani ta'minlash maqsadida yangi xavfsizlik protokolini joriy qilmoqchi. Hozirda ular ma'lumotlarni shifrlash va autentifikatsiya qilishda muhim bo‘lgan **kriptografik kalitlarni almashish** jarayonini qanday amalga oshirishni hal qilishi kerak.

Bankning tarmog‘i quyidagi talablarga javob berishi kerak:

1. **Maxfiylik:** Kalitlar almashinuvi paytida ma'lumotlar boshqa tomonlar tomonidan ko‘rilmasligi kerak.
2. **Himoyalanish:** Xakerlar yoki vositachi hujumlaridan (Man-in-the-Middle attack) himoya qilish kerak.
3. **Tezkorlik va samaradorlik:** Almashish jarayoni sekin bo‘lmasligi lozim, chunki tizim real vaqtda ishlashi kerak.

Bank rahbariyati ikkita asosiy usuldan birini tanlashni rejalashtirmoqda:

1. **Diffie-Hellman kalit almashish protokoli.**
2. **Elliptik egri chiziq kriptografiyasi (ECC) asosidagi kalit almashish.**

Vazifa:

Siz IT xavfsizlik bo‘yicha maslahat beruvchi sifatida quyidagi savollarga javob bering:

- Diffie-Hellman protokoli va ECC o‘rtasidagi asosiy farqlarni aniqlang.
- Har bir protokolning afzalliklari va kamchiliklarini baholang.
- Bank uchun xavfsizlik va samaradorlikni hisobga olgan holda optimal tanlovni tavsiya eting.

VI-BO‘LIM

GLOSSARIY

VI. GLOSSARIY

Termin	O‘zbek tilidagi sharhi	Ingliz tilidagi sharhi
Avtomatik tekshirish	ma’lumotlar segmentining to‘g‘riligini har qanday nodasturiy tekshirish	any dumb validation data segment.
Axborot	muayyan obyekt xususidagi bilimlarimizning noaniqlik darajasini pasaytirishga imkon beruvchi har qanday ma’lumot.	data that reduces the uncertainty degree of knowledge about certain object.
Avtomatik nazorat	apparat vositalari yordamida avtomatik bajariluvchi nazorat.	control performed automatically by hardware.
Axborot himoyasining avtonom (injener) vositasi	axborotni ishlashning texnik vositalari komplektiga kirmaydigan maxsus himoya inshoati, qurilmasi yoki moslamasi hamda himoyalash maqsadida foydalanuvchi umummaqsad qurilma.	special protective structure, device or device not included in the scope of the technical means of information processing as well as general-purpose device that is used for protection purposes.
Avtorizatsiya	tizimda foydalanuvchiga, uning ijobiy autentifikatsiyasiga asosan, ma’lum foydalanish huquqlarini taqdim etish.	View user specific access rights on the basis of a positive result in its authentication system.
Mualliflik huquqi	fan, adabiyot va san’at asarlarini yaratish, foydalanish va huquqiy himoyalashda vujudga keladigan munosabatlarni tartibga soluvchi huquqiy normalar majmui.	the body of law (Civil Law Section), which regulate the relations arising in connection with the creation and use of scientific, literary and artistic works (copyright.
Foydalanish ma’muri	ma’lumotlar bazasidan foydalanishni tashkil etishga javobgar, ma’lumotlar banki ma’muriyati tarkibidagi lavozimli shaxslardan biri.	one of the officials in the administration part of the data bank, the organization responsible for user access to databases.
Himoya ma’muri	avtomatlashtirilgan tizimni axborotdan ruxsatsiz foydalanishdan himoyalashga javobgar foydalanish subyekti.	access entity responsible for the protection of the automated system from unauthorized access to information.
Axborotni himoyalash sohasidagi akkreditatsiya	himoyalangan buyumlar, texnik vositalar va axborotni himoyalash usullari sertifikatsiyasi sohasida qandaydir faoliyat olib borish huquqiyligining rasman tan olinishi.	official recognition of the powers to carry out any activities in the field of certification security products, tools and methods of information security.
Ma’lumotlar	odam ishtiroki bilan yoki avtomatik tarzda uzatishga, izohlashga yoki ishlashga yaroqli, formallashgan ko‘rinishda ifodalangan axborot.	information presented in a formalized manner suitable for communication, interpretation or processing involving human or automated means.

Faol taxdid	tizim holatiga atayin ruxsatsiz o‘zgartirish kiritish tahdidi.	the threat of a deliberate unauthorized system state changes.
Faol bekitish	axborotni himoyalashning texnik usuli. Ushbu usulga binoan axborot eltuvchisini aniqlashni va uni olishni qiyinlashtirish maqsadida signallarning, maydonlarning energetik xarakteristikalari yoki moddalar konsentratsiyasi oshiriladi.	a way of technical protection of information that consists in increasing the energy characteristics of the signals, fields or concentrations of substances difficult to detect carriers and receipt of information.
Himoya faolligi	himoya moxiyatiga mos xolda, texnik razvedkaga obyekt xususida yolg‘on tasavvurni maqsadli zo‘rlab qabul qildiruvchi hamda texnik razvedka imkoniyatlarini bostiruvchi himoyalash prinsipi.	protection principle, expressed in the imposition of targeted technical intelligence false representation of the object in accordance with the concept of protection and suppression capabilities of technical intelligence.
Himoya muolajalarining tahlili	qabul qilingan himoya strategiyasiga va operatsion muolajalarga mosligini ta’minlash uchun tizimli yozuvlarni va aktivlarni, ularning tizimli boshqarish funksiyalariga adekvatligini tekshirish maqsadida mustakil ko‘zdan kechirish, taxlillash, himoyadagi nuqsonlarni aniqlash va boshkarishdagi, strategiyadagi va muolajalardagi aniklangan har qanday o‘zgarishlar buyicha tavsiyalar berish.	Independent review and analysis of system records and activities in order to verify the adequacy of the system control functions to ensure compliance with established security policy and operating procedures, detecting gaps in protection and provide recommendations for any indicated changes in management, strategy and procedures.
Tarmoq taxlillagichlari (sniffer)	tarmoq trafigini “tinglash”ni va tarmoq trafigidan avtomatik tarzda foydalanuvchilar ismini, parollarni, kredit kartalar nomerini, shu kabi boshqa axborotni ajratib olishni amalga oshiruvchi dasturlar.	programs, asking for “listening” network traffic and automatically selects the network traffic of user names, passwords, credit card numbers, other similar information.
Hujum	bosqinchining operatsion muhitini boshqarishiga imkon beruvchi axborot tizimi xavfsizligining buzilishi.	breach of security of information system, which allows the invader to manage operating environment.
Faol xujum	kriptotizimga yoki kriptografik protokolga hujum bo‘lib, unga binoan dushman va/yoki buzg‘unchi konuniy foydalanuvchi xarakatiga ta’sir etishi, masalan, qonuniy foydalanuvchi xabarini almashtirishi yoki yo‘q qilishi va xabarni yaratib uning nomidan uzatishi va h.	attack on a cryptosystem or cryptographic protocol in which the offender or the enemy and can affect the legitimate user actions, for example, replace or remove legitimate posts, create and send messages on his behalf,

	mumkin.	etc.
Xizmat qilishdan voz kechishga undaydigan hujum	tizim buzilishiga sabab bo‘luvchi hujum, yani shunday sharoitlar tuhdiradiki, qonuniy foydalanuvchi tizim taqdim etgan resurslardan foydalana olmaydi yoki foydalanish anchagina qiyinlashadi.	attack to cause failure of the system, that is to create the conditions under which legitimate users can not get access to the resources provided by the system, or that access will be significantly hampered.
Ximoya attestatsiyasi	himoyani baholash malakali va kerakli qoidalarga mos amalga oshirilganligi xususidagi vakolatli bilimdon shaxsning tasdig‘i.	confirmation by the competent person that the assessment was made protection and qualified in accordance with the necessary rules.
Autentifikatsiya	odatda tizim resurslaridan foydalanishga ruxsat etish xususida qaror qabul uchun foydalanuvchining (haqiqiyligini), qurilmaning yoki tizimning boshqa tashkil etuvchisining identifikatsiyasini tekshirish; saqlanuvchi va uzatuvchi ma’lumotlarning ruxsatsiz modifikatsiyalanganligini aniqlash uchun tekshirish.	checking user authentication (authentication), device, or other component in the system, usually to make a decision about granting access to system resources; checking the integrity of stored or transmitted data to detect unauthorized modification.
Foydalanuvchining autentifikatsiyasi	foydalanuvchi taqdim etgan identifikator yordamida uning haqiqiyligini tasdiqlash. Yana – foydalanuvchini u taqdim etgan identifikatorga mosligini tekshirish.	user authentication using against them authenticator. Also, to check compliance against them user ID.
Axborot tarmog‘i xavfsizligi	axborot tarmog‘ini ruxsatsiz foydalanishdan, me’yoriy harakatiga tasodifan aralashishdan yoki komponentlarini buzishga urinishdan saqlash choralari.	measures designed to protect network information from unauthorized access, accidental or intentional interference with normal activities or attempts to destroy its components.
Axborot resurslari xavfsizligi	axborot resurslarining ob’yektiv va sub’yektiv, ichki va tashqi, tasodifan va atayin taxdidlar ta’siridan himoyalanish holati.	State security information resources from the operation of objective and subjective, internal and external, accidental and intentional threats.
Tarmoq xavfsizligi	axborot tarmog‘ini ruxsatsiz foydalanishdan, me’yoriy ishlashiga tasodifan yoki atayin aralashishdan yoki tarmoq komponentlarini buzishga urinishdan extiyot qiluvchi choralar. Asbob-uskunalarni, dasturiy	measures that protect the network information from unauthorized access, accidental or intentional interference with normal activities or attempts to destroy

	ta’minotni, ma’lumotlarni himoyalashni o‘z ichiga oladi.	its components. Includes the protection of hardware, software, data.
Foydalanishni blokirovka qilish	xotiraning cheklangan qismidan, masalan, nuqson aniqlangan disk qismidan, foydalanishning man qilinishi. Dasturiy yoki apparat vositalar yordamida bajariladi.	denying access to a restricted area of memory, e.g., disk track, so in this region detected defects. Perform a software or hardware.
Brandmauer	apparat-dasturiy vositalar yordamida tarmoqdan foydalanishni markazlashtirish va uni nazoratlash yo‘li bilan tarmoqni boshqa tizimlardan va tarmoqlardan keladigan xavfsizlikka taxdidlardan himoyalash usuli. Yana - bir necha komponentlardan (masalan, brandmauer dasturiy ta’minoti ishlaydigan marshrutizator yoki shlyuzdan) tashkil topgan ximoya to‘sig‘i hisoblanadi.	a method of protecting the network from security threats from other systems and networks by centralizing network access and control of hardware and software. Also, is a protective barrier, consisting of several components (such as a router or gateway that is running firewall software).
Paketlarni filtrlovchi brandmauer	kiruvchi va chiquvchi paketlarni ma’lum xillarini brakka chiqarish maqsadida konfiguratsiyalangan dasturiy ta’minot ishlaydigan marshrutizator yoki kompyuter.	a router or computer on which the software is running, configured so as to reject certain types of incoming and outgoing packets.
Yekspert sathidagi brandmauer	olinadigan paketlarni ISO modelining uchta sathida - tarmoq, seans va tatbiqiy sathlarda tekshiradi. Ushbu vazifani bajarishda paketlarni filtrlashning maxsus algoritmlari ishlatiladi. Ular yordamida har bir paket avtorizatsiyalangan paketlarning ma’lum shablonlari bilan taqqoslanadi.	checks the contents of the packets received on the three levels of the model OSI - network, session and application. To perform this task, use special packet filtering algorithms by which each packet is compared with the known pattern of authorized packets.
Ma’lumotlarni tiklash	eltuvchining asl nusxasida ma’lumotlar yaxlitligi buzilganida unga ma’lumotlarning himoya nusxasi bo‘lgan eltuvchidan nusxalash jarayoni.	the process of copying data from one media containing protecting your data on original carrier in case of violation of the integrity of the data on it.
Xavfsizlik domeni	xavfsizlikning bitta ma’muri tomonidan xavfsizlikning bir xil usuli qo‘llaniladigan xavfsizlik subyektlari va obyektlarining cheklangan guruxi.	limited group of objects and subjects of security, to which the one method of security from the same security administrator.

VII-BO‘LIM

ADABIYOTLAR RO‘YXATI

VII. ADABIYOTLAR RO'YXATI

I. O'zbekiston Respublikasi Prezidentining asarlari:

1. Mirziyoyev SH.M. Buyuk kelajagimizni mard va olijanob xalqimiz bilan birga quramiz. – T.: "O'zbekiston", 2017. – 488 b.
2. Mirziyoyev SH.M. Milliy taraqqiyot yo'limizni qat'iyat bilan davom ettirib, yangi bosqichga ko'taramiz. 1-jild. – T.: "O'zbekiston", 2017. – 592 b.
3. Mirziyoyev SH.M. Xalqimizning roziligi bizning faoliyatimizga berilgan eng oliy bahodir. 2-jild. –T.: "O'zbekiston", 2018. – 507 b.
4. Mirziyoyev SH.M. Niyati ulug' xalqning ishi ham ulug', hayoti yorug' va kelajagi farovon bo'ladi. 3-jild.– T.: "O'zbekiston", 2019. – 400 b.
5. Mirziyoyev SH.M. Milliy tiklanishdan – milliy yuksalish sari. 4-jild.– T.: "O'zbekiston", 2020. – 400 b.

II. Normativ-huquqiy hujjatlar:

6. O'zbekiston Respublikasining Konstitusiyasi.–T.:O'zbekiston, 2018.
7. O'zbekiston Respublikasining 2020 yil 23 sentabrda qabul qilingan "Ta'lim to'g'risida"gi O'RQ-637-sonli Qonuni.
8. O'zbekiston Respublikasi Prezidentining 2017 yil 7 fevral "O'zbekiston Respublikasini yanada rivojlantirish bo'yicha Harakatlar strategiyasi to'g'risida"gi 4947-sonli Farmoni.
9. O'zbekiston Respublikasi Prezidentining 2018 yil 21 sentabr "2019-2021 yillarda O'zbekiston Respublikasini innovatsion rivojlantirish strategiyasini tasdiqlash to'g'risida"gi PF-5544-sonli Farmoni.
10. O'zbekiston Respublikasi Prezidentining 2019 yil 27 may "O'zbekiston Respublikasida korrupsiyaga qarshi kurashish tizimini yanada takomillashtirish chora-tadbirlari to'g'risida"gi PF-5729-son Farmoni.
11. O'zbekiston Respublikasi Prezidentining 2019 yil 27 avgust "Oliy ta'lim muassasalari rahbar va pedagog kadrlarining uzluksiz malakasini oshirish tizimini joriy etish to'g'risida"gi PF-5789-sonli Farmoni.
12. O'zbekiston Respublikasi Prezidentining 2019 yil 8 oktabr "O'zbekiston Respublikasi oliy ta'lim tizimini 2030 yilgacha rivojlantirish konsepsiyasini tasdiqlash to'g'risida"gi PF-5847-sonli Farmoni.
13. O'zbekiston Respublikasi Prezidenti Shavkat Mirziyoyevning 2020 yil 25 yanvardagi Oliy Majlisga Murojaatnomasi.
14. O'zbekiston Respublikasi Vazirlar Mahkamasining 2001 yil 16 avgustdagi "Oliy ta'limning davlat ta'lim standartlarini tasdiqlash to'g'risida"gi 343-sonli Qarori.
15. O'zbekiston Respublikasi Vazirlar Mahkamasining 2015 yil 10 yanvardagi "Oliy ta'limning Davlat ta'lim standartlarini tasdiqlash to'g'risida"gi 2001 yil 16 avgustdagi "343-sonli qororiga o'zgartirish va qo'shimchalar kiritish haqida"gi 3-sonli qarori.

III. Maxsus adabiyotlar:

16. AVISPA Tool for the automated validation of internet security pro-tocols and applications. In CAV'2005, volume 3576 of LNCS, pages 281–285. Springer, 2005.
17. Y. Chevalier, L. Compagna, J. Cuellar, P. Hankes Drielsma, J. Man-tovani, S. M¨odersheim, and L. Vigneron. A high level protocol spec-ification language for industrial security-sensitive protocols. In Pro-ceedings of Workshop on Specification and Automated

Processing of Security Requirements (SAPS), Linz, Austria, 2004.

18. F. Jacquemard, M. Rusinowitch, and L. Vigneron. Compiling and Verifying Security Protocols. In Proceedings of 7th Conference on Logic for Programming and Automated Reasoning, volume 1955 of LNAI. Springer-Verlag, 2000.

19. A. Armando and L. Compagna. SATMC: a SAT-based model checker for security protocols. In Proceedings of the 9th European Conference on Logics in Artificial Intelligence (JELIA'04), volume 3229 of LNAI, pages 730–733, Lisbon, Portugal, 2004. Springer-Verlag.

20. The AVISPA Tool v1.1. Available at http://www.avispa-project.org/, 2006.

21. AVISPA. Deliverable 2.1: The High-Level Protocol Specification Language. Available at http://www.avispa-project.org/publications.html, 2003.

22. AVISPA. Deliverable 2.3: The Intermediate Format. Available at http://www. avispa-project.org/publications.html, 2003.

23. AVISPA. The AVISPA User Manual. Available at http://www.avispa-project.org/ publications.html, 2005.

24. D. Basin, S. M¨odersheim, and L. Vigan`o. OFMC: A Symbolic Model-Checker for Security Protocols. International Journal of Information Security, 2004.

25. Y. Boichut, P.-C. H´eam, O. Kouchnarenko, and F. Oehl. Improvements on the Genet and Klay Technique to Automatically Verify Security Protocols. In Proceedings of Automated Verification of Infinite States Systems (AVIS'04), ENTCS, 2004. To appear.

26. M. Burrows, M. Abadi, and R. Needham. A Logic of Authentication. ACM Transactions on Computer Systems, 8(1):18–36, 1990.

27. Y. Chevalier, L. Compagna, J. Cuellar, P. Hankes Drielsma, J. Mantovani, S. M¨odersheim, and L. Vigneron. A High Level Protocol Specification Language for Industrial Security- Sensitive Protocols. In Proc. SAPS'04. Austrian Computer Society, 2004.

28. J. Clark and J. Jacob. A Survey of Authentication Protocol Literature: Version 1.0, 17. Nov. 1997. URL:www.cs.york.ac.uk/~jac/papers/drareview.ps.gz.

29. D. Dolev and A. Yao. On the Security of Public-Key Protocols. IEEE Transactions on Information Theory, 2(29), 1983.

30. L. Lamport. The temporal logic of actions. ACM Transactions on Programming Languages and Systems, 16(3):872–923, May 1994.

31. L. Lamport. Specifying Systems. Addison-Wesley, 2002.

32. G. Lowe. A hierarchy of authentication specifications. In Proceedings of the 10th IEEE Computer Security Foundations Workshop (CSFW'97), pages 31–43. IEEE Computer Society

33. M. Turuani. The CL-Atse Protocol Analyser. In F. Pfenning, editor, Proceedings

of 17th International Conference on Rewriting Techniques and Applications, RTA, Lecture Notes in Computer Science, Seattle (WA), Aug. 2006. Springer.

IV. Internet saytlar:

34. http://edu.uz – O'zbekiston Respublikasi Oliy ta'lim, fan va innovasiyalar vazirligi.
35. http://lex.uz – O'zbekiston Respublikasi Qonun hujjatlari ma'lumotlari milliy bazasi.
36. http://bimm.uz – Oliy ta'lim tizimi pedagog va rahbar kadrlarini qayta tayyorlash va ularning malakasini oshirishni tashkil etish Bosh ilmiy-metodik markazi.
37. http://ziyonet.uz – Ta'lim portali ZiyonET.
38. http://natlib.uz – Alisher Navoiy nomidagi O'zbekiston Milliy kutubxonasi.
39. http://www.avispa-project.org
40. http://people.irisa.fr/Thomas.Genet/.