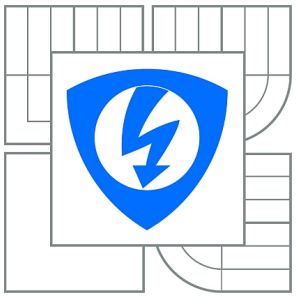
VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

DEPARTMENT OF TELECOMMUNICATIONS

KRYPTOGRAFICKÝ PROTOKOL VÝMĚNY KLÍČŮ DIFFIE-HELLMAN

DIFFIE-HELLMAN KEY EXCHANGE PROTOCOL

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

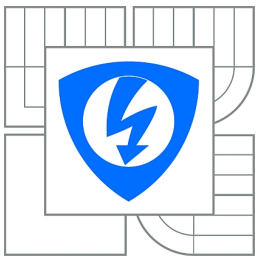
AUTOR PRÁCE — VLASTIMIL ČLUPEK

AUTHOR

VEDOUCÍ PRÁCE — Ing. JIŘÍ SOBOTKA

SUPERVISOR

BRNO 2010



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky a komunikačních technologií

Ústav telekomunikací

Bakalářská práce

bakalářský studijní obor
Teleinformatika

Student:	Vlastimil Člupek	***ID:***	106397
Ročník:	3	***Akademický rok:***	2009/2010

NÁZEV TÉMATU:

Kryptografický protokol výměny klíčů Diffie-Hellman

POKYNY PRO VYPRACOVÁNÍ:

Analyzujte základní verzi protokolu Diffie-Hellman a novější verzi Eliptic Curve Diffie-Hellman. Porovnejte tyto verze, pokuste se o aplikaci útoku Man in the middle a navrhněte ochranu před tímto útokem. Dále se zabývejte analýzou vzájemné kompatibility mezi jednotlivými verzemi protokolu a pokuste se navrhnout řešení kompatibility.

DOPORUČENÁ LITERATURA:

[1] SCHNEIER, Bruce. Applied cryptography. 2nd edition. [s.l.] : John Wiley & Sons, 1996. 784 s. ISBN 0-471-11709-9 .
[2] MENEZES, Alfred J. Elliptic Curve Public Key Cryptosystems. [s.l.] : [s.n.], 1993. 128 s. ISBN

Termín zadání:	29.1.2010	***Termín odevzdání:***	2.6.2010

Vedoucí práce: Ing. Jiří Sobotka

prof. Ing. Kamil Vrba, CSc.
Předseda oborové rady

UPOZORNĚNÍ:

ABSTRAKT

V této bakalářské práci je vysvětlena podstata kryptografie, šifrovací metody a hlavně kryptografický protokol výměny klíčů Diffie-Hellman (DH). Je zde popsán postup výměny klíče přes veřejný kanál. Problematika diskrétního logaritmu. Útok „Man in the middle“ na tento protokol a ochrana před tímto útokem. Následně je zde popsána novější verze tohoto protokolu, která pracuje s eliptickými křivkami. Její název je Elliptic Curve Diffie-Hellman (ECDH). U tohoto protokolu je zde dále popsán postup výpočtu tajného bodu na eliptické křivce. Problematika eliptického diskrétního logaritmu. Útok „Man in the middle“ na protokol ECDH a ochrana před tímto útokem. Dále se tato práce zabývá analýzou vzájemné kompatibility mezi protokolem DH a ECDH a možným jejím řešením.

KLÍČOVÁ SLOVA

Kryptografie, protokol Diffie-Hellman (DH), eliptická křivka, protokol Elliptic Curve Diffie-Hellman (ECDH), útok „Man in the middle“, kompatibilita mezi protokoly DH a ECDH.

ABSTRACT

In this bachelor's thesis there is explained principle of cryptography, encryption methods and primarily cryptographic protocol key exchange Diffie-Hellman. There is described key exchange process via public channel. Problem of discrete logarithm. Attack "Man in the middle" on this protocol and protection against this attack. Subsequently there is described a newer version of this protocol, which work's with elliptic curves. Its name is Elliptic Curve Diffie-Hellman (ECDH). In this protocol there is hereinafter described a procedure of calculate a secret point on elliptic curve. Problem of Elliptic discrete logarithm. Attack "Man in the middle" on protocol ECDH and protection against this attack. Furthermore this thesis deals with analyzing mutual compatibility between protocol DH and ECDH and its possible solutions.

KEYWORDS

Cryptography, protocol Diffie-Hellman (DH), elliptic curve, protocol Elliptic Curve Diffie-Hellman (ECDH), attack "Man in the middle", compatibility between the protocols DH and ECDH.

ČLUPEK, V. *Kryptografický protokol výměny klíčů Diffie-Hellman*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2010. 55 s. Vedoucí bakalářské práce Ing. Jiří Sobotka.

PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma „Kryptografický protokol výměny klíčů Diffie-Hellman“ jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

V Brně dne ....

(podpis autora)

OBSAH

Seznam obrázků **8**

Úvod **9**

1 Kryptografie **10**

1.1 Kryptografie a její historie ... 10

1.2 Šifrovací metody ... 11

1.2.1 „Klasické“ šifrování ... 12

1.2.2 Symetrické šifrování ... 12

1.2.3 Asymetrické šifrování ... 13

2 Kryptografické protokoly **15**

2.1 Požadavky na kryptografické protokoly ... 15

2.1.1 Utajení ... 15

2.1.2 Autentizace ... 15

2.1.3 Integrita ... 15

2.1.4 Ověřená výměna klíčů ... 16

2.1.5 Nepopiratelnost ... 16

2.1.6 Korektnost ... 16

2.1.7 Anonymita ... 16

3 Protokol Diffie-Hellman (DH) **17**

3.1 Diskrétní logaritmus ... 18

3.2 Výpočet tajného klíče protokolu Diffie-Hellman ... 19

3.3 Diffie-Hellman pro tři nebo více účastníků ... 20

3.4 Útok „Man in the middle“ ... 20

3.4.1 Realizace útoku „Man in the middle“ ... 21

3.5 Digitální podpis ... 22

3.5.1 Sestavení digitálního podpisu ... 23

3.5.2 Ověření digitálního podpisu ... 23

3.6 Ochrana před útokem „Man in the middle“ ... 23

4 Protokol Elliptic Curve Diffie-hellman (ECDH) **26**

4.1 Eliptické křivky ... 26

4.1.1 Eliptická křivka nad tělesem *Fp* ... 27

4.1.2 Eliptická křivka nad tělesem F_2^m ... 28
4.1.3 Bezpečnost eliptických křivek ... 30
4.2 Obecné stanovení tajného bodu pomocí ECDH ... 31
4.2.1 Ustanovení tajného bodu pomocí ECDH nad tělesem *Fp* ... 31
4.3 Útok „Man in the middle“ v ECDH ... 35
4.4 Ochrana před útokem „Man in the middle“ v ECDH ... 35

5 Kompatibilita protokolu DH s prokolem ECDH 37

5.1 Porovnání protokolu DH s protokolem ECDH ... 37
5.2 Propojení modulární aritmetiky s aritmetikou na eliptické křivce ... 39
5.3 Systém pracující s protokolem DH a protokolem ECDH ... 41
5.3.1 Návrh systému propojujícího protokol DH a ECDH ... 41
5.3.2 Obecný postup propojení protokolu ECDH s protokolem DH ... 42
5.3.3 Propojení protokolu DH s protokolem ECDH ... 43
5.3.4 Útok „Man in the middle“ v propojení protokolů ECDH a DH ... 46
5.3.5 Ochrana před „Man in the middle“ v propojení protokolů ECDH a DH ... 48
5.3.6 Praktické využití systému propojujícího protokol DH a ECDH ... 50

Závěr 52

Literatura 53

Seznam zkratek 55

SEZNAM OBRÁZKŮ

Obr. 1.1: Rozdělení šifrovacích metod a příklady nejznámějších algoritmů 11

Obr. 1.2: Princip symetrického šifrování 12

Obr. 1.3: Princip asymetrického šifrování 13

Obr. 3.1: Diskrétní mocnění a diskrétní logaritmování 17

Obr. 3.2: Algoritmus výpočtu tajného klíče 18

Obr. 3.3: Algoritmus útoku „Man in the middle“ 21

Obr. 4.1: Ukázka eliptické křivky a geometrického sčítání bodů $P + Q = R$ 25

Obr. 4.2: Algoritmus stanovení tajného bodu K na eliptické křivce nad tělesem Fp 32

Obr. 4.3: Obecný algoritmus útoku „Man in the middle“ na protokol ECDH 33

Obr. 4.4: Algoritmus ochrany před útokem „Man in the middle“ kdy jsou přenášené body M a N podepsány důvěryhodnou stranou 36

Obr. 5.1: Propojení protokolu ECDH s protokolem DH s využitím universálního systému při dohodě na tajném bodě a čísle a při posílání dat přes veřejný kanál 42

Obr. 5.2: Algoritmus výpočtu tajného bodu a čísla odesílatele, universálního systému a příjemce 45

Obr. 5.3: Algoritmus přenosu šifrovaných dat od odesílatele přes universální systém k příjemci 45

Obr. 5.4: První část algoritmu útoku „Man in the middle“, při které dochází k dohodě na tajných bodech na eliptické křivce a dále pak na dohodě na tajném čísle 47

Obr. 5.5: Druhá část algoritmu útoku „Man in the middle“, při které dochází k přenosu dat od odesílatele přes universální systém k příjemci 47

Obr. 5.6: Algoritmus ochrany před útokem „Man in the middle“ v propojení protokolů DH a ECDH a domluva na tajném klíči k 49

ÚVOD

V této bakalářské práci je v první kapitole vysvětlena podstata kryptografie a dále je zde stručně popsána její historie. Následně jsou zde rozebrány jednotlivé šifrovací metody, jejich principy, výhody a nevýhody.

Druhá kapitola se zabývá kryptografickými protokoly, jejich definicí a požadavky, které jsou na ně kladeny.

V následující kapitole je rozebrán konkrétní kryptografický protokol a to protokol Diffie-Hellman (DH). Je zde uveden jeho algoritmus dohody výpočtu tajného čísla mezi komunikujícími stranami přes veřejný nezabezpečený kanál. V této kapitole je dále zhodnocena bezpečnost algoritmu protokolu Diffie-Hellman. Následně je zde uvedena slabina tohoto protokolu a to útok „Man in the middle“. Tento útok je zde názorně předveden. Dále tato kapitola obsahuje návrhy ochran před tímto útokem.

Čtvrtá kapitola se zabývá novější verzí protokolu Diffie-Hellman, která pracuje s eliptickými křivkami nad konečnými tělesy Fp a $F_2{}^m$. Tato novější verze nese název Elliptic Curve Diffie-Hellman (ECDH). Je zde znázorněn algoritmus dohody výpočtu na tajném bodě mezi dvěma komunikujícími stranami. Dále jsou v této kapitole vysvětleny postupy při sčítání bodů na eliptické křivce a při zdvojení bodů na eliptické křivce, které se využívá v případě když je zapotřebí nějaký bod na eliptické křivce vynásobit určitým číslem. Toto násobení bodů na eliptické se využívá při výpočtu tajného bodu v protokolu ECDH, konkrétní postup je zde uveden. Také je zde rozebrána bezpečnost této novější verze protokolu DH. Protokol ECDH je analogický k protokolu DH, je tedy také náchylný na útok „Man in the middle“. Ochrana před tímto útokem je zde také obsažena, je analogická k ochraně k původní verzi.

Pátá kapitola se zabývá analýzou vzájemné kompatibility mezi protokolem Diffie-Hellman a protokolem Elliptic Curve Diffie-Hellman. Dále je zde navržen universální systém pracující s protokolem DH i ECDH umožňující vzájemnou kompatibilitu mezi oběma protokoly.

V závěru je provedeno zhodnocení a porovnání obou protokolů, je zde uvedeno, který z nich je rychlejší a bezpečnější a tudíž má perspektivnější výhledy do budoucna. Dále je v závěru zhodnoceno navržené řešení vzájemné kompatibility mezi protokoly DH a ECDH, výhody a nevýhody uvedeného řešení a jeho možné nasazení v praxi.

1 KRYPTOGRAFIE

1.1 Kryptografie a její historie

Kryptografie spadá do vědní disciplíny, která je označována jako **kryptologie**, ta se zabývá utajením a odhalením zpráv a dělí se na tři části a to na již zmiňovanou kryptografii, kryptoanalýzu a steganografii.

Kryptografie slouží k nalezení a popisu metod utajení zpráv pomocí šifer. Šifra představuje algoritmus, pomocí kterého je možné převést srozumitelné zprávy na zprávy nesrozumitelné. Tyto nesrozumitelné (šifrované) zprávy jsou nesrozumitelné jen pro osoby neznalé šifrovacího klíče. Rozluštění těchto zpráv bez znalosti klíče se zabývá nauka zvaná **kryptoanalýza,** ta hledá v dané šifře slabé místo, pomocí kterého dokáže zašifrovaný text přečíst.

První zmínky o kryptografii je možné najít už v 5. století před našim letopočtem. Lidé už v této době se potřebovali domlouvat na dálku bez toho aby byl jejich vzkaz čitelný pro jinou další osobu. V této době se začala vyvíjet a používat věda zvaná **steganografie,** tato věda se zabývá „skrýváním zpráv“. V praxi to funguje tak, že komunikace probíhá tam kde na první pohled není vůbec zřejmé, že nějaká komunikace probíhá. Příkladem může být text ukrytý v obrázku. Ve starém Řecku se např. text vyrýval do dřevěné destičky a následně zaléval voskem, aby nebyl čitelný. V dnešní době informačního věku může být zpráva ukryta třeba v souborech se zvukem nebo videem. Tato technika má jednu základní nevýhodu, po jejím odhalení je ihned odkryt celý text zprávy. To byl důvod, proč se začala rozvíjet kryptografie.

První kryptografická metoda nazývaná též „klasická“ využívá dvou způsobů šifrování: substituce (písmeno v textu je nahrazeno jiným písmenem) a transpozice (písmena v textu jsou přeuspořádána). Substituce byla základem posuvné Caesarovy šifry. Kde bylo postupně každé písmeno v textu nahrazeno jiným písmenem, které se nacházelo o několik pozic dál v abecedě. Klíčem pro rozluštění této zprávy bylo číslo v rozmezí 1 až 25 (počet písmen v abecedě je 26), které udávalo o kolik písmen je nutné se vrátit zpět aby byl text čitelný. Tento způsob šifrování byl poměrně slabý a brzy byl prolomený. V následujících letech byla vynalezena monoalfabetická šifra, která byla bezpečnější obměnou Caesarovy šifry, spočívala v tom že každé písmeno v textu bylo nahrazeno jiným písmenem v nezvyklém pořadí. Například písmeno A bylo nahrazeno písmenem M, dále třeba písmeno B bylo nahrazeno písmenem Z. Pro rozluštění této zprávy bylo zapotřebí mít tabulku, která udávala jakým písmenem bylo nahrazeno původní písmeno. Pro rozluštění této šifry použili kryptoanalitici tzv. frekvenční analýzu, která pracuje s četností výskytu písmene ve větách. Např. když je známo, že v češtině se nejvíce vyskytuje písmeno A a je zašifrován český text, ze kterého je vidět, že v tomto zašifrovaném textu se nejvíce vyskytuje písmeno M, tak je jednoduché určit, že pod zašifrovaným písmenem M se skrývá písmeno A. Takto se dále rozluštily nejfrekventovanější písmena abecedy a zbytek písmen se doplnil podle kontextu. Tento fakt, že se kryptoanalitikům dařilo šifry dešifrovat, přinutil kryptografiky vynalézat stále dokonalejší metody zaručující bezpečnost zpráv.

V 16. století objevil francouzský diplomat Blaise de Vigenère v té době tzv. neprolomitelnou šifru. Tato šifra měla kořeny v šifře monoalfabitické, avšak jednomu písmenu v textu mohlo náležet více písmen v šifrovaném textu. To znamená, že vůči frekvenční analýze byla tato šifra zcela odolná, protože každé písmeno v textu bylo zašifrováno podle jiného posunu. Například u slova KOLO by byl jiný posun pro písmeno K, jiný zase pro první písmeno O, jiný pro písmeno L a jiný pro druhý písmeno O. Po čase však tato šifra byla také prolomena.

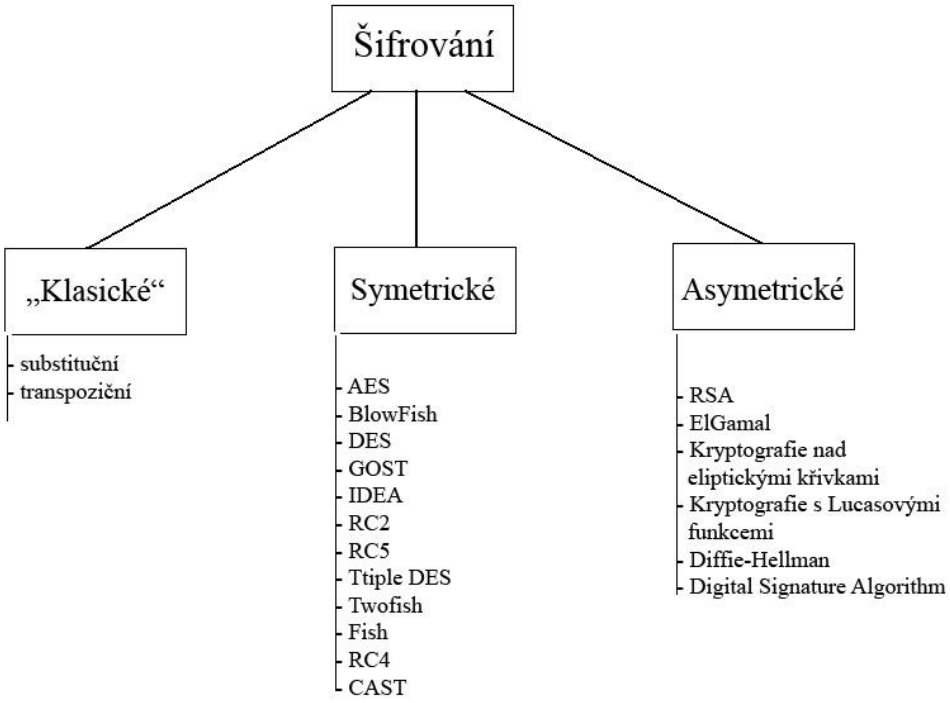
V roce 1917 vymyslel Gilbert Vernam tzv. Vernamovu šifru, která se také označuje pod pojmem One-time pad. Jejím základem byl posun písmen o předem náhodně vygenerovaná čísla, které tvořili klíč. Klíč je tedy stejně dlouhý jako přenášená zpráva, je zcela náhodný a lze ho tedy použít jenom jednou. Tyto skutečnosti zaručovaly této šifře za daných okolností neprolomitelnost. Největším a základním problémem této šifry je distribuce klíče. To byl důvod proč se využívala jen zřídka.

V roce 1918 si nechal německý vynálezce Arthur Scherbius patentovat šifrovací přístroj Enigma. Tento přístroj se skládá z pěti částí [5]: klávesnice pro psaní textu, propojovací desky, tří rotujících vyměnitelných šifrovacích jednotek–scramblerů, reflektoru a desky, na které se zobrazoval šifrovaný text. Bez klíče, pomocí kterého se dalo text rozluštit, by nepřítel musel vyzkoušet celkově 10^{16} možností na dešifrování zprávy. V počátcích byl mechanizmus tohoto přístroj považován za dokonalý, který nebylo možné dešifrovat. Později se však našel způsob, jak dešifrovat takto zašifrovaný text.

Rozvoj počítačů pomohl k tomu, že se šifry staly dostatečně bezpečné. Největším problémem byla distribuce klíče. Tento problém vyřešili pánové Whitfield Diffie a Martin Hellman, kdy v roce 1976 publikovali jejich protokol Diffie-Hellman. O rok později trojice amerických vědců: Leonard Adleman, Ronald Rivest a Adi Shamir vynalezli první asymetrické šifry RSA, tyto šifry jsou založeny na principu jednosměrné funkce (v přímém směru lze ji spočítat velmi snadno, zatímco v opačném směru se výpočty dají provést velmi obtížně), to umožnilo vznik soukromého a veřejného klíče. Bezpečnost této šifry roste exponenciálně s velikostí jejího klíče. Přesto se v roce 1999 podařilo prolomit šifru RSA s délkou klíče 512 bitů. Phil Zimmermann, autor šifrovacího softwaru PGP vycházel z této metody. Jeho šifrovací software používá současně symetrické a asymetrické šifrování. Tímto mechanizmem je umožňováno široké veřejnosti větší soukromí.

1.2 Šifrovací metody

Šifrovací metody je možné rozdělit na „historické“–v současné době nepoužívané–tzv. „klasické“ šifrování a na „současné“–v současné době používané, ty je dále možné dělit na symetrické a na asymetrické. Obrázek 1.1 znázorňuje rozdělení šifrovacích metod a příklady nejznámějších algoritmů (protokolů).



Obr. 1.1: Rozdělení šifrovacích metod a příklady nejznámějších algoritmů (protokolů).

1.2.1 „Klasické“ šifrování

Někdy bývá toto šifrování označováno jako „ruční“, vyplívá to z principů které jsou zde použity. Do tohoto typu šifrování se řadí dvě nejzákladnější metody, které stály u zrodu kryptografie.

Šifry využívající principu **transpozice**–v textu se mění pořadí písmen podle daných pravidel. Patří sem například Vigenèrova šifra.

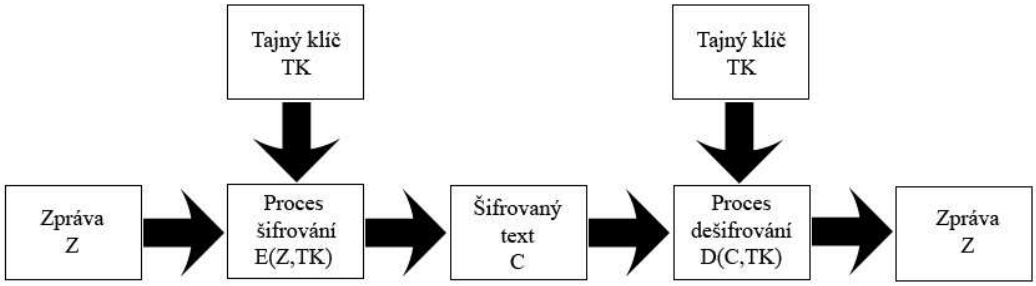
Šifry využívající principu **substituce**–každé písmeno v textu je nahrazeno jiným písmenem. Tento způsob náhrady je předem domluvený. Patří sem např. Caesarova šifra, nebo monoalfabetická substituční šifra.

Tyto způsoby šifrování se už v dnešní době stále rozvíjející se výpočetní techniky nepoužívají, nebo jen zřídka a to tam kde není požadována až tak velká bezpečnost. K rozluštění těchto šifer, totiž útočník ani nepotřebuje počítač.

1.2.2 Symetrické šifrování

U tohoto typu šifrování se používá k zašifrování a následnému dešifrování textu jeden šifrovací tajný klíč (TK). Na obr. 1.2 je vidět princip symetrického šifrování. Platí tedy, že každý kdo chce pracovat se zašifrovanými daty musí tento tajný klíč znát. Pro tento

typ šifrování je požadován bezpečný přenos klíče osobám, který budou pověřeny se zašifrovanými daty pracovat. V okamžiku kdy by byl prozrazen tajný klíč třeba jen jednou osobou, budou všechny informace zašifrované podle tohoto klíče prozrazeny.



Obr. 1.2: Princip symetrického šifrování.

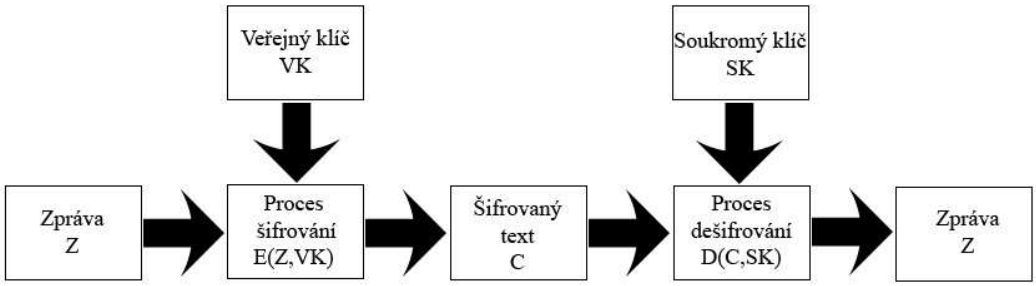
Mezi nejznámější symetrické šifrovací algoritmy patří AES, DES, Triple DES, IDEA, GOST, RC2, RC4, RC5, Fish, TwoFish, BlowFish, CAST. DES byl v 70. letech vyvinut firmou IBM a v roce 1977 se stal v USA normou pro šifrování. Šifrovací algoritmus DES používá klíč s délkou 56 bitů. V dnešní době s výrazně zdokonalenou výpočetní technikou už z hlediska bezpečnosti nedostačuje. Dokonce byl prolomen metodou tzv. „Hrubého útoku“ kdy útočník vyzkouší všechny možné kombinace klíče. Proto vědci vyvinuli bezpečnější formu DESu s názvem TripleDES. Princip této metody je velice jednoduchý, data jsou prostě dvakrát nebo třikrát přešifrována. Tento mechanizmus tedy pracuje buď to s klíčem dlouhým 112 bitů (2x56 bitů) nebo s klíčem dlouhým 168 bitů (3x56 bitů). Tímto dvojnásobným popřípadě trojnásobným přešifrováním dat je způsobeno, že TripleDES je oproti běžnému DES pomalejší, ale za to bezpečnější. Základní výhodou pro rozšíření TripleDES byla zpětná kompatibilita se staršími zařízeními využívající verzi DES. Další zmíněný algoritmus IDEA je z hlediska bezpečnosti a rychlosti perspektivnější než DES, pracuje s klíčem dlouhým 128 bitů a je výpočetně rychlejší než DES. Dále uvedený algoritmus BlowFish má zajímavou vlastnost tu, že může mít proměnnou délku šifrovacího klíče od 32 do 448 bitů, většinou se však používá s klíčem dlouhým 128 bitů. Tento algoritmus je rychlý, bezpečný a není zatížen patentovými právy. Algoritmus CAST používá také proměnnou délku klíče, touto vlastností se podobá algoritmu BlowFish. Algoritmus AES může mít délku klíče rovnu 128, 192 nebo 256 bitů. Tato šifra se vyznačuje vysokou rychlostí šifrování.

1.2.3 Asymetrické šifrování

U tohoto typu šifrování se používají dva různé, ale „vzájemně kompatibilní“ klíče. Jeden pro zašifrování–tzv. veřejný klíč (VK) a jeden pro dešifrování–tzv. soukromý klíč (SK). Obrázek 1.3 znázorňuje princip asymetrického šifrování. Veřejný klíč, jak už je patrné z jeho názvu, je určen k volnému šíření a distribuuje se všem osobám se kterými pomocí šifrovaných zpráv dochází ke komunikaci. Soukromý klíč, jak už je patrné

z jeho názvu, by měl zůstat v soukromí jeho vlastníka, neměl by se o něm dovědět nikdo jiný. Princip asymetrického šifrování je takový, že text který byl zašifrován veřejným klíčem lze dešifrovat pouze soukromým klíčem a naopak. Jenom jeden klíč (soukromý nebo veřejný) nelze použít k zašifrování a opětnému dešifrování, vždy musí být použity oba klíče veřejný i soukromý. To je způsobeno vlastností asymetrického šifrování, kde jsou použité matematické funkce, jejichž reverzní výpočet je buď to neproveditelný nebo výpočetně příliš náročný, tyto funkce bývají také označovány jako jednosměrné.

Asymetrické šifrovací algoritmy jsou v porovnání se symetrickými obecně výrazně pomalejší. Asymetrické kryptosystémy (např. RSA), kryptografické protokoly i metody digitálních podpisů používají komplikované operace s dlouhými čísly, které by standardnímu PC trvaly příliš dlouho. Proto se často šifruje klasickými symetrickými systémy (např. TripleDES, IDEA) a asymetrickými systémy se šifrují pouze relativně krátké použité symetrické klíče[13].



Obr. 1.3: Princip asymetrického šifrování.

V praxi je nevíce zažitý algoritmus RSA, nově se však začínají používat algoritmy na bázi eliptických křivek (Elliptic Curve Cryptography). Autoři RSA jak už bylo uvedeno v kapitole 1.1 jsou Rivest, Shamir, Adleman. Bezpečnost tohoto algoritmu roste exponenciálně s velikostí jeho klíče. Např. u elektronického podpisu se standardně používá klíč s minimální délkou 1024 bitů. Algoritmus RSA vznikl v roce 1977 a byl chráněn patentem do roku 2000. Jestliže dokáže útočníkův systém řešit úlohy faktorizace velkých čísel, pak je schopný tento algoritmus prolomit. Ve srovnání s DES je RSA výrazně pomalejší. Pokud je realizován softwarově uvádí se, že je pomalejší přibližně 100 krát, pokud je realizován hardwarově uvádí se, že je pomalejší 1000 krát až 10 000 krát. Je tedy výhodnější zašifrovat data symetrickým algoritmem s náhodným klíčem a ten následně zašifrovat pomocí asymetrického algoritmu. Tento zašifrovaný klíč je poté přibalen k symetricky zašifrovaným datům. Poté je celý tento „balíček“ odeslán příjemci, ten pak následně asymetricky dešifruje symetrický klíč, pomocí kterého dešifruje symetricky zašifrována data.

2 KRYPTOGRAFICKÉ PROTOKOLY

Kryptografický protokol definuje podmínky, podle kterých se určuje způsob komunikace a druh sdílených informací. Dalo by se říct, že každý kryptografický protokol vychází z určitého kryptografického algoritmu. Důvodem proč jsou kryptografické protokoly tak masivně používané je fakt, že se při zabezpečování přenosu dat zabývají řešením velmi rozsáhlých okruhů problémů. Kryptografický protokol v praxi zajišťuje autentizaci (ověřování pravosti) účastníků protokolu, vytvoření dohody o kryptografickém klíči, výměnu tohoto klíče/klíčů apod. Správně navržený a v praxi správně fungující kryptografický protokol neumožní zúčastněným uživatelům získat jiné informace nebo provést jiné akce než přesně ty, které jsou v protokolu předem určeny.

2.1 Požadavky na kryptografické protokoly

Kryptografický protokol [15] je vždy vytvořen za určitým účelem a na základě jistých požadavků, které má splňovat s ohledem na bezpečnost při komunikaci. V následujících podkapitolách jsou uvedeny nejzásadnějších vlastnosti, které jsou po protokolech požadovány, včetně jejich otevřenější interpretace. Přesný význam je totiž stanovován při samotném modelování protokolu.

2.1.1 Utajení

Pro bezpečný přenos je požadováno zakrytí určitých informací před nepovolanými osobami. Každá aplikace požaduje jinou úroveň zakrytí, také si vybírá jaký druh informací má být zakryt. Někdy je požadováno aby útočník nezískal žádné informace které probíhají mezi komunikujícími stranami. V tomto případě by mělo dojít k zakrytí všech druhů informací. Ve většině případů je však dostačující, když je znemožněna analýza síťového provozu. Útočník sice dokáže zjistit, že mezi sebou dvě strany komunikují, má přístup k zašifrovanému textu, ale nedokáže ho dešifrovat.

2.1.2 Autentizace

Úkolem autentizace je ověření pravosti, tím je myšleno, že se ověřuje jestli odeslaná zpráva, která se tváří, že pochází z určitého zdroje byla opravdu tímto zdrojem vytvořena a odeslána.

2.1.3 Integrita

Aby byla zajištěna integrita, musí být přijímaná data celistvá a neporušená, musí být zachován formát zprávy. Pokud jedna strana pošle druhé straně nějaká data a je zachována integrita, musí mít přijatá zpráva stejný obsah jako před odesláním. Integrita může být narušena buď to útočníkem, který během přenosu pozmění data, nebo chybou při přenosu dat, např. výpadkem spojení.

2.1.4 Ověřená výměna klíčů

V některých případech prostá bezpečná výměna klíčů nepostačuje, komunikující strany jsou si jisty, že klíče byly vyměněny bezpečně, ale nejsou si jisty s kým si klíče a následná data vyměnily. Tento problém řeší ověřená výměna klíčů (Authenticated Key Exchange).

2.1.5 Nepopiratelnost

Nepopiratelnost vnáší do kryptografických protokolů takové mechanizmy, které zajistí důkazy možného nekorektního chování jedné nebo obou zúčastněných stran. Tímto nekorektním chováním je možné chápat např. situaci kdy jeden účastník pošle zprávu druhému účastníkovi a poté tvrdí, že tuto zprávu neodeslal.

2.1.6 Korektnost

V některých případech je korektní chování zúčastněných stran velmi důležité. Příkladem může být bankovnictví. Je důležité, aby obě strany provedly všechny kroky protokolu, vynecháním nějakého kroku by mohla zúčastněná strana získat výhodu nad druhou stranou. Tato snaha zajistit korektní chování vedla k nárůstu množství přenášených dat. Jedním ze způsobů jak zajistit korektní chování je požádání o dohled třetí nezaujatou důvěryhodnou stranu, na kterou se v případě neshody komunikující strany obrátí.

2.1.7 Anonymita

V určitých aplikacích je vyžadováno skrytí identity komunikujícího subjektu. Například u elektronického hlasování je někdy požadována anonymita, zároveň je však požadováno, aby přes danou anonymitu nebylo možné volit vícekrát. Požadavek anonymity tedy zahrnuje[15]: zakrytí identity subjektu tak, aby nebyly patrné vazby na další subjekty, případně na akce či operace spojené s tímto subjektem. Anonymita je narušena, pokud je možné spojit jistý subjekt s akcí, kterou vykonal, či s jiným subjektem.

3 PROTOKOL DIFFIE-HELLMAN (DH)

Protokol Diffie-Hellman jako úplně první začal využívat algoritmus, který umožňuje vznik a výměnu tajného klíče po veřejných nezabezpečených sítí. Jedná se o první algoritmus asymetrické kryptografie. Autoři tohoto protokolu jsou Whitfield Diffie a Martin Hellman., publikovali ho v roce 1976 a dali mu název podle svých příjmení. Bezpečnost této metody je dána složitým výpočtem diskrétního logaritmu. Protokol Diffie-Hellman je možné použít pro distribuci klíče dvou komunikujících stran, tento algoritmus může být použit ke generaci bezpečného klíče, ale nemůže být použít k šifrování a dešifrování zpráv. K tomuto účelu je možné použít například některý z uvedených algoritmů (protokolů) symetrického šifrování v kapitole 1.2.2. Pro výpočet tajného klíče je zapotřebí znát několik základních operací v matematice. Protokol Diffie-Hellman při výpočtu klíče pracuje s prvočísly a modulární aritmetikou.

Prvočíslo je definováno jako přirozené číslo, které je dělitelné beze zbytku pouze dvěma čísly a to číslem jedna a sebou samým, z toho vyplívá, že číslo jedna není prvočíslo. **Modulární aritmetika** je [8] aritmetikou na množině celých čísel Z v níž se čísla opakují po dosažení určité hodnoty n, která je nazývána modulem. Na rozdíl od běžných celočíselných operací se zde po každé operaci provede ještě celočíselné dělení modulem n a výsledkem operace je zbytek po tomto dělení. Následující tři příklady jsou vypočteny pomocí zmíněného principu.

9 mod 7 = 2

(3 + 5) mod 5 = 3

3^2 mod 2 = 1

V praxi se v protokolu Diffie-Hellman při výpočtu tajného klíče používají vysoké prvočíselné moduly.

Matematický postup pro stanovení klíče je jednoduchý. Nejdřív se obě strany dohodnou na velkém prvočísle n, jedna strana je navrhne a druhá strana vyjádří souhlas. Toto prvočíslo je nazýváno modulem. Poté je nutné aby se dohodly na dalším čísle (g), vygenerovaném generátorem, které tvoří základ modulu n, pro číslo g platí $1 \leq g \leq n-1$. Není požadováno, aby g bylo co největší číslo, může být klidně i jednomístné. Prvočíslo n musí být z hlediska bezpečnosti vybráno tak, aby bylo dostatečně veliké a aby hodnota (n-1)/2 byla také prvočíslo. Tyto dvě celá čísla n a g jsou veřejná, nemusí být tedy tajná. Mohou být přenášena přes nechráněný kanál a dokonce mohou být i veřejná mezi skupinou uživatelů. Postup výpočtu tajného klíče bude následující:

1) Odesílatel a příjemce se dohodnou na prvočísle n a na přirozeném čísle g.
2) Odesílatel si zvolí tajné náhodně velké přirozené číslo x.
3) Příjemce si zvolí tajné náhodně velké přirozené číslo y.
4) Odesílatel si vypočítá hodnotu $A = g^x \bmod n$ a odešle ji nezabezpečeně příjemci.
5) Příjemce si vypočítá hodnotu $B = g^y \bmod n$ a odešle ji nezabezpečeně odesílateli.
6) Odesílatel si vypočítá tajné číslo $k = B^x \bmod n$.
7) Příjemce si vypočítá tajné číslo $k' = A^y \bmod n$.

Tyto dvě čísla k a k' jsou totožná a jsou rovny g^{xy} mod n. Dále se používají jako tajný klíč K. Nikdo odposlouchávající kanál nedokáže zjistit hodnotu tajného klíče. Dokáže zjistit pouze hodnotu čísel n, g, A, B. Pokud útočník dokáže vypočítat diskrétní logaritmus a obnovit číslo x nebo y, pak dokáže určit tajný společný klíč odesílatele a příjemce. Bezpečnost tajného klíče je dána složitostí výpočtu diskrétního logaritmu.

3.1 Diskrétní logaritmus

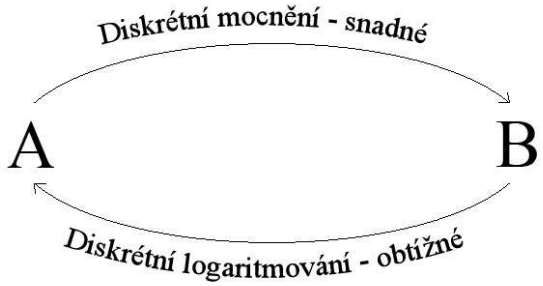
Diskrétní logaritmus je definován pro konečné cyklické grupy. Cyklická grupa je definována jako skupina čísel, splňující určité vlastnosti. Diskrétní logaritmus vychází z normálního logaritmu.

Logaritmus je exponent, na který když je umocněn základ, je dosáhnuto logaritmovaného čísla. Platí $\log_a x = v \Leftrightarrow x = a^v \; ; \; a \in R^+ \setminus \{1\}$, kde a je základ logaritmu, x je logaritmované číslo, v je logaritmus čísla x o základu a.

Definici diskrétní logaritmu je možné ukázat na výpočtu tajného čísla v protokolu Diffie-Hellman (kapitola 3 bod 6). Je dáno $k = B^x \bmod n \Leftrightarrow \log_B k = x \bmod n$, za těchto podmínek je možné nazvat x diskrétním logaritmem čísla k o základu B vzhledem k modulu n.

Pro lepší pochopení je dobré vysvětlit diskrétní logaritmus na příkladu. Na začátku bude vysvětleno diskrétní mocnění. Je zvolena cyklická skupina s modulem $n = 11$. Je požadováno vypočítat příklad $13^5 \bmod 11$, postup bude následující, číslo 13 bude umocněno číslem 5, z tohoto výsledku bude následně vypočítán zbytek po celočíselném dělení modulem $n = 11$. Tímto bude dosáhnuto výsledku příkladu, tedy že $13^5 \bmod 11 = 10$. Tento výpočet byl rychlý a nebyl vůbec náročný.

Diskrétní logaritmus je opačnou operací k diskrétnímu mocnění. Je-li dán příklad $13^x \bmod 11 = 10$, neexistuje žádný efektivní algoritmus, pomocí kterého je možné zjistit hodnotu x. Z ukázky diskrétního mocnění, je vidět, že x je rovno číslu 5. Nejjednodušší postup, jak zjistit hodnotu x je postupně od 1 zvyšovat hodnotu mocniny (číslo x) dokud nebude dosáhnuto správného výsledku, v tomto případě čísla 10. Tento postup je však časově velmi náročný a nevede k řešení v polynomiálním čase. Polynomiální čas má tu vlastnost, že se změnou počítaných parametrů velice rychle narůstá. Existují i další metody používané k řešení diskrétního logaritmu, většinou se inspirují v řešením problému faktorizace, které časovou náročnost snižují, žádná ale neřeší problém diskrétního logaritmu v polynomiálním čase.



Obr. 3.1: Diskrétní mocnění a diskrétní logaritmování.

3.2 Výpočet tajného klíče protokolu Diffie-Hellman

Odesílatel a příjemce se dohodli na prvočísle $n = 11$, dále se dohodli na přirozeném čísle $g = 6$, které bylo vygenerováno generátorem. Odesílatel, tyto dvě čísla zvolil, příjemce s nimi souhlasil. Postup by mohl být i opačný, příjemce by zvolil tyto dvě čísla a odesílatel by vyjádřil souhlas.

Odesílatel si zvolil tajné číslo $x = 12$. Příjemce si zvolil tajné číslo $y = 7$.

Odesílatel si vypočítal hodnotu $A = g^x \bmod n = 6^{12} \bmod 11 = 3$ a odeslal ji příjemci.

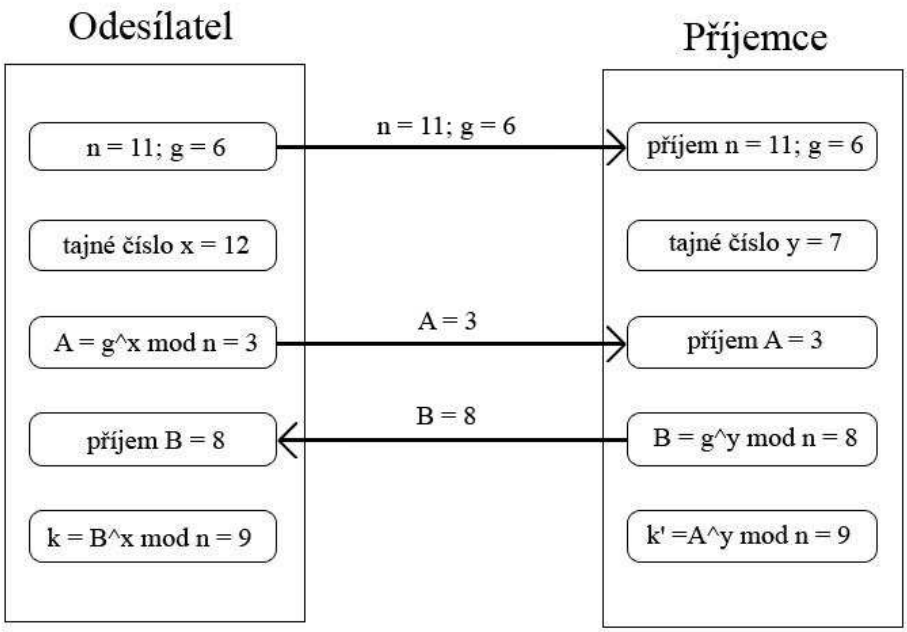
Příjemce si vypočítal hodnotu $B = g^y \bmod n = 6^7 \bmod 11 = 8$ a odeslal ji odesílateli.

Odesílatel si vypočítal tajný klíč $k = B^x \bmod n = 8^{12} \bmod 11 = 9$.

Příjemce si vypočítá tajný klíč $k' = A^y \bmod n = 3^7 \bmod 11 = 9$.

Při této ukázce byly zvoleny pro jednoduchost výpočtu malá čísla, v praxi by se kvůli bezpečnosti použily mnohem větší čísla. Je vidět, že odesílatel a příjemce došli ke stejnému tajnému klíči $k = k' = 9$. Výpočet tajného klíče tedy proběhl bezchybně. Nyní je možné tento klíč použít k šifrování pomocí nějakého symetrického algoritmu.

Ze znalosti x, y, g a n je možné vypočítat tajný klíč $k = g^{xy} \bmod n = 6^{84} \bmod 11 = 9$. Obrázek 3.2 znázorňuje algoritmus výpočtu tajného sdíleného klíče (čísla) mezi odesílatelem a příjemcem.



Obr. 3.2: Algoritmus výpočtu tajného klíče.

3.3 Diffie-Hellman pro tři nebo více účastníků

Protokol Diffie-Hellman lze jednoduše rozšířit pro práci se třemi nebo více účastníky. Pro přehlednost budou jednotliví účastníci pojmenováni Alice, Bob a Cyril. Následující postup popisuje jak si Alice, Bob a Cyril společně vytvoří tajný klíč.

1) Alice, Bob a Cyril se dohodnou na prvočísle n a na přirozeném čísle g.
2) Alice si zvolí tajné náhodně velké přirozené číslo x.
3) Bob si zvolí tajné náhodně velké přirozené číslo y.
4) Cyril si zvolí tajné náhodně velké přirozené číslo z.
5) Alice si vypočítá hodnotu $A = g^x \bmod n$ a odešle ji nezabezpečeně Bobovi.
6) Bob si vypočítá hodnotu $B = g^y \bmod n$ a odešle ji nezabezpečeně Cyrilovi.
7) Cyril si vypočítá hodnotu $C = g^z \bmod n$ a odešle ji nezabezpečeně Alici.
8) Alice si vypočítá hodnotu $A' = C^x \bmod n$ a odešle ji nezabezpečeně Bobovi.
9) Bob si vypočítá hodnotu $B' = A^y \bmod n$ a odešle ji nezabezpečeně Cyrilovi.
10) Cyril si vypočítá hodnotu $C' = B^z \bmod n$ a odešle ji nezabezpečeně Alici.
11) Alice si vypočítá tajný klíč $k = C'^x \bmod n$.
12) Bob si vypočítá klíč $k = A'^y \bmod n$.
13) Cyril si vypočítá klíč $k = B'^z \bmod n$.

Tajný klíč k je roven $g^{xyz} \bmod n$ a nikdo odposlouchávající komunikaci nedokáže určit jeho hodnotu. Protokol je možné snadno rozšířit pro čtyři nebo více účastníků. Přidáním více účastníků roste počet kroků výpočtu.

3.4 Útok „Man in the middle“

Z uvedených informací z předchozích kapitol vyplývá, že útočník odposlouchávající komunikaci dvou uživatelů nedokáže zjistit hodnotu tajného klíče. Slabinou protokolu Diffie-Hellman však je, že neověřuje autentizaci přenášených zpráv v bodech 4 a 5 (kapitola 3), kdy odesílatel posílá příjemci hodnotu $A = g^x \bmod n$ a následně příjemce posílá odesílateli hodnotu $B = g^y \bmod n$. Tento fakt umožňuje útočníkovi provést útok **„Man in the middle“** v překladu „Muž uprostřed“.

Tento útok spočívá v tom, že útočník vstoupí mezi komunikující strany a začne se vydávat za odesílatele respektive příjemce, dohodne si vlastní klíče s oběma stranami a s pomocí těchto klíčů dešifruje všechny vyměněné zprávy mezi odesílatelem a příjemcem bez toho, aby komunikující strany zjistily, že si klíč vyměnily s někým jiným. Postup poté co útočník vstoupí mezi komunikující strany by mohl být následující:

1) Útočník vydávající se za příjemce přijme od odesílatele čísla n a g, uloží si je a vydávajíc se za odesílatele odešle tyto dvě čísla příjemci.
2) Odesílatel si zvolí tajné číslo x, příjemce si zvolí tajné číslo y.
3) Útočník si zvolí své tajné číslo z a vypočítá si hodnotu $C = g^z \bmod n$.
4) Odesílatel si vypočítá $A = g^x \bmod n$ a příjemce si vypočítá $B = g^y \bmod n$.

5) Útočník vydávající se za příjemce přijme od odesílatele hodnotu A a pošle mu svou hodnotu C.
6) Útočník vydávající se za odesílatele odešle příjemci svou hodnotu C a následně příjme od příjemce jeho hodnotu B.
7) Odesílatel si vypočítá tajný klíč $k = C^x \bmod n$.
8) Útočník si vypočítá první tajný klíč $k_1 = A^z \bmod n$.
9) Příjemce si vypočítá tajný klíč $k' = C^y \bmod n$.
10) Útočník si vypočítá druhý tajný klíč $k_2 = B^z \bmod n$.

Nyní si odesílatel i příjemce myslí, že mají totožný tajný klíč, na kterém se dohodli, a že nikdo jiný nezná hodnotu tohoto klíče, ale opak je pravdou.

Pokud výpočty proběhly korektně klíče k a k_1 budou totožné a budou se rovnat $K_1 = g^{xz} \bmod n$. A klíče k' a k_2 budou také totožné, budou se rovnat $K_2 = g^{zy} \bmod n$. Ve výsledku tedy vzniknou dva různé tajné klíče K_1 a K_2, tyto oba dva klíče bude vlastnit útočník. Odesílatel bude vlastnit jenom tajný klíč K_1 a příjemce bude vlastnit jenom tajný klíč K_2. Pomocí prvního tajného klíče K_1 odesílatel symetricky zašifruje zprávu a odešle ji útočníkovi vydávajícího se za příjemce. Ten tuto symetricky zašifrovanou zprávu pomocí tajného klíče K_1 dešifruje (dešifrovanou zprávu může i pozměnit) a následně ji symetricky zašifruje pomocí tajného klíče K_2 a vydávajíc se za odesílatele ji odešle příjemci. Příjemce následně tuto symetricky zašifrovanou zprávu dešifruje pomocí tajného klíče K_2.

3.4.1 Realizace útoku „Man in the middle“

Odesílatel poslal příjemci prvočíslo $\boldsymbol{n = 7}$ a číslo $\boldsymbol{g = 4}$, útočník vydávajíc se za příjemce přijal od odesílatele tyto dvě čísla a uložil si je. Následně útočník vydávajíc se za odesílatele odeslal příjemci tyto dvě čísla.

Odesílatel si zvolil tajné číslo $\boldsymbol{x = 8}$. Příjemce si zvolil tajné číslo $\boldsymbol{y = 6}$. Útočník si zvolil tajné číslo $\boldsymbol{z = 5}$.

Útočník si vypočítal hodnotu $\boldsymbol{C = g^z \bmod n = 4^5 \bmod 7 = 2}$.

Odesílatel si vypočítal hodnotu $\boldsymbol{A = g^x \bmod n = 4^8 \bmod 7 = 9}$ a příjemce si vypočítal hodnotu $\boldsymbol{B = g^y \bmod n = 4^6 \bmod 7 = 1}$.

Útočník vydávající se za příjemce přijal od odesílatele hodnotu $\boldsymbol{A = 9}$ a odeslal mu svoji hodnotu $\boldsymbol{C = 2}$.

Útočník vydávající se za odesílatele odeslal příjemci hodnotu $\boldsymbol{C = 2}$ a přijal od něj jeho hodnotu $\boldsymbol{B = 1}$.

Odesílatel si vypočítal tajný klíč $\boldsymbol{k = C^x \bmod n = 2^8 \bmod 7 = 4}$.

Útočník si vypočítal první tajný klíč $\boldsymbol{k_1 = A^z \bmod n = 9^5 \bmod 7 = 4}$.

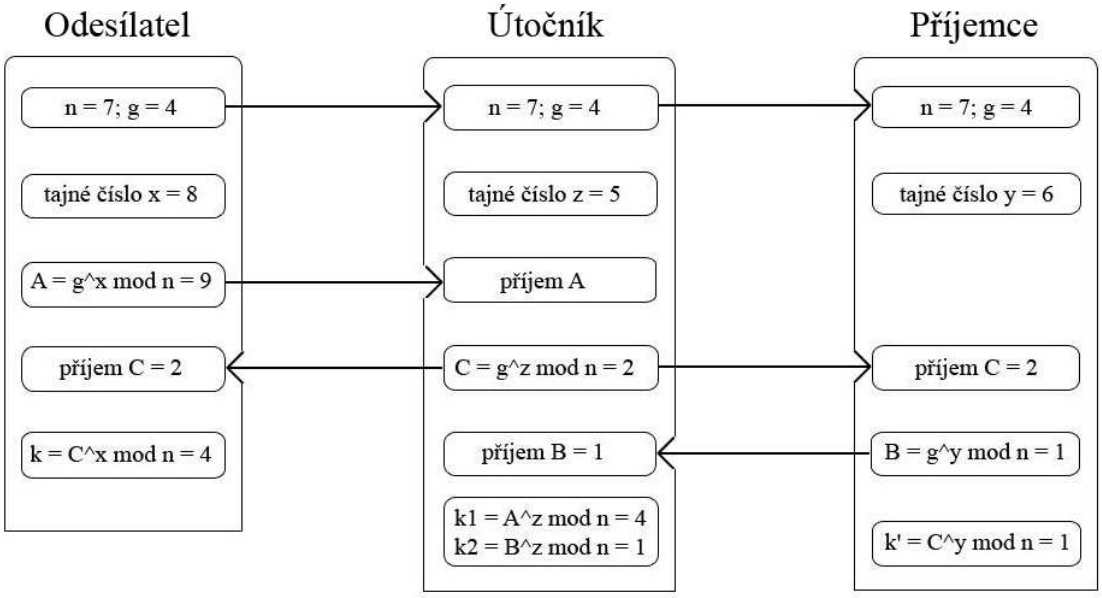
Příjemce si vypočítal tajný klíč $\boldsymbol{k' = C^y \bmod n = 2^6 \bmod 7 = 1}$.

Útočník si vypočítal druhý tajný klíč $\boldsymbol{k_2 = B^z \bmod n = 1^5 \bmod 7 = 1}$.

Při této ukázce byly pro jednoduchost výpočtu zvoleny malá čísla, v praxi by se kvůli bezpečnosti volily mnohem větší čísla.

Je vidět že, $\boldsymbol{k = k_1 = 4}$ a $\boldsymbol{k' = k_2 = 1}$, útok „Man in the middle“ byl proveden úspěšně, útočník získal dva klíče. Pomocí prvního klíče (který zná i odesílatel) dešifruje přijatou zprávu od odesílatele, následně ji zašifruje pomocí druhého klíče (který zná i příjemce) a odešle ji příjemci. Příjemce pak pomocí svého klíče přijatou zprávu dešifruje.

Odesílatel i příjemce se myslí, že se bezpečně dohodli na tajném klíči, a že nikdo nedokáže dešifrovat zprávu, kterou si poslali. Skutečnost je však jiná, útočník dokáže dešifrovat zprávu a dokonce může změnit její obsah.



Obr. 3.3: Algoritmus útoku „Man in the middle“.

3.5 Digitální podpis

Jak už bylo řečeno slabinou protokolu Diffie-Hellman je, že neověřuje autentizaci zúčastněných stran. Tuto slabinu dokáže odstranit digitální podpis, který tedy zajistí autentizaci autorů. Digitální podpis je založen na metodách asymetrického šifrování. Digitální podpis má hned několik zásadních výhod [13] :

- Jeho základní vlastností je nepopiratelnost,
- je prakticky nemožné jej zfalšovat,
- lze jednoduše ověřit jeho autenticitu,
- jeho použitím je zaručena neporušenost zprávy (resp. zjištění jejího porušení),
- v kombinaci se šifrováním je zpráva chráněna před vyzrazením obsahu. Navíc může obsahovat časovou značku a být tak jednoznačně určena v čase.

3.5.1 Sestavení digitálního podpisu

Při vytváření digitálního podpisu určitých čísel (určitého souboru) pro případ protokolu Diffie-Hellman *A* a *B* nejdříve musí podepisující osoba (příjemce) určit hash (otisk) daných čísel (daného souboru). Jakýkoliv soubor je tedy chápán jako sled čísel. Na tyto čísla *A* a *B* (na tento sled čísel) je aplikován hash algoritmus. Výstupem tohoto algoritmu je určité číslo o určité délce, které jednoznačně reprezentuje vstupní čísla *A* a *B* (data). Tento hash asymetricky zašifruje podepisující osoba pomocí jejího soukromého klíče, tím vznikne digitální podpis, k tomuto digitálnímu podpisu se přidáním certifikátu získají digitálně podepsaná data.

3.5.2 Ověření digitálního podpisu

Postup při ověřování digitálního podpisu je následující. Ověřující osoba (odesílatel) si nejprve vypočte svůj hash z původních čísel (z původního souboru) pro případ protokolu Diffie-Hellman z čísel *A* a *B*. Poté dešifruje přijatý podpis pomocí veřejného klíče příjemce (uloženého v certifikátu), tím získá hash příjemce. V dalším kroku porovná svůj hash s hashem příjemce, jestli jsou shodné, může si být jist, že komunikuje opravdu s tou osobou, se kterou chce komunikovat, tedy s vlastníkem soukromého klíče, který tímto soukromým klíčem zašifroval svůj hash a tím vytvořil digitální podpis, a který dále poskytnul certifikační autoritě svůj veřejný klíč pro vytvoření certifikátu. Pokud se tyto dva hashe liší, může to znamenat, že se někdo pokusil padělat podpis, nebo po cestě pozměnil přenášené číslo, pro protokol DH *A* nebo *B* (nebo přenášený soubor), nebo cokoliv jiného, co by ve výsledku vedlo k změnění nebo porušení přenášených dat či podpisu samotného.

3.6 Ochrana před útokem „Man in the middle“

Jeden ze způsobů jak zabránit útočníkovi provést útok „Man in the middle“ je, že si odesílatel a příjemce navzájem nechají podepsat jejich přenášené zprávy další důvěryhodnou stranou, pro případ protokolu Diffie-Hellman uvedený v kapitole 3 jsou to hodnoty *A* a *B*.

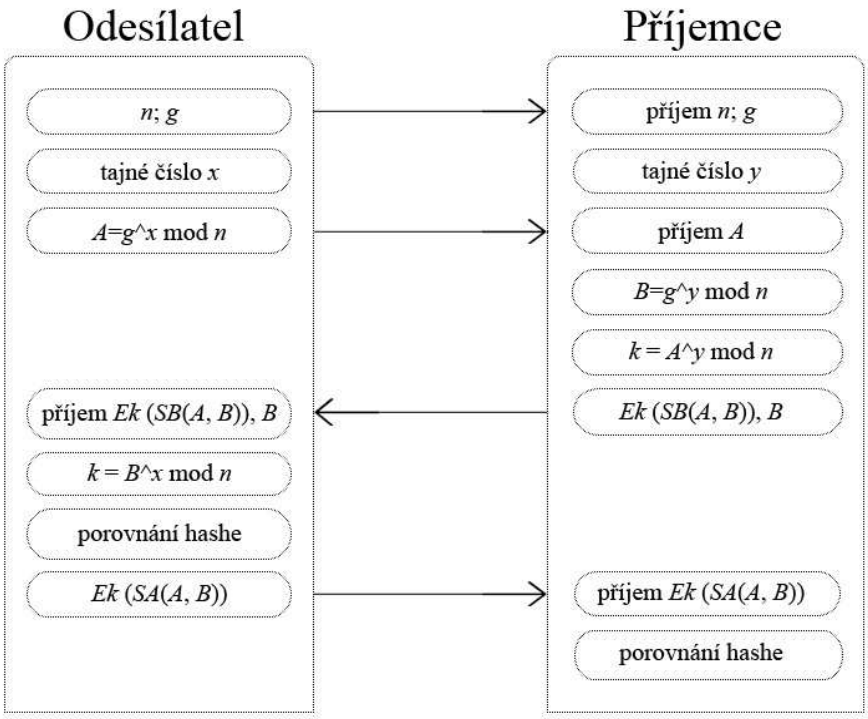
Při tomto postupu se předpokládá, že odesílatel má certifikát s veřejným klíčem příjemce a příjemce má certifikát s veřejným klíčem odesílatele. Oba tyto certifikáty byly podepsány nějakou důvěryhodnou certifikační autoritou mimo právě probíhající domluvu klíče (byly podepsány již dříve). Následující postup popisuje jak odesílatel a příjemce generují tajný klíč *k*.

1) Odesílatel a příjemce se dohodnou na prvočísle *n* a na přirozeném čísle *g*.
2) Odesílatel si zvolí tajné náhodně velké přirozené číslo *x*.
3) Příjemce si zvolí tajné náhodně velké přirozené číslo *y*.
4) Odesílatel si vypočítá hodnotu $A = g^x \bmod n$ a odešle ji nezabezpečeně příjemci.
5) Příjemce si vypočítá hodnotu $B = g^y \bmod n$.
6) Příjemce si vypočítá sdílený klíč $k = A^y \bmod n$.
7) Příjemce nechá přijaté *A* a své *B* podepsat nějakou důvěryhodnou autoritou a zašifruje tento podpis pomocí *k*.

8) Příjemce pošle nezabezpečeně odesílateli hodnotu B a zašifrovaný podpis $E_k(S_B(A, B))$.
9) Odesílatel si vypočítá sdílený klíč $k = B^x \bmod n$.
10) Odesílatel dešifruje zašifrovaný podpis E_k ($S_B(A, B)$) a následně ho ověří (porovná přijatý hash se svým hashem, který si vytvoří ze svého A a přijatého B)
11) Odesílatel nechá své A a přijaté B podepsat důvěryhodnou autoritou a zašifruje tento podpis pomocí k.
12) Odesílatel pošle příjemci zašifrovaný podpis $E_k(S_A(A, B))$.
13) Příjemce dešifruje zašifrovaný podpis $E_k(S_A(A, B))$ a následně ho ověří (porovná přijatý hash se svým hashem, který si vytvořil ze svého B a přijatého A).

Jestliže tento postup proběhne bezchybně (potvrdí se pravost podpisů), mohou si být komunikující strany jisty, že opravdu komunikují mezi sebou. Případný zásah útočníka do přenášených čísel A a B by se ukázal při ověřování podpisu (nerovnali by se hashe). Celý algoritmus zabezpečení protokolu DH před útokem „Man in the middle“ popisuje obrázek 3.4.

Jediný způsob jak by se mohl útočník nabourat mezi dvě komunikující strany, tak aby to komunikující strany nepoznaly je, že by útočník získal certifikáty s veřejnými klíči a také by znal soukromé klíče komunikujících stran využívané při digitálním podpisu, pomocí veřejných klíčů by dešifroval podpisy a poté by vytvořil nové hashe ze svých hodnot, které by následně zašifroval soukromými klíči. Tyto upravené podpisy by zašifroval pomocí tajných klíčů, které by si dohodl s komunikujícími stranami a následně by jim je tyto zašifrované klíče zaslal.



Obr. 3.4: Algoritmus ochrany před útokem „Man in the middle“ kdy jsou přenášená čísla A a B podepsány důvěryhodnou stranou.

Další způsob jak by se komunikující strany mohly bránit proti útoku „Man in the middle“ je, že by si po výměně čísel A a B ověřily jejich pravost nějakým dalším způsobem, například telefonicky.

Komunikující strany by se mohly také bránit útoku „Man in the middle“ tak, že by si vyměnily čísla A a B nějakým bezpečným kanálem. Tenhle postup však není moc efektivní, ztrácí se tím totiž všechny výhody asymetrické kryptografie. A rovnou by se mohl tímto bezpečným kanálem vyměnit tajný klíč, popřípadě požadovaná data.

4 PROTOKOL ELLIPTIC CURVE DIFFIE-HELLMAN (ECDH)

Protokol Elliptic Curve Diffie-Hellman je novější verzí klasického protokolu Diffie-Hellman. Algoritmus tohoto protokolu je analogický s původním algoritmem protokolu Diffie-Hellman. Komunikujícím stranám umožňuje získat přes veřejný nezabezpečený kanál tajný klíč, který mohou komunikující strany použít k zašifrování a následnému dešifrování přenášené zprávy. V tomto protokolu je ovšem modulární aritmetika nahrazena aritmetikou budovanou na základě operací s body na eliptické křivce.

4.1 Eliptické křivky

Užití eliptických křivek pro návrh asymetrických kryptosystémů k ustanovení tajného klíče nebo k podpisu (k šifrování se příliš nepoužívají) poprvé navrhli v roce 1985 nezávisle na sobě Victor Miller a Neal Koblitz. U asymetrických kryptosystémů využívajících ke své činnosti eliptické křivky se hierarchicky volí dva typy algebraických struktur: konečné těleso a eliptická křivka reprezentující grupu bodů, nad níž je vlastní asymetrický algoritmus definován. Volba obou těchto algebraických struktur významně ovlivňuje bezpečnost a efektivitu kryptosystému. Požadavky kladené na tyto dvě struktury spolu vzájemně souvisí [13].

Pro kryptografické účely je možné slovo *křivka* nahradit termínem *grupa*. Je možné tedy říct, že je definována grupa *E*. Při práci v rovině má bod ležící na křivce souřadnice (*x*, *y*), kde *x* a *y* jsou reálná čísla.

Eliptická křivka nad reálnými čísly je definována jako množina bodů (*x, y*), jejíž prvky vyhovují rovnici (4.1)[13]:

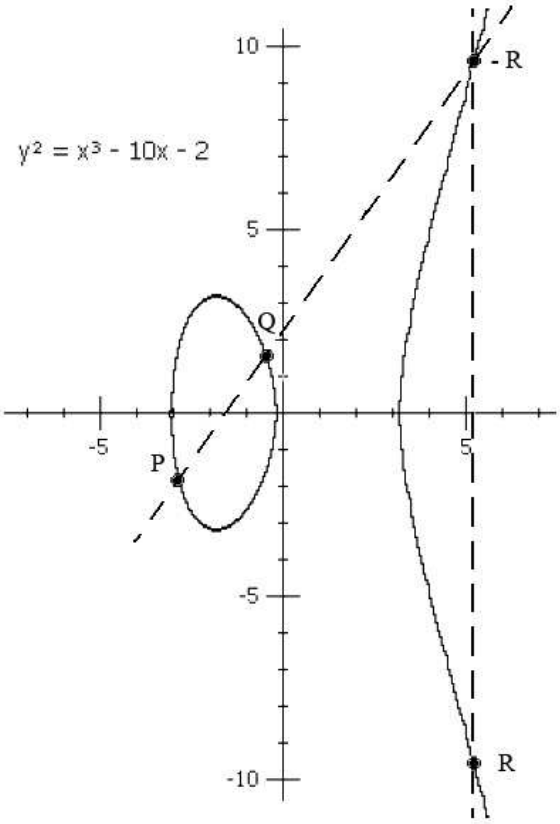
$$y^2 = x^3 + ax + b \tag{4.1}$$

Kde *x, y, a, b* jsou reálná čísla.

Důvodem proč se u eliptických křivek nezůstává u reálných čísel a zavádí se konečné tělesa je, že výpočty nad množinou reálných čísel jsou pomalé a mnohdy kvůli zaokrouhlovacím chybám i nepřesné. Kryptografie vyžaduje přesné a rychlé výpočty. V praxi jsou tedy zaváděny grupy založené na eliptických křivkách nad konečnými tělesy typu *Fp* a F_2^m. Základní vlastností grup používaných v kryptografii je, že mají konečný počet prvků.

Eliptická křivka *E* nad tělesem *F* se značí *E*(*F*). Počet bodů (mohutnost) na eliptické křivce *E* se značí *#E*(*F*) nebo jen *#E*.

Na obrázku 4.1 je vidět příklad eliptické křivky s ukázkou geometrického sčítání dvou bodů na eliptické křivce.



Obr. 4.1: Ukázka eliptické křivky a geometrického sčítání bodů $P + Q = R$.

4.1.1 Eliptická křivka nad tělesem *Fp*

Eliptická křivka nad tělesem *Fp* (kde *p* je prvočíslo) je definována jako množina bodů $P = (x, y)$, kde souřadnice *x* a *y* jsou z tělesa *Fp* a splňují rovnici (4.2) [13], k této množině bodů je dále nutné připočítat bod v nekonečnu *O*. Koeficienty *a* a *b* v rovnici (4.2) jsou prvky tělesa *Fp* (eliptická křivka je jimi určená) a musí splňovat rovnici (4.3) [13].

$$y^2 \bmod p = (x^3 + ax + b) \bmod p \tag{4.2}$$

$$4a^3 + 27b^2 \pmod p \neq 0 \tag{4.3}$$

Podmínka z rovnice (4.3) zaručuje, že takto definovaná množina bodů tvoří grupu. Koeficienty *a* a *b* je jinak možné volit libovolně, označují se jako veřejné parametry příslušného kryptosystému.

Eliptické křivky nad tělesem *Fp* jsou vhodnější pro softwarovou realizaci kryptosystému. Používané hodnoty *p* jsou [13]:

$p = 2^{192} - 2^{64} - 1$

$p = 2^{224} - 2^{96} - 1$

$p = 2^{256} - 2^{224} + 2^{192} + 2^{96} - 1$

$p = 2^{384} - 2^{128} - 2^{96} - 2^{32} - 1$

$p = 2^{521} - 1$

Algebraický součet bodů P a Q

Jsou dány body $P = (x_P, y_P)$ a $Q = (x_Q, y_Q)$. Opačný bod k bodu Q je $-Q=(x_Q, y_Q \bmod p)$. Pro sčítání je definováno, že $P \neq Q$ a součet $P + Q = R$. Souřadnice bodu R se určí s využitím rovnic (4.4), (4.5), (4.6) [13] následujícím způsobem:

$$s = \frac{y_P - y_Q}{x_P - x_Q} \bmod p \tag{4.4}$$

$$x_R = (s^2 - x_P - x_Q) \bmod p \tag{4.5}$$

$$y_R = (-y_P + s \cdot (x_P - x_R)) \bmod p \tag{4.6}$$

Hodnota s je definována jako směrnice přímky protínající body P a Q.

Zdvojení bodu P

Pokud je zapotřebí bod P ležící na eliptické křivce vynásobit například číslem 2, je nutné bod P zdvojit, to znamená sečíst ho se s sebou samým $P + P = 2P = R$. Kdyby bylo zapotřebí vynásobit bod P číslem 3, musel by se nejprve bod P zdvojit a pak by se k němu musel přičíst bod P ($R + P = 3P$). Zdvojení bodu P pro $y_P \neq 0$ se provede s využitím rovnic (4.7), (4.8), (4.9) [13] následujícím způsobem:

$$s = \frac{3x_P^2 + a}{2y_P} \bmod p \tag{4.7}$$

$$x_R = (s^2 - 2x_P) \bmod p \tag{4.8}$$

$$y_R = (-y_P + s \cdot (x_P - x_R)) \bmod p \tag{4.9}$$

Kde a je jeden z parametrů určující eliptickou křivku a p je prvočíslo.

4.1.2 Eliptická křivka nad tělesem F_2^m

Prvky tělesa F_2^m jsou m-bitové posloupnosti. Aritmetické operace v tělese F_2^m lze definovat buď v reprezentaci polynomiální, nebo v reprezentaci optimální normální báze. Protože prvky F_2^m jsou bitové posloupnosti, tak počítačová realizace aritmetických operací je velmi efektivní [13].

Eliptická křivka s podložním tělesem F_2^m je určena výběrem prvků a a b z F_2^m. Při výběru čísla b je nutné dodržet podmínku $b \neq 0$. Eliptická křivka nad tělesem F_2^m zahrnuje všechny body $P = (x, y)$ (kde souřadnice x a y jsou prvky z F_2^m), které splňují rovnici eliptické křivky nad tělesem F_2^m (4.10) [13].

$$y^2 + xy = x^3 + ax^2 + b \tag{4.10}$$

Body ležící na eliptické křivce společně s bodem v nekonečnu O tvoří grupu eliptické křivky nad tělesem F_2^m. Nutno zdůraznit, že takováto eliptická křivka má velký a zároveň konečný počet bodů a její aritmetika nemá žádnou zaokrouhlovací chybu.

Vlastnosti spojené s binární povahou pole vedou k tomu, že početní operace mohou být velmi efektivně implementovány hardwarově[13]. Používané hodnoty m pak jsou[13]:

$m = 163$

$m = 233$

$m = 283$

$m = 409$

$m = 571$

Algebraický součet bodů P a Q

Jsou dány body $P = (x_P, y_P)$ a $Q = (x_Q, y_Q)$. Opačný bod k bodu Q je $-Q = (x_Q, x_Q + y_Q)$. Pro sčítání je definováno, že $P \neq -Q$ a součet $P + Q = R$. Souřadnice bodu R se určí s využitím rovnic (4.11), (4.12), (4.13) [13] následujícím způsobem:

$$s = \frac{y_P - y_Q}{x_P + x_Q} \tag{4.11}$$

$$x_R = s^2 + s + x_P + x_Q + a \tag{4.12}$$

$$y_R = s \cdot (x_P + x_R) + x_R + y_P \tag{4.13}$$

Hodnota s je definována jako směrnici přímky protínající body P a Q.

Zdvojení bodu P

Pokud je požadováno vynásobení bodu P ležícího na eliptické křivce číslem 2, je nutné bod P zdvojit, to znamená sečíst ho se s sebou samým $P + P = 2P = R$. Kdyby bylo požadováno vynásobení bodu P číslem 3, musel by se nejprve bod P zdvojit a pak by se k němu musel přičíst bod P ($R + P = 3P$). Jestliže $x_P = 0$, pak $2P = O$. Zdvojení bodu

P pro $x_P \neq 0$ se provede s využitím rovnic (4.14), (4.15), (4.16) [13] následujícím způsobem:

$$s = \frac{x_P + y_P}{x_P} \tag{4.14}$$

$$x_R = s^2 + s + a \tag{4.15}$$

$$y_R = x_P^2 + x_R(s+1) \tag{4.16}$$

Kde a je jeden z parametrů určující eliptickou křivku.

4.1.3 Bezpečnost eliptických křivek

Základní výhodou kryptosystémů založených na bázi eliptických křivek je jejich velká bezpečnost i při malé velikosti klíče. Kratší délka klíče vede ke kratším a rychlejším výpočtům algoritmů a tím tedy k menším požadavkům na parametry systému. Další výhoda je, že všechny kryptografické systémy využívající modulární aritmetiku je možné převést do systému využívající eliptické křivky. Dalo by se říct, že [13] existuje nekonečné množství konečných komutativních grup.

Bezpečnost systémů založených na eliptických křivkách je dána složitostí řešení eliptického diskrétního logaritmu.

Pro vynásobení bodu P ležícího na eliptické číslem 3, je nejprve nutné bod P zdvojit a následně ke zdvojenému P je nutné přičíst bod P, tím se dosáhne bodu $3P = 2P + P = T$. Určení bodu nP (v uvedeném případě $3P$) uvedeným způsobem je označováno jako skalární násobení bodu. Problém eliptického diskrétního logaritmu spočívá v obtížnosti získání čísla pomocí kterého bylo provedeno skalárního násobení bodu na eliptické křivce[9].

Příklad diskrétního logaritmu eliptické křivky

Je dána eliptická křivka nad tělesem F_{23} definovaná rovnicí: $y^2 = x^3 + 9x + 17$. Úkolem je určit diskrétní logaritmus n při základu $P = (16, 5)$ pro $T = (4, 5)$. Nejjednodušší ale časově velmi náročný způsob stanovení n je, počítání násobků P tak dlouho, dokud se nenalezne odpovídající T.

Devět prvních násobků P odpovídá následujícím bodům:

P = (16, 5), $2P$ = (20, 20), $3P$ = (14, 14), $4P$ = (19, 20), $5P$ = (13, 10), $6P$ = (7, 3), $7P$ = (8, 7), $8P$ = (12, 17), $\mathbf{9P = (4, 5)}$.

Je vidět, že $\mathbf{9P = (4, 5) = T}$, je tedy možné říct, že $n = 9$ je diskrétní logaritmus T při základu P. V praxi číslo n je voleno hodně velké, aby určení tohoto čísla bylo pro útočníka co nejsložitější.

4.2 Obecné stanovení tajného bodu pomocí ECDH

K ustanovení tajného klíče (bodu) přes veřejný kanál potřebují komunikující strany znát parametry vybrané eliptické křivky a výchozí bod $V = (x_V, y_V)$. Tyto parametry jsou veřejné. Obecný postup stanovení tajného klíče bude následující:

1) Odesílatel a příjemce pečlivě vyberou eliptickou křivku E (s parametry a a b) a na ní si zvolí výchozí veřejný bod $V = (x_V, y_V)$.
2) Odesílatel si zvolí tajné náhodně velké přirozené číslo c.
3) Příjemce si zvolí tajné náhodně velké přirozené číslo d.
4) Odesílatel si vypočítá skalárním násobením nový bod na eliptické křivce $M = c \cdot V$ a následně ho nezabezpečeně pošle příjemci.
5) Příjemce si vypočítá skalárním násobením nový bod na eliptické křivce $N = d \cdot V$ a následně ho nezabezpečeně pošle odesílateli.
6) Odesílatel si vypočítá skalárním násobením tajný bod K ležící na eliptické křivce, $K = c \cdot N$.
7) Příjemce si vypočítá skalárním násobením tajný bod K ležící na eliptické křivce, $K = d \cdot M$.

Tajný bod K reprezentuje společný tajný klíč. Pokud vše proběhne korektně, komunikující strany dospějí ke stejnému tajnému bodu K, je to dáno tím, že vycházejí ze stejného výchozího bodu V. Vzhledem ke složitosti výpočtu diskrétního logaritmu eliptické křivky je téměř nemožné, aby útočník vypočítal hodnotu c nebo d z bodů M nebo N a tyto hodnoty by pak použil k výpočtu tajného bodu K ležícího na eliptické křivce.

Tento postup může být různě modifikován, odesílatel i příjemce mohou být dopředu dohodnuti na eliptické křivce a na výchozím bodě, přes veřejný kanál by se potom posílaly pouze hodnoty M, N.

4.2.1 Ustanovení tajného bodu pomocí ECDH nad tělesem *Fp*

Je uvažována eliptická křivka nad tělesem f_{11} (tedy $p = 11$), dále jsou uvažovány parametry křivky $\boldsymbol{a = 1, b = 3}$. Rovnice eliptické křivky bude:

$$y^2 = x^3 + x + 3$$

Nyní je nutné ověřit jestli platí rovnice (4.3) pro vytvoření grupy:

$$4a^3 + 27b^2 \pmod{p} \neq 0$$

$$4 \cdot 1^3 + 27 \cdot 3^2 \pmod{11} \neq 0$$

$$247 \pmod{11} \neq 0$$

$$5 \neq 0$$

Rovnice platí => je možné použít danou eliptickou křivku.

Na zadané eliptické křivce leží 20 bodů + bod v nekonečnu *O*. Tabulka 4.1 znázorňuje 20 bodů ležících na eliptické křivce.

Tab. 4.1: Body ležící na eliptické křivce $y^2 = x^3 + x + 3$.

(3, 0)	(5, 1)	(7, 1)	(10, 1)	(9, 2)
(1, 4)	(4, 4)	(6, 4)	(0, 5)	(11, 5)
(0, 6)	(11, 6)	(1, 7)	(4, 7)	(6, 7)
(9, 9)	(5, 10)	(7, 10)	(10, 10)	(3, 11)

Výchozí bod na eliptické křivce je volen ***V* = (6, 4)**, nyní se ještě může pomocí rovnice 4.2 ověřit, že opravdu tento bod leží na eliptické křivce:

$y^2 \bmod p = (x^3 + ax + b) \bmod p$

$4^2 \bmod 11 = (6^3 + 6 + 3) \bmod 11$

$5 = 5$

Rovnice platí => bod *V* leží na eliptické křivce $y^2 = x^3 + x + 3$.

Odesílatel si zvolil tajné číslo ***c* = 2**, příjemce si zvolil tajné číslo ***d* = 3**.

Odesílatel si vypočítá bod $M = c \cdot V = 2V$.

$$s' = \frac{3x_V^2 + a}{2y_V} \bmod p = \frac{3 \cdot 6^2 + 1}{2 \cdot 4} \bmod 11 = \frac{109}{8} \bmod 11$$

Nyní je nutné najít takové číslo *X*, které když bude vynásobeno jmenovatelem zlomku dané směrnice modulo dané prvočíslo *p* (11), tak se dosáhne čísla jedna.

$X \cdot 8 \bmod 11 = 1$

Bylo nalezeno vyhovující číslo $X = 7$, platí tedy: $7 \cdot 8 \bmod 11 = 1$.

Výsledná směrnice bude rovna $s = 109 \cdot 7 \bmod 11 = 4$.

$$x_M = (s^2 - 2x_V) \bmod p = (4^2 - 2 \cdot 6) \bmod 11 = 4$$

$$y_M = (-y_V + s \cdot (x_V - x_M)) \bmod p = (-4 + 4 \cdot (6 - 4)) \bmod 11 = 4$$

***M* = (4, 4) – vyhovuje (leží na eliptické křivce viz. tab. 4.1)**.

Příjemce si vypočítá bod $N = d \cdot V = 3V$, tento bod se získá tak, že se k bodu $2V$ přičte bod *V*. V=(6, 4), $2V$=(4, 4).

$$s = \frac{y_V - y_{2V}}{x_V - x_{2V}} \bmod p = \frac{4-4}{6-4} \bmod 11 = 0$$

$$x_N = (s^2 - x_V - x_{2V}) \bmod 11 = (0^2 - 6 - 4) \bmod 11 = -10 \bmod 11 = 1$$

$$y_N = (-y_V + s \cdot (x_V - x_N)) \bmod p = (-4 + 0 \cdot (6-1)) \bmod 11 = -4 \bmod 11 = 7$$

***N* = (1, 7) – vyhovuje (leží na eliptické křivce viz. tab. 4.1)**.

Odesílatel si vypočítá tajný bod $K = c \cdot N = 2N$.

$$s' = \frac{3x_N^2 + a}{2y_N} \bmod p = \frac{3 \cdot 1^2 + 1}{2 \cdot 7} \bmod 11 = \frac{2}{7} \bmod 11$$

$X \cdot 7 \bmod 11 = 1$

Bylo nalezeno vyhovující číslo $X = 8$, platí tedy: $8 \cdot 7 \bmod 11 = 1$.
Výsledná směrnice bude rovna $s = 2 \cdot 8 \bmod 11 = 5$.

$$x_K = (s^2 - 2x_N) \bmod p = (5^2 - 2 \cdot 1) \bmod 11 = 1$$

$$y_K = (-y_N + s \cdot (x_N - x_K)) \bmod p = (-7 + 5 \cdot (1-1)) \bmod 11 = -7 \bmod 11 = 4$$

***K* = (1, 4) – vyhovuje (leží na eliptické křivce viz. tab. 4.1)**.

Příjemce si vypočítá tajný bod $K = d \cdot M = 3M$.
Nejprve je nutné vypočítat $2M$ a následně k $2M$ pak přičíst $M = (4,4)$. $K = 2M + M$.

$$s' = \frac{3x_M^2 + a}{2y_M} \bmod p = \frac{3 \cdot 4^2 + 1}{2 \cdot 4} \bmod 11 = \frac{49}{8} \bmod 11$$

$X \cdot 8 \bmod 11 = 1$

Bylo nalezeno vyhovující číslo $X = 7$, platí tedy: $7 \cdot 8 \bmod 11 = 1$.
Výsledná směrnice bude rovna $s = 49 \cdot 7 \bmod 11 = 2$.

$$x_{2M} = (s^2 - 2x_M) \bmod p = (2^2 - 2 \cdot 4) \bmod 11 = -4 \bmod 11 = 7$$

$$y_{2M} = (-y_M + s \cdot (x_M - x_{2M})) \bmod p = (-4 + 2 \cdot (4-7)) \bmod 11 = -10 \bmod 11 = 1$$

2M = (7, 1)

$K = M + 2M$

$$s = \frac{y_M - y_{2M}}{x_M - x_{2M}} \bmod p = \frac{4-1}{4-7} \bmod 11 = -1 \bmod 11 = 10$$

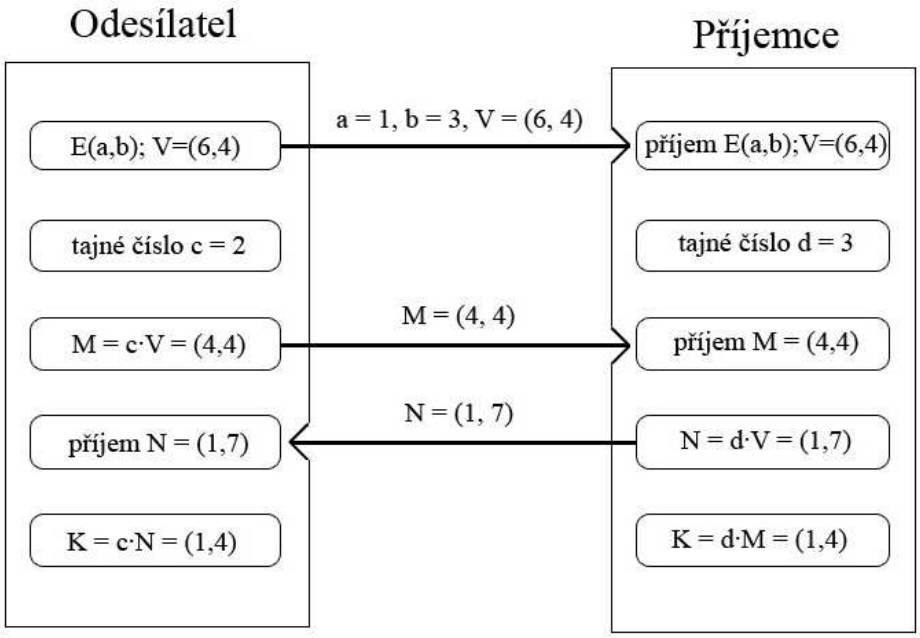
$$x_K = (s^2 - x_M - x_{2M}) \bmod 11 = (10^2 - 4 - 7) \bmod 11 = 89 \bmod 11 = 1$$

$$y_K = (-y_M + s \cdot (x_M - x_K)) \bmod p = (-4 + 10 \cdot (4-1)) \bmod 11 = 26 \bmod 11 = 4$$

***K* = (1, 4) – vyhovuje (leží na eliptické křivce viz. tab. 4.1)**.

Při této ukázce byly pro jednoduchost výpočtu zvoleny malá čísla (*a*, *b*; *p*) a malý výchozí bod (*V*), v praxi by se kvůli bezpečnosti volily mnohem větší čísla (větší bod).

Je vidět, že odesílatel a příjemce došli ke stejnému tajnému bodu ***K* = (1, 4)**. Výpočet tajného bodu tedy proběhl bezchybně. Nyní by mohl být tento tajný bod, respektive tajný klíč použít k šifrování pomocí nějakého symetrického algoritmu. Celý tento postup popisuje obrázek 4.2.

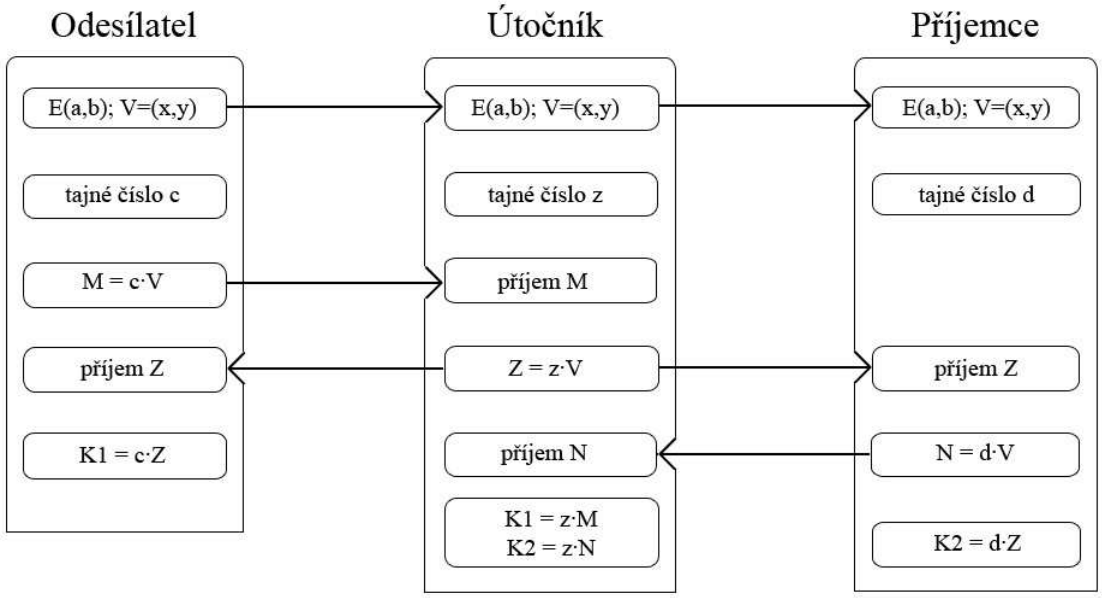


Obr. 4.2: Algoritmus stanovení tajného bodu *K* na eliptické křivce nad tělesem *Fp*.

Kdyby bylo požadováno stanovit tajný bod (klíč) pomocí eliptické křivky nad tělesem respektive s podložním tělesem $F_2{}^m$, tak by se postupovalo obdobně jako při stanovení tajného bodu pomocí eliptické křivky nad tělesem *Fp*, využily by se teoretické poznatky z kapitoly 4.1.2 s tím rozdílem, že by se počítalo s bitovými posloupnostmi.

4.3 Útok „Man in the middle“ v ECDH

Kdyby útočník chtěl provést útok „Man in the middle“ na protokol Elliptic Curve Diffie-Hellman, jeho postup by analogický k postupu útoku na původní verzi protokolu Diffie-Hellman. Útočník vstoupí mezi komunikující strany a začne se vydávat za odesílatele respektive příjemce, dohodne si vlastní tajné body na eliptické křivce s oběma stranami a s pomocí těchto tajných bodů, respektive tajných klíčů dešifruje všechny vyměněné zprávy mezi odesílatelem a příjemcem bez toho, aby komunikující strany zjistily, že si klíč vyměnily s někým jiným. Útočník tedy bude znát dva tajné body na eliptické křivce. Pomocí prvního tajného bodu respektive tajného klíče K_1 dešifruje přijatou zprávu od odesílatele (může ji i pozměnit), ten ji zašifroval pomocí K_1, následně ji zašifruje pomocí druhého tejného klíče K_2 a odešle ji příjemci, ten si ji poté dešifruje pomocí svého tajného klíče K_2.



Obr. 4.3: Obecný algoritmus útoku „Man in the middle“ na protokol ECDH.

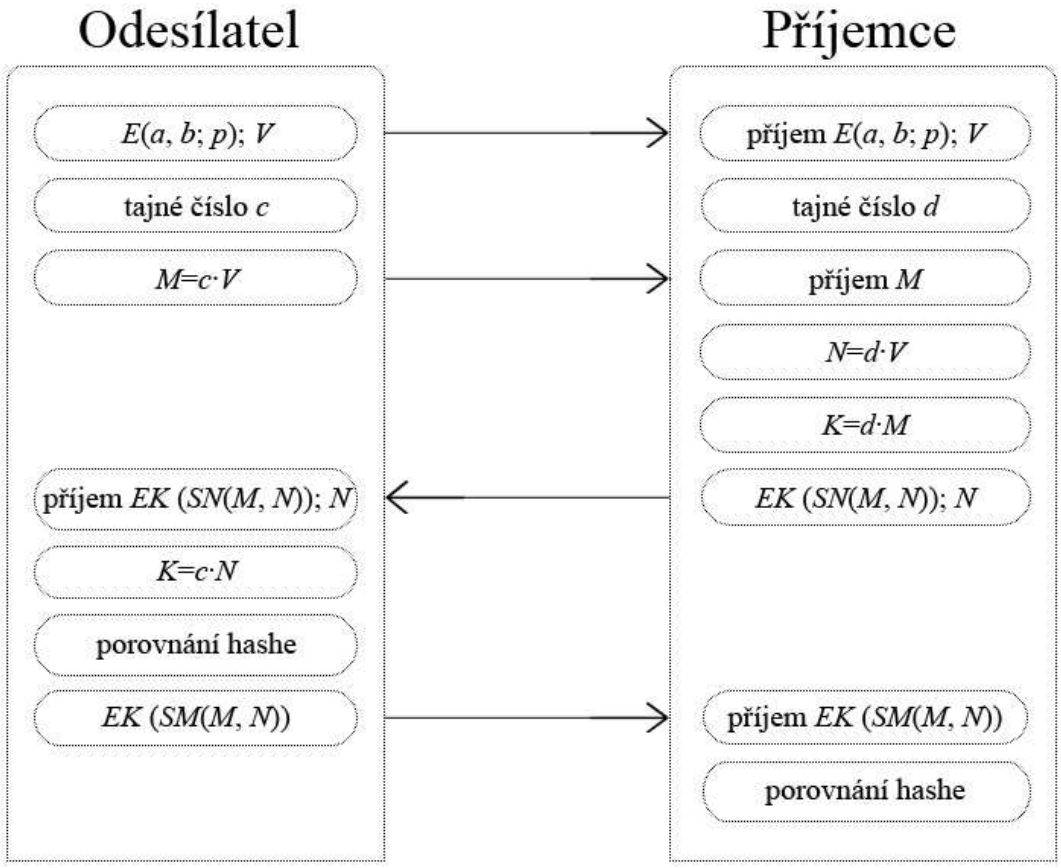
4.4 Ochrana před útokem „Man in the middle“ v ECDH

Jestliže je útok „Man in the middle“ v protokolu Elliptic Curve Diffie-Hellman jakousi analogií k útoku k původnímu protokolu Diffie-Hellman, tak i ochrana před tímto útokem v protokolu ECDH bude také analogická.

Jeden ze způsobů jak zabránit útočníkovi provést útok „Man in the middle“ je analogický se způsobem, který byl uveden v kapitole 3.6. Odesílatel a příjemce si nechají navzájem podepsat jejich zprávy (v případě protokolu ECDH body M a N– uvedené v kapitole 4.2) další důvěryhodnou stranou.

Při tomto postupu se předpokládá, že odesílatel vlastní certifikát s veřejným klíčem příjemce a příjemce vlastní certifikát s veřejným klíčem odesílatele. Oba tyto certifikáty byly podepsány nějakou důvěryhodnou certifikační autoritou mimo právě probíhající

domluvu klíče. Odesílatel i příjemce nakonec ověří pravost podpisu protistrany. Celý tento algoritmus popisuje obrázek 4.4.



Obr. 4.4: Algoritmus ochrany před útokem „Man in the middle“ kdy jsou přenášené body *M* a *N* podepsány důvěryhodnou stranou.

Dalším způsob jak by se komunikující strany mohly bránit proti útoku „Man in the middle“ je, že by si po výměně bodů *M* a *N* ověřily jejich pravost nějakým dalším způsobem, například telefonicky.

Komunikující strany by se mohly také bránit útoku „Man in the middle“ tak, že by si vyměnily parametry eliptické křivky a výchozí bod *V* nějakým bezpečným kanálem, dříve než by se samotná dohoda na tajném bodě uskutečnila. Tím se však ztrácí základní výhoda asymetrické kryptografie a rovnou by se mohl tímto bezpečným kanálem vyměnit tajný klíč, nebo celá požadovaná data.

5 KOMPATIBILITA PROTOKOLU DH S PROKOLEM ECDH

Činnost prvních kryptografických protokolů byla postavena na využití modulární aritmetiky, tento fakt platí i pro protokol Diffie-Hellman. V dnešní době jsou pro svoji velkou bezpečnost zaváděny systémy využívající eliptické křivky, příkladem může být protokol Elliptic Curve Diffie-Hellman. Eliptické křivky mají tu výhodu, že se dají snadno implementovat tam, kde je využívána modulární aritmetika. Vymyšlené algoritmy daného systému jsou zachovány, dochází pouze k nahrazení modulární aritmetiky aritmetikou budovanou na základě operací s body na eliptické křivce. Příkladem může být protokol Diffie-Hellman a jeho novější verze Elliptic Curve Diffie-Hellman. Algoritmy obou protokolů jsou k sobě analogické, avšak každý z nich k výpočtu tajného klíče využívá jinou aritmetiku.

Systémy využívající modulární aritmetiku jsou v kryptografii už několik desítek let zavedeny a využívány. V dnešní době jsou pro svoji bezpečnost hojně zaváděny systémy využívající eliptické křivky. Nyní však vyvstává problém jak dosáhnout kompatibility „starých“ zavedených systémů využívající modulární aritmetiku s „novými“ systémy využívající eliptické křivky. Tento problém vzniká i mezi protokolem Diffie-Hellman a protokolem Elliptic Curve Diffie-Hellman. Následující podkapitoly se budou zabývat řešením tohoto problému.

5.1 Porovnání protokolu DH s protokolem ECDH

Pro nalezení kompatibility mezi oběma protokoly je nutné si uvědomit, které kroky mají ve svých algoritmech stejné a ve kterých se liší. Tyto shody respektive rozdíly popisuje tabulka 5.1.

Tab. 5.1: Porovnání protokolu DH s protokolem ECDH.

krok	Diffie-Hellman	Elliptic Curve Diffie-Hellman
1	Dohoda na prvočísle n a na přirozeném čísle g.	Dohoda na parametrech eliptické křivky E (a, b) a na bodě V.
2	Odesílatel si zvolí přirozené číslo x.	Odesílatel si zvolí přirozené číslo c.
3	Příjemce si zvolí přirozené číslo y.	Příjemce si zvolí přirozené číslo d.
4	Odesílatel si vypočítá číslo $A = g^x \bmod n$ a odešle ho příjemci.	Odesílatel si vypočítá bod $M = c \cdot V$ a odešle ho příjemci.
5	Příjemce si vypočítá číslo $B = g^y \bmod n$ a odešle ho odesílateli.	Příjemce si vypočítá bod $N = d \cdot V$ a odešle ho odesílateli.
6	Odesílatel si vypočítá číslo $k = B^x \bmod n$.	Odesílatel si vypočítá bod $K = c \cdot N$.
7	Příjemce si vypočítá číslo $k = A^y \bmod n$.	Příjemce si vypočítá bod $K = d \cdot M$.

Z tabulky je vidět, že k matematické shodě dochází pouze v kroku 2 a 3. Ostatní kroky, ve kterých nedochází ke shodě budou nyní rozebrány.

Krok 1: V protokolu **DH** se volí prvočíslo n, např. $n = 11$ a číslo g, např. $g = 6$. V protokolu **ECDH** se volí parametry eliptické křivky, např. $a = 1$, $b = 3$ a dále těleso nad kterým je definována, např. Fp kde p je prvočíslo, např. $p = 11$. Poté se volí výchozí bod V, který leží na eliptické křivce, např. $V = (6, 4)$.

Je vidět, že hned v prvním kroku v protokolu ECDH je výběr parametrů sofistikovanější než v protokolu DH. V protokolu ECDH jsou voleny dvě čísla, jedno prvočíslo a jeden bod na eliptické křivce. Zatímco v protokolu DH je voleno jedno číslo a jedno prvočíslo.

Krok 4: V protokolu **DH** si odesílatel vypočítá číslo $A = g^x \bmod n$, např. $A = 6^{12} \bmod 11 = 3$ a odešle ho příjemci. V protokolu ECDH si odesílatel vypočítá bod $M = c \cdot V$, který leží na eliptické křivce, např. $M = 2 \cdot V = 2 \cdot (6,4) = (4,4)$ a odešle ho příjemci.

V tomto kroku je vidět hlavní rozdíl mezi oběma protokoly. V protokolu DH je číslo A vypočítáno pomocí diskrétního mocnění. Kdežto v protokolu ECDH je bod M vypočítán pomocí skalárního násobení bodu V přirozeným číslem c.

Krok 5: Tento krok je analogický s krokem 4, s tím rozdílem, že nyní dané číslo (bod) vypočítá příjemce a odešle ho odesílateli.

Krok 6: V protokolu **DH** si odesílatel vypočítá tajné číslo $k = B^x \bmod n$, např. $k = 8^{12} \bmod 11 = 9$. V protokolu **ECDH** si odesílatel vypočítá tajný bod $K = c \cdot N$, který leží na eliptické křivce, např. $K = 2 \cdot N = 2 \cdot (1, 7) = (1, 4)$.

V tomto kroku je opět vidět základní rozdíl mezi oběma protokoly. V protokolu **DH** je na výstupu tajná informace reprezentovaná číslem a u protokolu ECDH je na výstupu tajná informace reprezentovaná bodem na eliptické křivce.

Krok 7: Tento krok je analogický s krokem 6, s tím rozdílem, že nyní si dané tajné číslo (bod) vypočítá příjemce.

Z tohoto přehledu, je vidět, že počet rozdílných kroků mezi oběma verzemi je podstatný. Při detailnějším pohledu na jednotlivé rozdílné kroky je vidět, že kroky 4 až 7 jsou z hlediska aritmetických operací v jednotlivých protokolech stejné. Protokol DH v krocích 4 až 7 používá k výpočtu daného čísla diskrétní mocnění využívající modulární aritmetiku. Protokol ECDH v krocích 4 až 7 používá k výpočtu daného bodu skalární násobení bodu přirozeným číslem, k tomuto početnímu úkonu využívá aritmetiku budovanou na základě operací s body na eliptické křivce. Hlavním problémem tedy je, že každý protokol využívá ke svému výpočtu tajné informace jiný aritmetický úkon (jinou aritmetiku). Výsledkem tedy je, že oba protokoly dospějí k jiné tajné informaci, DH k číslu, ECDH k bodu.

5.2 Propojení modulární aritmetiky s aritmetikou na eliptické křivce

Z předchozí kapitoly vyplývá, že řešením vzájemné kompatibility mezi protokolem Diffie-Hellman a protokolem Elliptic Curve Diffie-Hellman by bylo propojení modulární aritmetiky (diskrétního mocnění) s aritmetikou budovanou na základě operací s body na eliptické křivce (skalární násobení bodu přirozeným číslem). Je však vůbec možné tyto rozdílné aritmetiky nějakým způsobem propojit? Touto otázkou se bude zabývat tato kapitola.

Nyní bude ukázána rozdílnost jednotlivých aritmetických úkonů.

Příklad diskrétního mocnění s využitím modulární aritmetiky bude následující:

$$g^x \bmod n = 3^2 \bmod 7 = 2$$

Příklad skalárního násobení bodu na eliptické křivce přirozeným číslem s využití aritmetiky budované na základě operací s body na eliptické křivce bude následující:

Je dán bod $V = (6, 4)$, ležící na eliptické křivce, dále jsou dány parametry eliptické křivky $a = 1$, $b = 3$ ($y^2 = x^3 + x + 3$), křivka je definována nad tělesem Fp, kde $p = 11$. Dále je dáno přirozené číslo $c = 2$.

1.) určení směrnice dané eliptické křivky:

$$s' = \frac{3x_V^2 + a}{2y_V} \bmod p = \frac{3 \cdot 6^2 + 1}{2 \cdot 4} \bmod 11 = \frac{109}{8} \bmod 11$$

$X \cdot 8 \bmod 11 = 1$; $X = 7$

$s = 109 \cdot 7 \bmod 11 = 4$

2.) protože je požadováno daný bod V vynásobit číslem 2, musí se provést zdvojení tohoto bodu:

$$x_{2V} = (s^2 - 2x_V) \bmod p = (4^2 - 2 \cdot 6) \bmod 11 = 4$$

$$y_{2V} = (-y_V + s \cdot (x_V - x_M)) \bmod p = (-4 + 4 \cdot (6 - 4)) \bmod 11 = 4$$

$c \cdot V = 2 \cdot (6, 4) = (4, 4)$

Hned na první pohled je vidět, že skalární násobení bodu na eliptické křivce je mnohem sofistikovanější než diskrétní mocnění. Výpočet diskrétního mocnění byl proveden na jednom řádku, kdežto skalární násobení bodu číslem 2 na eliptické křivce bylo provedeno na pěti řádcích (s větším číslem by počet řádků výpočtů rostl). Při výpočtu diskrétního mocnění je nutné znát dvě čísla (g, x) a jedno prvočíslo (n). Při skalárním násobením bodu na eliptické křivce je nutné znát parametry eliptické křivky (a, b), těleso nad kterým je definovaná (např. Fp, kde p je prvočíslo). Dále pak bod na dané eliptické křivce (např. V) a přirozené číslo, kterým chceme daný bod vynásobit (např. c).

I když se vezme v úvahu fakt, že skalární násobení bodu na eliptické křivce využívá ke svému výpočtu jednotlivých souřadnic daného bodu, ale i k výpočtu směrnice dané eliptické křivky modulární aritmetiku dojde se k tomu, že matematické postupy jednotlivých aritmetických úkonů v jednotlivých protokolech jsou rozdílné. Výsledky jednotlivých aritmetických úkonů jsou také naprosto rozdílné, u protokolu DH je výsledkem číslo a u protokolu ECDH je výsledkem bod na eliptické křivce.

Při hledání vzájemné kompatibility mezi jednotlivými verzemi obou protokolů je nutné brát v potaz fakt, že u protokolu DH odesílatel respektive příjemce předpokládá, že mu protistrana pošle číslo vypočítané na základě stejných vstupních číslech (n, g), se kterými disponuje i on sám, pomocí tohoto čísla poté vypočítá dané tajné číslo. To samé platí i pro protokol ECDH s tím rozdílem, že odesílatel respektive příjemce předpokládá, že mu protistrana pošle souřadnice bodu vypočítaného na základě stejných vstupních parametrech eliptické křivky a stejného výchozího bodu ležícího na dané eliptické křivce. Pomocí tohoto bodu poté vypočítá souřadnice daného tajného bodu ležícího na eliptické křivce.

Pokud by u protokolu DH příjemce, respektive odesílatel přijal od protistrany číslo vypočítané na základě jiných vstupních čísel (n, g), stalo by se, že každá z komunikujících stran by si vypočítala jiné tajné číslo, nedošlo by ke shodě. A poroto že algoritmy obou protokolů jsou k sobě analogické, tak tento fakt platí i pro protokol ECDH. Tedy pokud by u protokolu ECDH příjemce, respektive odesílatel přijal od protistrany bod vypočítaný na základě jiných vstupních parametrech eliptické křivky (a, b) a jiném výchozím bodě (V), stalo by se, že každá z komunikujících stran by si vypočítala jiný tajný bod ležící na jiné eliptické křivce, nedošlo by taktéž ke shodě.

Z faktů uvedených v předchozích odstavcích vyplývá, že propojení modulární aritmetiky (kterou využívá protokol DH) s aritmetikou budovanou na základě operací s body na eliptické křivce (kterou využívá protokol ECDH) není reálné.

Dále pokud by bylo předpokládáno, že systém pracující s protokolem DH umí pouze provádět diskrétní mocnění, protože ke své funkci výpočtu tajného klíče ani jiný aritmetický úkon umět nepotřebuje, nebylo by možné daný systém zuniverzálnit tak, aby uměl pracovat s diskrétním mocněním i se skalárním násobením bodu na eliptické křivce přirozeným číslem, tedy aby byl schopný vykonávat algoritmus protokolu DH a protokolu ECDH (tím by se dal vyřešit problém komunikace mezi jednotlivými protokoly), protože systém využívající protokol ECDH potřebuje ke své činnosti mimo jiné ovládat aritmetické úkony jako sčítání, odčítání, násobení, dělení.

Opačný případ by mohl být reálný, tedy případ kdy by byl systém pracující s protokolem ECDH doplněn algoritmem protokolu DH. V tomto případě kdyby daný systém pracující s protokolem ECDH doplněný o algoritmus protokolu DH chtěl komunikovat se systémem využívající protokol DH by se přepnul do režimu, kdy by využíval protokol DH a celá domluva na tajném klíči by poté probíhala prostřednictvím protokolu DH. Toto řešení by v praxi mohlo být reálné, ale z hlediska bezpečnosti, kdy jsou pro svoji velkou bezpečnost zaváděny eliptické křivky zcela nevhodné. Bezpečnost domluvy na tajném klíči by se několikanásobně zmenšila.

5.3 Systém pracující s protokolem DH a protokolem ECDH

Z předchozí kapitoly vyplývá, že nelze žádným reálným způsobem dosáhnout propojení modulární aritmetiky (diskrétní mocnění–protokol DH) s aritmetikou bodovanou na základě operací s body na eliptické křivce (skalární násobení bodu na eliptické křivce přirozeným číslem–protokol ECDH) tak, aby obě komunikující strany dosáhly stejného výsledku, tedy stejné tajné informace.

Možným řešením problému vzájemné kompatibility mezi protokolem Diffie-Hellman a protokolem Elliptic Curve Diffie-Hellman by mohl být universální systém, který by dokázal pracovat s protokolem DH a protokolem ECDH. Tento sytém by fungoval jako prostředník mezi protokolem DH a protokolem ECDH.

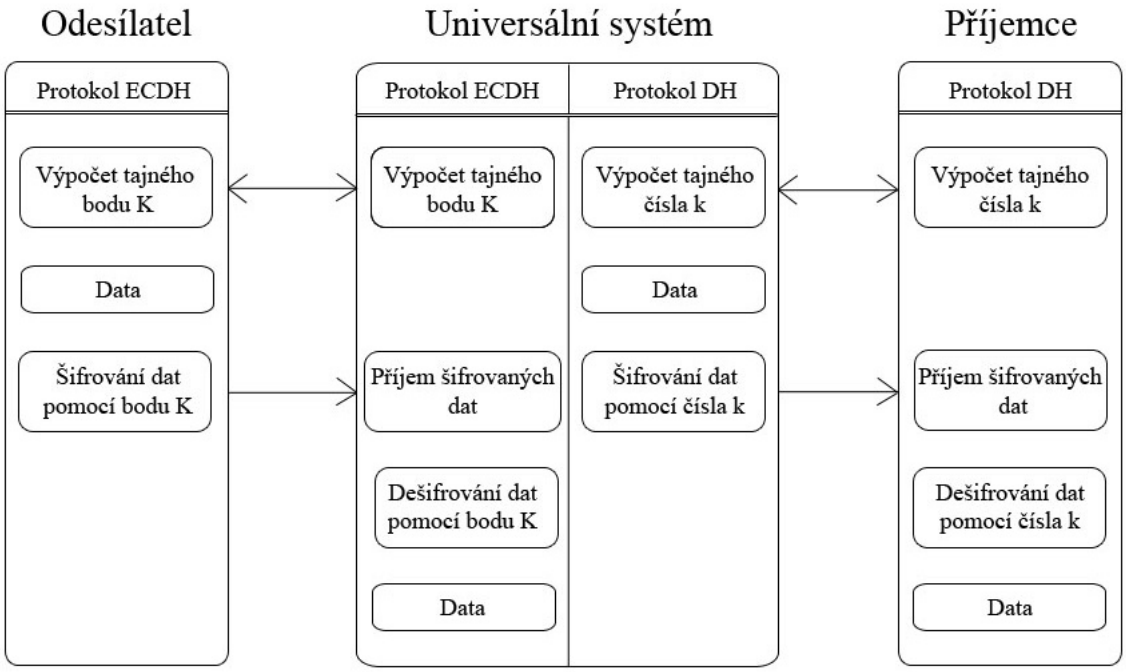
5.3.1 Návrh systému propojujícího protokol DH a ECDH

Tento systém, jak už z názvu vyplývá by měl být schopen pracovat s protokolem DH i protokolem ECDH. Tedy měl by být schopen pracovat s modulární aritmetikou (s diskrétním mocněním) a s aritmetikou budovanou na základě operací s body na eliptické křivce (se skalárním násobením bodu přirozeným číslem).

Protože při využití eliptických křivek se dosahuje mnohem větší bezpečnosti než při využití modulární aritmetiky, měl by být tento systém umístěn na té straně, kde se nachází systém pracující s protokolem DH. Pro dosáhnutí maximální bezpečnosti by měl tento systém v ideálním případě stát hned vedle systému pracujícího s protokolem DH. Tento systém by fungoval jako jakýsi propojující rozhraní mezi systémem využívající protokol DH a systémem využívající protokol ECDH.

Funkce tohoto systému by se mohla přirovnat k útoku „Man in the middle“ s tím rozdílem, že by se nejednalo o žádný útok na přenášenou tajnou informaci. Vše by probíhalo v naprostém bezpečí a nikdo jiný kromě komunikujících stran by nemohl zjistit přenášený tajný klíč. Vzájemné propojení jedné strany pracující s protokolem ECDH s druhou stranou pracující s protokolem DH při posílání dat přes veřejný kanál by bylo následující. Odesílatel pracující s protokol ECDH by si pomocí tohoto protokolu dohodnul (vypočítal) s univerzálním systémem tajný bod na eliptické křivce (v tomto okamžiku by universální systém zastupoval příjemce–protokol DH), poté by si universální systém pomocí protokolu DH s příjemcem využívající tento protokol dohodnul (vypočítal) tajné číslo (v tomto okamžiku by zastupoval odesílatele–protokol ECDH). Odesílatel by pomocí vypočítaného tajného bodu zašifroval požadovaná data a odeslal by je danému universálnímu systému, tento systém by je dešifroval pomocí tajného bodu a následně je zašifroval pomocí tajného čísla. Tato zašifrovaná data by poté odeslal příjemci. Příjemce by nakonec zašifrovaná data pomocí tajného čísla dešifroval.

Obrázek 5.1 názorně ukazuje vzájemné propojení protokolu ECDH s protokolem DH s využitím universálního systému při dohodě na tajném bodě a čísle a při posílání dat přes veřejný kanál.



Obr. 5.1: Propojení protokolu ECDH s protokolem DH s využitím universálního systému při dohodě na tajném bodě a čísle a při posílání dat přes veřejný kanál.

5.3.2 Obecný postup propojení protokolu ECDH s protokolem DH

K propojení protokolu Elliptic Curve Diffie-Hellman s protokolem Diffie-Hellman bude využit universální systém, který umí pracovat s oběma verzemi protokolu. Obecný postup propojení odesílatele (protokol ECDH) s příjemcem (protokol DH) bude následující.

a) Výpočet tajného bodu a čísla.

1) Odesílatel s universálním systémem, který zastupuje příjemce si vybere eliptickou křivku E (definovanou parametry a a b a tělesem nad kterým leží, např. Fp) a na ní si zvolí výchozí veřejný bod $V = (xv, yv)$.
2) Odesílatel si zvolí tajné náhodně velké přirozené číslo c.
3) Universální systém zastupujíc příjemce si zvolí tajné náhodně velké přirozené číslo d.
4) Odesílatel si vypočítá skalárním násobením nový bod na eliptické křivce $M = c \cdot V$ a následně ho nezabezpečeně pošle universálnímu systému zastupujícího příjemce.
5) Universální systém zastupujíc příjemce si vypočítá skalárním násobením nový bod na eliptické křivce $N = d \cdot V$ a následně ho nezabezpečeně pošle odesílateli.
6) Odesílatel si vypočítá skalárním násobením tajný bod K ležící na eliptické křivce, $K = c \cdot N$.

7) Universální systém zastupujíc příjemce si vypočítá skalárním násobením tajný bod K ležící na eliptické křivce, $K = d \cdot M$.

8) Universální systém zastupujíc odesílatele se s příjemcem dohodne na prvočísle n a na přirozeném čísle g.
9) Universální systém zastupujíc odesílatele si zvolí tajné náhodně velké přirozené číslo x.
10) Příjemce si zvolí tajné náhodně velké přirozené číslo y.
11) Universální systém zastupujíc odesílatele si vypočítá číslo $A = g^x \bmod n$ a odešle ho nezabezpečeně příjemci.
12) Příjemce si vypočítá číslo $B = g^y \bmod n$ a odešle ho nezabezpečeně universálnímu systému zastupujícího odesílatele.
13) Universální systém zastupujíc odesílatele si vypočítá tajné číslo $k = B^x \bmod n$.
14) Příjemce si vypočítá tajné číslo $k = A^y \bmod n$.

b) Přenos šifrovaných dat po veřejném kanálu od odesílatele (protokol ECDH) přes universální systém k příjemci (protokol DH).

1) Odesílatel zašifruje s využitím nějakého symetrického algoritmu požadovaná data pomocí tajného bodu K a odešle je universálnímu systému, který zastupuje příjemce.
2) Universální systém zašifrovaná data přijatá od odesílatele dešifruje pomocí tajného bodu K (s využitím dohodnutého symetrického algoritmu).
3) Universální systém s využitím nějakého symetrického algoritmu zašifruje dešifrovaná data pomocí tajného čísla k a zastupujíc odesílatele je odešle příjemci.
4) Příjemce zašifrovaná data dešifruje pomocí tajného čísla k (s využitím dohodnutého symetrického algoritmu).

Před šifrováním a dešifrováním dat se musí odesílatel a příjemce, respektive odesílatel a universální systém a následně universální systém a příjemce dohodnout na symetrickém algoritmu pomocí kterého budou požadovaná data šifrována a následně dešifrována.

5.3.3 Propojení protokolu DH s protokolem ECDH

Při této ukázce propojení protokolu DH s protokolem ECDH budou využity příklady výpočtů tajného čísla a tajného bodu z kapitol 3.2 a 4.2.1. Nebudou zde uvedeny konkrétní výpočty daných čísel a bodů, ale bude zde uveden pouze jejich obecný zápis a výsledek.

Je uvažována následující situace. Odesílatel pracující s protokolem ECDH si chce s příjemcem, který pracuje s protokolem DH dohodnout tajný klíč pomocí kterého by s využitím nějakého symetrického algoritmu zašifroval požadovaná data a následně by je odeslal příjemci. Příjemce by poté zašifrovaná data pomocí tajného klíče dešifroval.

Odesílatel si pod pojmem tajný klíč představuje tajný bod na eliptické křivce a příjemce si pod pojmem tajný klíč představuje tajné číslo vypočítané pomocí modulární aritmetiky. Postup bude následující:

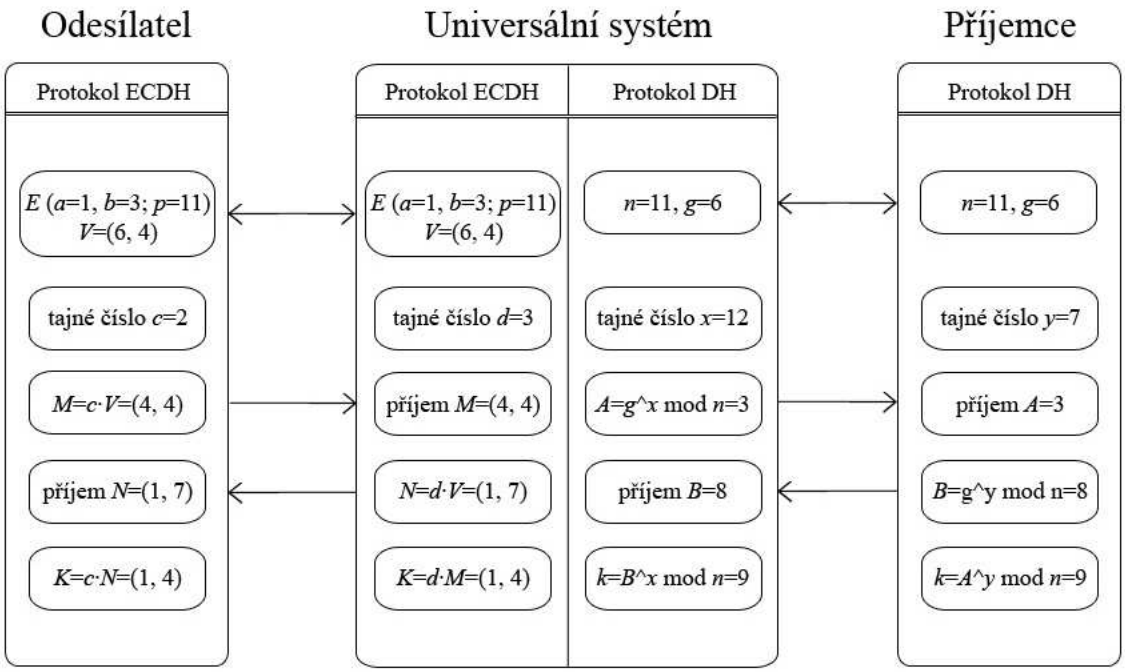
a) Výpočet tajného bodu a čísla odesílatele, universálního systému a příjemce.

1) Odesílatel se s universálním systémem, který v tomto okamžiku zastupuje příjemce dohodl na eliptické křivce $y^2 = x^3 + x + 3$ (**a = 1, b = 3**), která je definovaná nad tělesem F_{11} (**p = 11**), dále se dohodli na výchozím bodě **V = (6, 4)**, který leží na dané eliptické křivce.
2) Odesílatel si zvolil tajné číslo **c = 2**.
3) Universální systém zastupujíc příjemce si zvolil tajné číslo **d = 3**.
4) Odesílatel si vypočítal bod **$M = c \cdot V$ = 2·(6, 4) = (4, 4)** a následně ho odeslal universálnímu systému zastupujícího příjemce.
5) Universální systém zastupujíc příjemce si vypočítal bod **$N = d \cdot V$ =3·(6, 4)=(1, 7)** a následně ho odeslal odesílateli.
6) Odesílatel si vypočítal tajný bod **$K = c \cdot N$ = 2·(1, 7) = (1, 4)**.
7) Universální systém zastupujíc příjemce si vypočítal tajný bod **$K = d \cdot M$ = 3·(4, 4) = (1, 4)**.
8) Universální systém v tomto okamžiku zastupujíc odesílatele se s příjemcem dohodl na prvočísle **n = 11** a na přirozeném čísle **g = 6**.
9) Universální systém zastupujíc odesílatele si zvolil tajné číslo **x = 12**.
10) Příjemce si zvolil tajné číslo **y = 7**.
11) Universální systém zastupujíc odesílatele si vypočítal číslo **$A = g^x \bmod n = 6^{12} \bmod 11 = 3$** a odeslal ho příjemci.
12) Příjemce si vypočítal číslo **$B = g^y \bmod n = 6^7 \bmod 11 = 8$** a odeslal ho universálnímu systému zastupujícího odesílatele.
13) Universální systém zastupujíc odesílatele si vypočítal tajné číslo **$k = B^x \bmod n = 8^{12} \bmod 11 = 9$**.
14) Příjemce si vypočítal tajné číslo **$k = A^y \bmod n = 3^7 \bmod 11 = 9$**.

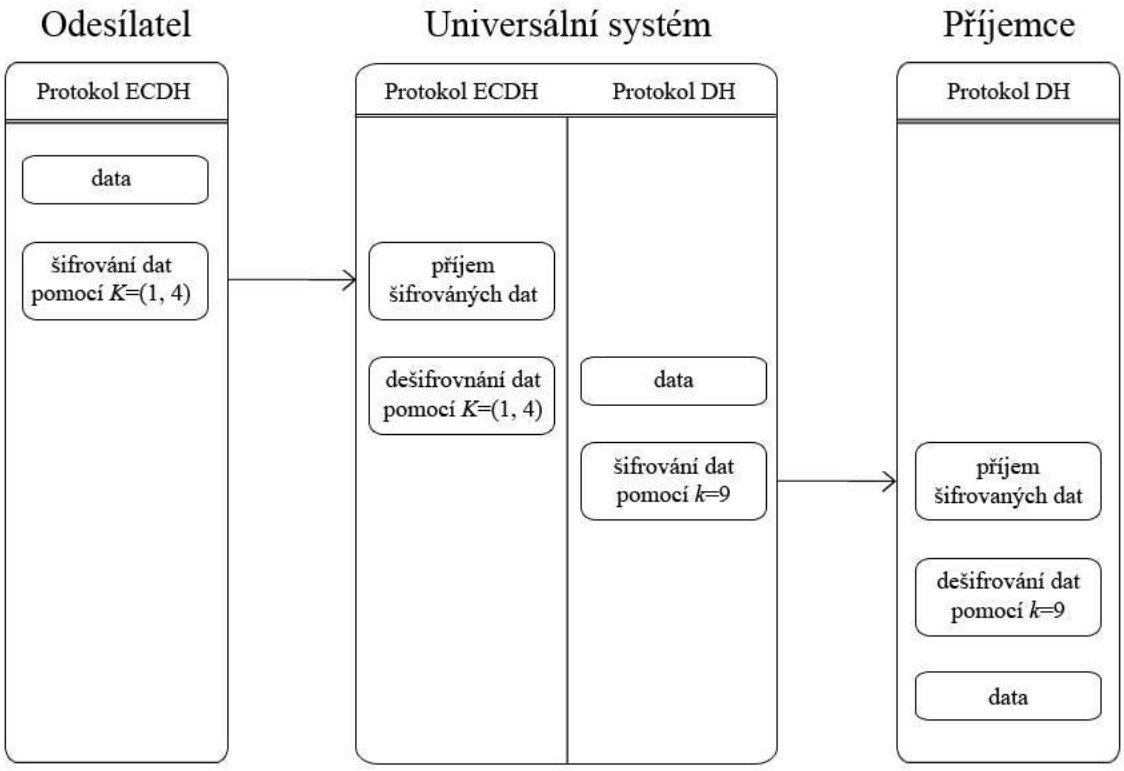
b) Přenos šifrovaných dat přes veřejný kanál od odesílatele (protokol ECDH) přes universální systém (protokol ECDH a protokol DH) k příjemci (protokol DH).

1) Odesílatel zašifroval požadovaná data pomocí tajného bodu **K = (1, 4)** a odeslal je universálnímu systému, který v tuto chvíli zastupoval příjemce.
2) Universální systém zašifrovaná data dešifroval pomocí tajného bodu **K = (1, 4)**.
3) Universální systém dešifrovaná data zašifroval pomocí tajného čísla **k = 9** a zastupujíc odesílatele je odeslal příjemci.
4) Příjemce zašifrovaná data dešifroval pomocí tajného čísla **k = 9**.

Obrázek 5.2 znázorňuje algoritmus výpočtu tajného bodu a čísla odesílatele (protokol ECDH), universálního systému (protokol ECDH a protokol DH) a příjemce (protokol DH). Obrázek 5.3 znázorňuje algoritmus přenosu šifrovaných dat od odesílatele přes universální systém k příjemci.



Obr. 5.2: Algoritmus výpočtu tajného bodu a čísla odesílatele, universálního systému a příjemce.



Obr. 5.3: Algoritmus přenosu šifrovaných dat od odesílatele přes universální systém k příjemci.

5.3.4 Útok „Man in the middle“ v propojení protokolů ECDH a DH

Na protokol Diffie-Hellman i na protokol Elliptic Curve Diffie-Hellman lze provést útok „Man in the middle“. Tento útok lze provést i v situaci kdy dochází k propojení protokolů ECDH a DH prostřednictvím universálního systému, který pracuje s oběma protokoly.

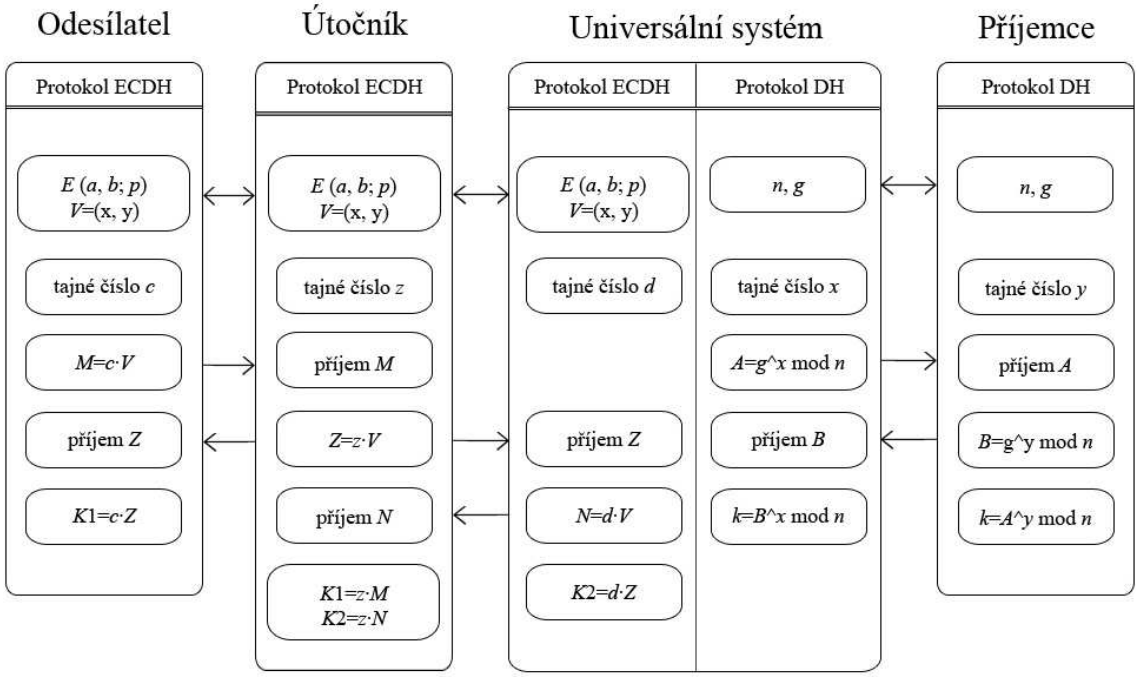
Protože universální systém, který zprostředkovává dané propojení je z důvodu bezpečnosti uvažován na straně kde je využit protokol DH, nejlépe hned vedle daného systému (tyto systémy by měly být propojeny privátním kanálem) lze provést útok „Man in the middle“ pouze na protokol ECDH, který je využit mezi universálním systémem a odesílatelem respektive příjemcem.

„Útok man in the middle“ v tomto případě bude totožný s útokem, který byl popsán v kapitole 4.3. Jediný rozdíl bude v tom, že útočník vstoupí mezi odesílatele respektive příjemce a universální systém a ne mezi odesílatele a příjemce.

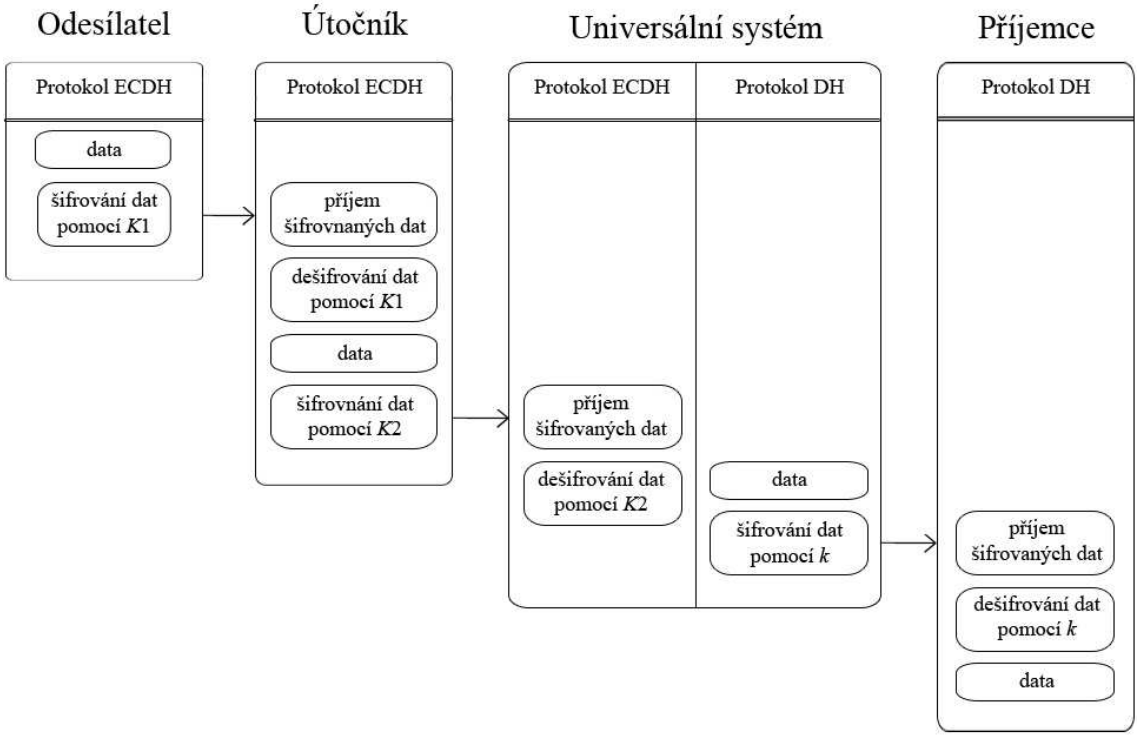
Jestliže bude uvažován případ kdy odesílatel bude používat protokol ECDH a příjemce bude používat protokol DH algoritmus útoku „Man in the middle“ bude následující:

1) Útočník vstoupí mezi odesílatele a universální systém.
2) Útočník odposlechne přenášené parametry eliptické křivky a výchozí bod.
3) Útočník vydávajíc se za universální systém (který v tomto okamžiku zastupuje příjemce) si dohodne s odesílatelem tajný bod na eliptické křivce K_1.
4) Útočník vydávajíc se za odesílatele si s universálním systémem (který v tomto okamžiku zastupuje příjemce) dohodne tajný bod na eliptické křivce K_2.
5) Universální systém zastupujíc odesílatele si s příjemcem dohodne tajné číslo *k*.
6) Odesílatel zašifruje data pomocí tajného bodu K_1 a odešle je útočníkovi vydávajícího se za universální systém.
7) Útočník přijatá data dešifruje pomocí tajného bodu K_1, může je i pozměnit, následně je zašifruje pomocí tajného bodu K_2 a odešle je universálnímu systému.
8) Universální systém přijatá data dešifruje pomocí tajného bodu K_2 a následně je zašifruje pomocí tajného čísla *k* a odešle je příjemci.
9) Příjemce přijatá data dešifruje pomocí tajného čísla *k*.

Obrázek 5.4 znázorňuje první část algoritmu útoku „Man in the middle“ kdy si útočník dohodne s každou z komunikujících stran (odesílatelem a universálním systémem) vlastní tajný bod na eliptické křivce a dále pak dohodu tajného čísla (universálního systému a příjemce), mezi kterou už útočník vstoupit nemůže. Obrázek 5.5 znázorňuje druhou část algoritmu útoku „Man in the middle“ kdy útočník dešifruje přenášená data a následně je opět zašifruje, dále je předá universálnímu systému, který je znovu dešifruje a následně zašifruje pomocí tajného čísla a předá je příjemci.



Obr. 5.4: První část algoritmu útoku „Man in the middle“, při které dochází k dohodě na tajných bodech na eliptické křivce a dále pak na dohodě na tajném čísle.



Obr. 5.5: Druhá část algoritmu útoku „Man in the middle“, při které dochází k přenosu dat od odesílatele přes universální systém k příjemci.

5.3.5 Ochrana před „Man in the middle“ v propojení protokolů ECDH a DH

Při této ochraně před útokem „Man in the middle“ v propojení protokolů ECDH a DH se předpokládá, že universální systém je se systémem pracujícím s protokolem DH spojen privátním kanálem, který nelze nijak odposlechnout a nelze na něj aplikovat útok „Man in the middle“. Ochrana před tímto útokem bude tedy zaměřena na protokol ECDH se kterým je přes veřejný kanál spojen universální systém se systémem pracujícím s ECDH. Ochrana před tímto útokem bude analogická s ochranou před útokem „Man in the middle“, která byla popsána v kapitole 4.4.

Protože je universální systém se systémem pracujícím s protokolem DH spojen privátním kanálem nebude mezi těmito systémy aplikována žádná ochrana před útokem „Man in the middle“, hodnoty mezi systémy budou přenášeny volně.

Jeden ze způsobů jak zabránit útočníkovi provést útok „Man in the middle“ v propojení protokolů ECDH a DH je, že si odesílatel respektive příjemce a universální systém nechají navzájem podepsat přenášené body ležící na eliptické křivce další důvěryhodnou stranou. Kdy by byl uvažován příklad propojení protokolů z kapitoly 5.3.3 jednalo by se o body M a N.

Při tomto postupu se předpokládá, že odesílatel má certifikát s veřejným klíčem universálního systému a universální systém má certifikát s veřejným klíčem odesílatele. Oba tyto certifikáty byly podepsány nějakou důvěryhodnou certifikační autoritou mimo právě probíhající domluvu klíče (byly podepsány již dříve). Následující postup popisuje jak si zabezpečeně proti útoku „Man in the middle“ odesílatel (protokol ECDH) s universálním systém vypočítá tajný klíč–bod K.

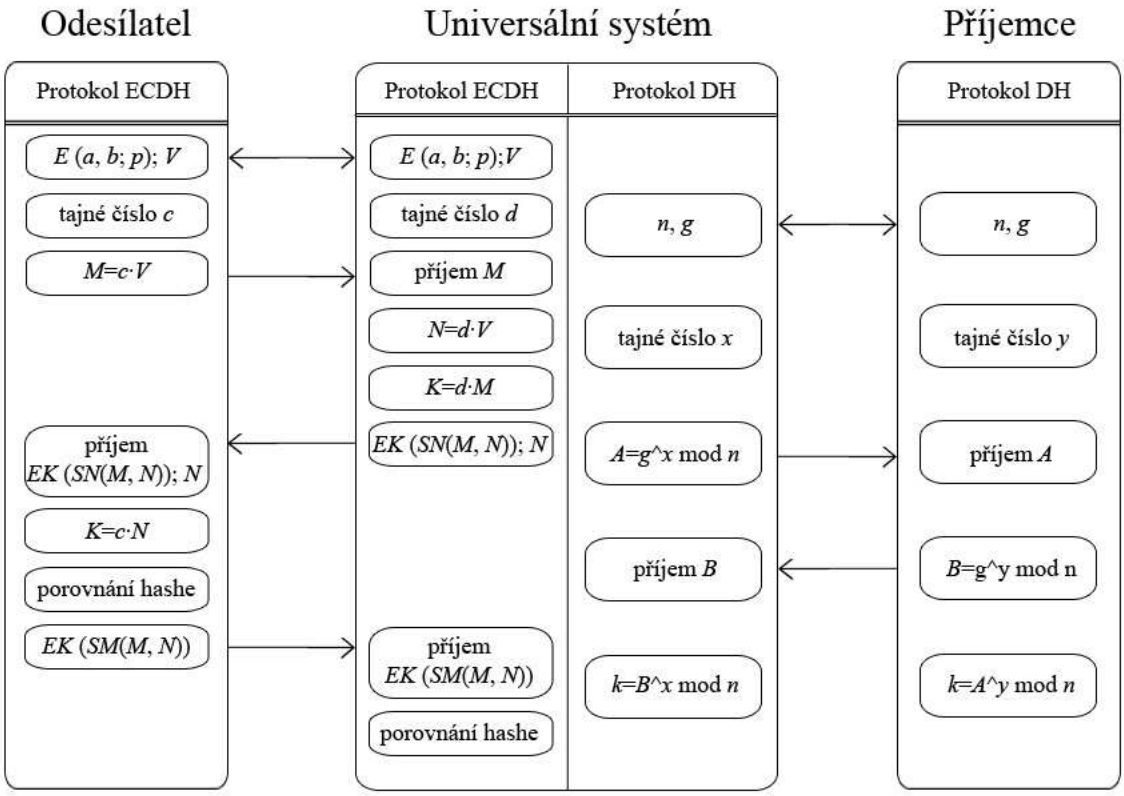
1) Odesílatel s universálním systémem, který zastupuje příjemce si vybere eliptickou křivku E (definovanou parametry a a b a tělesem nad kterým leží, např. Fp) a na ní si zvolí výchozí veřejný bod $V=(x_V, y_V)$.
2) Odesílatel si zvolí tajné náhodně velké přirozené číslo c.
3) Universální systém si zvolí tajné náhodně velké přirozené číslo d.
4) Odesílatel si vypočítá skalárním násobením nový bod na eliptické křivce $M = c \cdot V$ a následně ho nezabezpečeně pošle universálnímu systému.
5) Universální systém si vypočítá skalárním násobením nový bod na eliptické křivce $N = d \cdot V$.
6) Universální systém si vypočítá skalárním násobením tajný bod K ležící na eliptické křivce $K = d \cdot M$.
7) Universální systém nechá přijatý bod M a svůj bod N podepsat nějakou důvěryhodnou autoritou a zašifruje tento podpis pomocí bodu K.
8) Universální systém pošle nezabezpečeně odesílateli bod N a zašifrovaný podpis $E_K(S_N(M, N))$.
9) Odesílatel si vypočítá skalárním násobením tajný bod K ležící na eliptické křivce $K = c \cdot N$.
10) Odesílatel dešifruje zašifrovaný podpis $E_K(S_N(M, N))$ a následně ho ověří (porovná přijatý hash se svým hashem, který si vytvořil ze svého bodu M a přijatého N).

11) Odesílatel nechá svůj bod *M* a přijatý bod *N* podepsat důvěryhodnou autoritou a zašifruje tento podpis pomocí *K*.
12) Odesílatel pošle universálnímu systému zašifrovaný podpis $E_K(S_M(M, N))$.
13) Universální systém dešifruje zašifrovaný podpis $E_K(S_M(M, N))$ a následně ho ověří (porovná přijatý hash se svým hashem, který si vytvořil ze svého bodu *N* a přijatého *M*).

Jestliže tento postup proběhne bezchybně (potvrdí se pravost podpisů), mohou si být komunikující strany jisty, že opravdu komunikují mezi sebou. Případný zásah útočníka do přenášených hodnot bodů *M* a *N* by se projevil při ověřování podpisu (nerovnali by se hashe komunikujících stran).

Jediný způsob jak by se mohl útočník nabourat mezi odesílatele a universální systém, tak aby to ani jedna strana nepoznala by byl takový, že by útočník získal certifikáty s veřejnými klíči, a také by znal soukromé klíče komunikujících stran, které využívají při digitálním podpisu bodů ležících na eliptické křivce. Pomocí veřejných klíčů by dešifroval podpisy a poté by vytvořil nové hashe ze svých vlastních hodnot, které by následně zašifroval soukromými klíči. Tyto upravené podpisy by zašifroval pomocí tajných klíčů, které by si dohodl s komunikujícími stranami a následně by jim tyto zašifrované podpisy zaslal.

Obrázek 5.6 znázorňuje algoritmus ochrany před útokem „Man in the middle“ a dále pak domluvu na tajném klíči *k*. V tomto případě je k ochraně před tímto útokem využita ochrana kdy jsou přenášené body *M* a *N* podepsány důvěryhodnou autoritou.



Obr. 5.6: Algoritmus ochrany před útokem „Man in the middle“ v propojení protokolů DH a ECDH a domluva na tajném klíči *k*.

Další způsob jak by se komunikující strany mohly bránit proti útoku „Man in the middle“ by byl takový, že by si po výměně bodů *M* a *N* ověřily jejich pravost nějakým dalším způsobem, například telefonicky.

Komunikující strany by se mohly také bránit útoku „Man in the middle“ tak, že by si vyměnily body *M* a *N* nějakým bezpečným kanálem. Tenhle postup však není moc efektivní, ztrácí se tím totiž všechny výhody asymetrické kryptografie. A rovnou by se mohl tímto bezpečným kanálem vyměnit tajný klíč reprezentovaný bodem *K*, nebo rovnou požadovaná data.

5.3.6 Praktické využití systému propojujícího protokol DH a ECDH

V předchozích kapitolách bylo předvedeno, že lze protokol Diffie-Hellman propojit s protokolem Elliptic Curve Diffie-Hellman. Toto propojení však není úplně jednoduché. Aby se dané propojení dalo uskutečnit je k němu zapotřebí universální systém pracující jak s protokolem DH, tak s protokolem ECDH. Způsob tohoto propojení byl názorně předveden v kapitole 5.3.3. Protože v tomto způsobu propojení je využit universální sytém, který využívá protokol DH a ECDH je též toto propojení náchylné na útok „Man in the modele“. Opět se ale lze proti tomuto útoku bránit.

Nyní však vyvstávají dvě otázky:

1) Nebude celá komunikace, při které dochází k ustanovení tajného bodu a čísla a dále pak k přenosu dat, kdy jsou data prvně šifrována pomocí tajného bodu, následně dešifrována a znovu šifrována, ale už pomocí tajného čísla příliš pomalá a tudíž nevhodná pro běžnou komunikaci zúčastněných stran?
2) Nebude praktické zrealizování daného universálního systému příliš finančně náročné a tedy pro běžnou komunikaci zúčastněných stran nevhodné?

Ad1)

Jestliže při útoku „Man in the middle“ nepoznají (z hlediska celkového času) komunikující strany, že někdo vstoupil mezi ně a s každou stranou si dohodl svůj tajný klíč, kdy pomocí prvního odeslaná zašifrovaná data od odesílatele dešifroval a následně je zašifroval pomocí druhého tajného klíče a odeslal je příjemci, tak ani při využití universálního systému, který také přijatá zašifrovaná data dešifruje a následně zašifruje by neměly komunikující strany pocítit při dohodě na tajném klíči a přenosu dat výrazné zpomalení celé komunikace.

Samozřejmě, že dojde ke zpomalení dané komunikace, protože proces šifrování a dešifrování dat zabere určitý čas, ale při použití dostatečně výpočetně výkonného universálního sytému (rychlé šifrování a dešifrování dat) by mělo být dané zpoždění komunikace přijatelné.

Ad2)

Praktické zrealizování universálního systému by nemělo být o moc finančně náročnější než praktické zrealizování systému pracujícího s protokolem ECDH. Je to dáno tím, že systém pracující s protokolem ECDH dokáže provádět stejné aritmetické operace jako systém pracující s protokolem DH, obráceně to nemusí platit. Jediné rozšíření systému pracujícího s protokolem ECDH by bylo rozšíření o algoritmus protokolu DH, což by nemělo být nijak finančně náročné. Dále by měly být na tento universální systém kladeny vyšší požadavky z hlediska rychlosti šifrování a dešifrování dat, aby dané šifrování a dešifrování dat příliš nezpomalovalo komunikaci zúčastněných stran. Míra splnění tohoto požadavku je přímo úměrná s finanční náročností tohoto universálního systému.

Z uvedených odpovědí na dané dvě otázky vyplívá, že by tento universální systém umožňující propojení protokolu DH s protokolem ECDH by mohl být prakticky využitelný.

Dále by mohl být tento universální systém pracující s protokolem DH a protokolem ECDH využit v následující situaci:

- Určité množství kryptografických systémů pracuje s protokolem DH, tyto kryptografické systémy mají svůj jeden hlavní kryptografický systém, který je zastupuje, a který dále komunikuje se vzdáleným systémem využívající protokol ECDH.

V této situaci by mohl být daný hlavní kryptografický systém nahrazen universálním systémem pracujícím s protokolem DH a ECDH. Se systémem využívající protokol ECDH by se domlouval pomocí protokolu ECDH a se systémy využívající protokoly DH by se domlouval pomocí nich. Přenos dat mezi jednotlivými systémy by probíhal podle uvedených principů v kapitole 5.3.2. Jelikož v této situaci by byly obě dvě rozhraní (vstupní/výstupní) universálního systému sloužící pro dohodu na tajném čísle (protokol DH) a na tajném bodě (protokol ECDH) napojeny na nechráněné komunikační kanály, musela by se zajistit ochrana před útokem „Man in the middle“ jak pro protokol Elliptic Curve Diffie-Hellman, tak i pro protokol Diffie-Hellman. Způsoby těchto ochran jsou popsány v kapitolách 3.7 a 4.4.

ZÁVĚR

Úkolem této bakalářské práce byla analýza protokolu Diffie-Hellman (DH) a jeho novější verze Elliptic Curve Diffie-Hellman (ECDH). Podrobné rozebrání obou protokolů, určení jejich předností a jejich slabin, zanalyzování jejich vzájemné kompatibility a možný návrh řešící vzájemnou kompatibilitu.

Protokol Diffie-Hellman je protokol asymetrické kryptografie umožňující zúčastněným komunikujícím stranám výpočet tajného sdíleného čísla přes veřejný nezabezpečený kanál tak, aby nikdo nebyl schopný odposlechnout nebo vypočítat jeho hodnotu. To vše je možné díky tomu, že protokol DH pracuje s modulární aritmetikou při které jsou vzniklé výpočty chráněny složitostí výpočtu diskrétního logaritmu. Slabinou tohoto protokolu je, že neověřuje autentizaci zúčastněných stran. Této slabiny využívá útok „Man in the middle“. Realizace tohoto útoku spočívá v tom, že útočník vstoupí mezi komunikující strany a s každou ze zúčastněných stran si dohodne vlastní tajný klíč (číslo). Pomocí těchto dvou klíčů (čísel) dokáže přenášená data dešifrovat, přečíst (může je i pozměnit) a znovu zašifrovat tak, že to ani jedna z komunikujících stran nepozná. Nejužívanější ochrana před tímto útokem spočívá v podepsání přenášených vypočtených čísel nějakou důvěryhodnou autoritou

Protokol Elliptic Curve Diffie-Hellman je novější verzí protokolu DH. Tento protokol je analogický k původnímu protokolu DH, má stejné přednosti i slabiny, jediný rozdíl mezi těmito protokoly je, že protokol DH pracuje s čísly a protokol ECDH pracuje s body na eliptické křivce. I na protokol ECDH lze provést útok „Man in the middle“, také se lze proti němu bránit. Bezpečnost protokolu ECDH je mnohem větší. Je to dané tím, že v tomto protokolu je modulární aritmetika nahrazena aritmetikou budovanou na základě operací s body na eliptické křivce. Řešení eliptického diskrétního logaritmu je totiž mnohem složitější než řešení normálního diskrétního logaritmu. Protokol ECDH je velmi bezpečný i při použití menší velikosti klíče než u protokolu DH. Kratší klíč vede ke kratším a rychlejším výpočtům. Tedy na kryptosystémy pracující s protokol ECDH nemusí být kladeny tak velké požadavky z hlediska výpočetní rychlosti.

Při analýze vzájemné kompatibility mezi protokolem DH a protokolem ECDH jsem zjistil, že kvůli rozdílným aritmetickým úkonům jednotlivých protokolů není daná přímá kompatibilita možná. Navrhl jsem řešení umožňující vzájemnou kompatibilitu obou protokolů prostřednictvím universálního systému, který umí pracovat s oběma protokoly. Výhodou tohoto řešení je, že dokáže zprostředkovat komunikaci mezi protokoly DH a ECDH. Jeho nevýhodou však je, že dochází k výpočtu dvou tajných klíčů (čísla a bodu), a že přenášená data jsou dvakrát šifrována. To vše vede ke zpoždění celkové komunikace. V praxi by však s potřebou komunikace mezi jednotlivými protokoly mohlo toto řešení najít uplatnění.

LITERATURA

[1] HÁLA, V. *Kvantová kryptografie*. Aldebaran bulletin [online]. 2005, roč. 3, č. 14. Dostupný z WWW: <http://aldebaran.cz/bulletin/2005_14_kry.php>. ISSN 1214-1674.

[2] KŘIVKA, P. *Kryptografické systémy nad eliptickými křivkami*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2009. 50 s. Vedoucí bakalářské práce Ing. Peter Stančík.

[3] KŘÍŽ, J. *Softwarová podpora výuky kryptosystémů založených na problému diskrétního logaritmu*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2008. 74 s. Vedoucí diplomové práce doc. Ing. Karel Burda, CSc.

[4] MILOŠ, J. *Kryptografické metody zabezpečení dat*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2008. 43 s. Vedoucí bakalářské práce Ing. Petra Lambertová.

[5] PAJTINOVÁ, M. *Metody kvantové kryptografie*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2009. 49 s. Vedoucí bakalářské práce doc. Ing. Václav Zeman, Ph.D.

[6] PINKAVA, J. *Úvod do kryptografie*, květen 1998. 26 s. Dostupný z WWW: <https://akela.mendelu.cz/~lidak/bif/kryptografie.pdf >.

[7] POPOVSKÝ, M. *Útoky postranními kanály*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2009. 71 s. Vedoucí diplomové práce Ing. Zdeněk Martinásek.

[8] PŘIBYL, J. Matematické algoritmy - čínská věta o zbytcích, soubor přednášek na ČVUT, 2007. Dostupné z www: <http://euler.fd.cvut.cz/predmety/ma/files/ma-05-2007.pdf>

[9] PŘIBYL, J. *Šifry veřejného klíče*, přednáška na ČVUT, 2006. 38 s. Dostupné z WWW: <http://www.comtel.cz/files/download.php?id=2339>

[10] RYAN, P., SCHNEIDER, S. *The modeling and analysis of security protocols*. London: Addison Wesley Professional, 2001. 300 s.

[11] SANGMA, Z. *Hardware Implementation of Elliptic Curve Diffie-Hellman Key Agreement Scheme in GF(p)*. New York, 2008. 79 s. Dostupné z WWW: <https://ritdml.rit.edu/dspace/bitstream/1850/7899/1/ZSangmaThesis10-2008.pdf>

[12] SCHNEIER, B. *Applied cryptography. 2nd edition*. [s.l.]: John Wiley & Sons, 1996. 784 s. ISBN 0-471-11709-9 .

[13] SZTURC, J. *Softwarová podpora výuky kryptosystémů založených na eliptických křivkách*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2009. 65 s. Vedoucí diplomové práce doc. Ing. Karel Burda, CSc.

[14] SZTURC, J. *Šifrátory pro mobilní telefony*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2007. 54 s. Vedoucí bakalářské práce doc. Ing. Václav Zeman, Ph.D.

[15] TRUNČEK, P. *Kryptografické protokoly v praxi*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2009. 48 s. Vedoucí diplomové práce Ing. Tomáš Pelka.

[16] VYCHODIL, P. *Softwarová podpora výuky kryptosystémů založených na problému faktorizace velkých čísel*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2009. 67 s. Vedoucí diplomové práce doc. Ing. Karel Burda, CSc.

SEZNAM ZKRATEK

AES	Advanced Encryption Standard
BlowFISH	Blow FIbonacci SHrinking (cipher)
CAST	Algoritmus pro šifrování, iniciály autorů Carlisle Adams a Stafford Taverns
DES	Data Encryption Standard
DH	Diffie-Hellman
ECDH	Elliptic Curve Diffie-Hellman
FISH	FIbonacci SHrinking (cipher)
GOST	Gosudarstvennyi Standard
IBM	International Business Machines
IDEA	International data encryption algorithm
PGP	Pretty Good Privacy
RC2	Rivest Cipher 2
RC4	Rivest Cipher 4
RC5	Rivest Cipher 5
RSA	Algoritmus pro šifrování, iniciály autorů Rivest, Shamir, Aleman
SK	Soukromý klíč
TK	Tajný klíč
Triple DES	Triple Data Encryption Standard
TwoFISH	FIbonacci SHrinking (cipher)
USA	United States of America
VK	Veřejný klíč