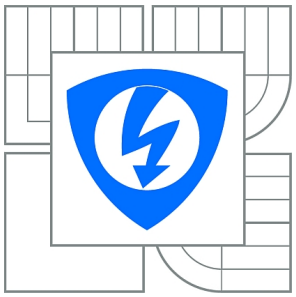
VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

SOFTWARE PRE PODPORU VÝUČBY BEZPEČNOSTNÝCH PROTOKOLOV RIADENIA PRÍSTUPU

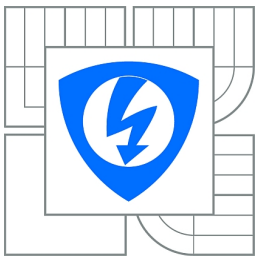
SOFTWARE FOR ASSISTED STUDY OF ACCESS CONTROL PROTOCOLS

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE — MICHAL KOPULETÝ
AUTHOR

VEDOUCÍ PRÁCE — Ing. LUKÁŠ VLČEK
SUPERVISOR

BRNO 2014



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky a komunikačních technologií

Ústav telekomunikací

Bakalářská práce

bakalářský studijní obor
Teleinformatika

Student: Michal Kopuletý ***ID:*** 146866
Ročník: 3 ***Akademický rok:*** 2013/2014

NÁZEV TÉMATU:

Software pre podporu výučby bezpečnostných protokolov riadenia prístupu

POKYNY PRO VYPRACOVÁNÍ:

Úlohou tejto práce bude zmapovať súčasnú situáciu na poli protokolov riadenia prístupu (access control), a to nielen v oblasti networking. Práca bude zameraná na ich rozbor, správanie v rôznych scenároch. Dôraz bude kladený na ich prednosti a slabiny. Súčasťou práce bude taktiež rozprava o možnostiach ich vzájomnej interoperability a trendoch v tejto oblasti. Výstupom práce budú výučbové materiály vo forme textu s nákresmi, a taktiež názorných animácii, rovnako i spätnoväzobný kvíz pre učiaceho sa. Samotná finálna forma edukačného materiálu bude spustiteľná vo web-prehliadači.

DOPORUČENÁ LITERATURA:

[1] HASSEL, Jonathan. RADIUS. 1st edition. O'Reilly Media, 2002. ISBN 0-596-00322-6.
[2] NT LAN Manager (NTLM) Authentication Protocol. [citované 2013-09-19] <http://download.microsoft.com/download/9/5/E/95EF66AF-9026-4BB0-A41D-A4F81802D92C/[MS-NLMP].pdf>.

Termín zadání: 10.2.2014 ***Termín odevzdání:*** 4.6.2014

Vedoucí práce: Ing. Lukáš Vlček
Konzultanti bakalářské práce:

doc. Ing. Jiří Mišurec, CSc.
Předseda oborové rady

UPOZORNĚNÍ:

ABSTRAKT

Tato bakalářská práce poskytuje čtenáři možnost proniknout do problematiky protokolů řízení přístupu v oblastech, jako jsou počítačové sítě, bankovnictví a přístupové karty.

Každá kapitola uvádí příklady často využívaných způsobů autentizace a uvádí jejich výhody a nevýhody, kterých by mohl využít případný útočník. Částečně jsou popsány i způsoby jakými probíhá autorizace.

Hlavním výstupem práce je výukový software spustitelný ve webovém prohlížeči. Pro lepší pochopení čtenářem, jsou výukové texty opatřeny animacemi probírané látky. Aby si student mohl ověřit svoje nabyté znalosti, obsahuje každá kapitola krátký vědomostní test.

Díky nastudování velkého množství autentizačních protokolů se mohl stát součástí práce návrh autentizačního protokolu umožňujícího implementaci do protokolu ACP. Navržený autentizační protokol dokáže přenášet informace pro široké množství autentizačních metod (hašování, symetrická kryptografie, asymetrická kryptografie).

KLÍČOVÁ SLOVA

Řízení přístupu, autentizace, autorizace, bezkontaktní karty, RFID, NFC, EMV, RADIUS, TACACS, DIAMETR, ACP.

ABSTRACT

This bachelor's thesis allows the reader to penetrate into the issues of access control protocols in area such as computer network, banking and access contactless cards.

Each chapter provides example of frequently used authentication methods and lists of the pros and cons. Partially describes the methods of authoritzation.

The main outcome of this bachelor's thesis is an educational software executable in a web browser. For the reader's better understanding there are, animation in the software. In order to verify student knowledge, there are test in teaching material.

The thesis proposal provides authentication protocol that can be implemented into ACP protocol. The proposed authentication protocol can transmit information to a wide number of authentication methods (hash, symmetric cryptography, asymmetric cryptography)

KEYWORDS

Access control, authentication, authorization, contactless card, RFID, NFC, EMV, RADIUS, TACACS, DIAMETR, ACP.

KOPULETÝ, M. *Software pre podporu výučby bezpečnostných protokolov riadenia prístupu.* Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2014. 73 s. Vedoucí bakalářské práce Ing. Lukáš Vlček.

PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma Software pre podporu výučby bezpečnostných protokolov riadenia prístup 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 bakalářské práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

V Brně dne ....

(podpis autora)

PODĚKOVÁNÍ

Děkuji vedoucímu bakalářské práce ing. Lukáši Vlčkovi, za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé bakalářské práce.

V Brně dne ....

(podpis autora)

OBSAH

SEZNAM OBRÁZKŮ

SEZNAM TABULEK

Úvod

Na internetu se můžeme setkat s velkým množstvím článků, pojednávajících o různých možnostech řízení přístupu žadatele k aktivům. Tedy o tom jak regulovat přístup žadatelů k něčemu o co mají zájem. A i když jsou to články z prestižních českých severů, setkáváme se v nich s častými záměnami pojmů. Je až dechberoucí, jak autoři těchto článků jsou schopni zaměňovat pojmy jako autentizace a autorizace.

Po přečtení práce případně přiloženého výukového materiálu, by již měl být čtenář schopen, vysvětlit všechny základní pojmy spojené s řízením přístupu. Zároveň práce mapuje současnou situaci na poli protokolů řízení přístupu a možnosti jejich zabezpečení v různých oblastech (vstup do budov, přístup k bankovnímu účtu, přihlášení k wi-fi síti apod.).

Text práce je částečně zdrojem podkladů využitých výukovým softwarem, který pro lepší pochopení protokolů řízení přístupu bude obsahovat interaktivní animace a vědomostní kvízy. Výukový materiál je spustitelný v běžném webovém prohlížeči a nic by nemělo bránit jeho širšímu rozšíření.

Výukový materiál se skládá z několika částí. První popisuje všeobecné pojmy, spojené s řízením přístupu. Druhá kapitola je spjata s řízením přístupu na základě přístupových karet. Další pojednává o využívaných biometrických systémech. Potom následuje blok věnovaný řízení přístupu v bankovnictví. A poslední kapitola názorně předvádí protokoly řízení přístupu využité v počítačových sítích.

Dalším bodem práce by mělo být zhodnocení interoperability protokolů řízení přístupu. Interoperabilita mezi stávajícími protokoly je však mizivá. Proto jsem se rozhodl popsat protokol ACP navržený doc. Burdou na Fakultě elektrotechniky VUT v Brně. ACP sice v praxi zatím není využíván, ale má potenciál být funkční ve všech oblastech řízení přístupu. V dosavadním návrhu protokolu není popsána žádná autentizační metoda využívající pouze ACP. V základní podobě se totiž ACP opírá o autentizaci zajištěnou zprávami EAP.

Na základě předchozího studia autentizačních metod jsem popsal autentizační mechanismus a jím využité zprávy, který by byl schopný provést autentizaci širokého množství zařízení za pomoci hašování, symetrické a asymetrické kryptografie.

1. Řízení přístupu

S řízením přístupu se setkáváme dennodenně a možná si to ani neuvědomujeme. Řízení přístupu je využíváno všude tam, kde je potřeba ověřit identita žadatele a umožnit mu přístup k *aktivům*. Pod slovním spojením „přístup k aktivům“ si můžeme představit otevření domovních dveří, provedení bankovní transakce nebo přihlášení do sítě.

V řízení přístupu vystupuje několik hlavních rolí. První z nich je *žadatel*, který se snaží přistoupit k aktivům. Žadatel většinou komunikuje s *autentizátorem*, ten provádí ověření identity. Dalším prvkem je *kontrolér* (*autorita)*, který rozhoduje o oprávnění žadatele k přístupu k aktivům. Následně *brána*, na základě rozhodnutí autority, umožní nebo odepře žadateli přístup k aktivům. Blíže se rolím jednotlivých zařízení věnuje [1]

1.1. Autentizace

Jednou z nejdůležitějších částí řízení přístupu je *autentizace*. Je to proces, kdy probíhá ověření identity žadatele. Pokud bychom chtěli příklad z běžného života, můžeme si představit předložení řidičského a občanského průkaz při silniční policejní kontrole. Způsobů jak ověřit identitu je nepřeberné množství. Všechny způsoby se však dají shrnout do tří hlavních skupin:

- Autentizace na základě znalosti
- Autentizace vlastností předmětu
- Autentizace biometrikou

Pro důkladnější ověření identity je možné využívat i více faktorovou autentizaci, jež kombinuje dvě nebo všechny tři výše zmíněné možnosti.

1.1.1. Autentizace na základě znalosti

V případě autentizace na základě znalosti je vyžadováno pro potvrzení identity znalost hesla. Pro správné fungování této metody se předpokládá, že heslo zná pouze jeho právoplatný majitel a má ho uloženo v paměti odkud ho vyjímá pouze při autentizaci. Autentizátor poté porovná heslo se svou databází, a pokud najde shodu, je autentizace považována za úspěšnou.

Pokročilejším způsobem autentizace je formou výzvy a na ní vytvořené odpovědi. Zde nejprve kontrolér vygeneruje náhodné číslo nebo řetězec, jež odešle žadateli o autentizaci. Žadatel následně aplikuje *hašovací funkci* na heslo a obdržené náhodné číslo. Takto zakódovaná zpráva je odeslána jako odpověď. Autentizátor ji poté porovnává s haši vytvořenými z náhodného čísla a hesel uložených v databázi. V případě, že najde stejný haš je žadateli povolen přístup.

Nejnovější a zároveň nejsofistikovanější je autentizace *důkazem nulové znalosti*. Tato technika je založena na tom, že jedna strana se snaží dokázat pomocí interaktivního důkazu, straně druhé pravdivost nějakého tvrzení, aniž by tato získala jakékoliv další informace kromě té, že tvrzení je pravdivé.

1.1.2. Autentizace na základě vlastnictví předmětu

Vlastností předmětu je zde myšlen lístek, průkaz, USB disk nebo čipová karta. Souhrnně se označují jako *autentizační tokeny*. Možnosti provedení autentizace pomocí tokenů jsou široké. Od pouhého přenosu nezašifrovaného hesla uloženého v paměti po tvorbu asymetrických klíčů. Vše je závislé jen od ceny a složitosti tokenu. V praxi jsou čipové karty

využívány jako bankovní platební karty, přístupové karty do budov. USB disky se pak využívají k uložení a ochranně soukromých klíčů a certifikátů.

1.1.3. AUTENTIZACE BIOMETRIKOU

Autentizace *biometrikou* je založena na jedinečnosti a neměnnosti dané charakteristiky. Kvůli nutnosti jedinečnosti je využívána pouze při autentizaci osob. Je postavena buď na odlišnostech ve fyziologii člověka, nebo v jeho chování. Rysů využívajících se k autentizaci je několik. Lze použít jedinečnost otisku prstů, barevných skvrn na duhovce, rozmístění krevního řečiště v sítnici, geometrie tváře a geometrie ruky. Metody, které mají základ v lidském chování, využívají jedinečnosti lidského hlasu, psaní podpisu, případně psaní na klávesnici.

1.2. AUTORIZACE

Autorizace je proces získávání souhlasu s provedením nějaké činnosti. Tím, že byla provedena autentizace, je jednoznačně potvrzena identita žadatele, na jejímž základě může autorita dále rozhodovat. Autorita prochází svůj přístupový seznam a vyhledává záznam odpovídající žadateli o přístup. Na základě čehož umožní, případně odepře přístup k aktivům.

1.3. AKTIVA

Pojmem aktiva obecně vyjadřujeme jakoukoliv službu, věc nebo třeba vstup do budovy, o který má žadatel zájem. Přístup k aktivům, ale bývá omezen pouze určitým osobám (zaplatili poplatek, mají dostatečnou bezpečnostní prověrku), které určuje poskytovatel služby. O tom, zda žadatel může přistoupit k aktivům, rozhodne proces autentizace a autorizace.

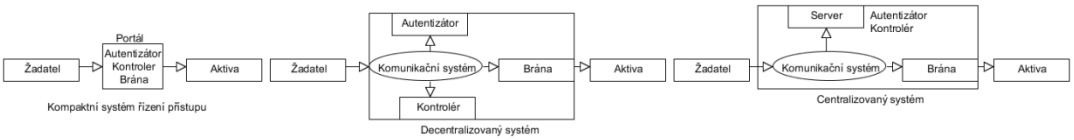
1.4. DRUHY SYSTÉMŮ

Autentizátor, brána i kontrolér jsou prvky využívané při řízení přístupu. Systémy se liší podle toho, jak jsou tyto prvky integrovány do jednoho zařízení případně rozloženy mezi několik prvků.

Kompaktní systémy jsou takové, které mají sloučenou funkci kontroléru, autentizátoru i brány v jednom zařízení. V případě takovéto integrace potom můžeme hovořit o portálu řízení přístupu.

Dalším systémem je *centralizovaný systém*. Ten už neslučuje veškeré prvky do jednoho zařízení, ale je rozdělen na bránu a server, plnící funkci kontroléru a autentizátoru. Žadatel pak může volně přistupovat jak k serveru tak i bráně.

Posledním možným rozložením systému je *decentralizovaná varianta*. V ní působí každý prvek na jiném zařízení. Systémy se pak jenom liší podle toho s kým je umožněno komunikovat žadateli. V případě, kdy žadatele má přístup k autentizátoru, bráně i kontroléru můžeme hovořit o systému s přímým připojením žadatel. Pokud však žadatel je schopný komunikovat pouze s některým z prvků (většinou to bývá brána), který zprostředkovává další komunikaci, jedná se o decentralizovaný systém se zprostředkovaným připojením žadatele.



Obr. 1.1 Systémy řízení přístupu

1.5. KOOPERACE SYSTÉMŮ

Kooperace systémů umožňuje žadateli přistupovat k aktivům jiného systému řízení přístupu, než u které je registrován. To funguje v případě, kdy jsou autority obou systémů propojeny a schopny využívat stejný protokol řízení přístupu. Takovéto spojení více systémů můžeme nazvat síť s kooperujícími systémy řízení přístupu.

Aby mohl žadatel přistoupit do cizího systému, musí být nejdříve autentizován svou vlastní autoritou. V případě úspěšné autentizace, pak tato autorita musí být schopna zaručit identitu žadatel cizí autoritě, do jejíhož systému se žadatel snaží přistoupit.

Pokud si autority navzájem důvěřují, už nic nebrání tomu, aby žadatel přistupoval k aktivům, které nepatří jeho domovské síti.

Podmínkou této funkce je využívání stejného protokolu, protože interoperabilita protokolů řízení přístupu zatím není dostatečná.

2. ZÁKLADY KRYPTOGRAFIE

Při řízení přístupu je nutné, aby některé přenášené zprávy zůstali důvěrné. Jinak řečeno aby obsah zpráv znali pouze autorizované komunikující strany. Abychom toho mohli dosáhnout, musíme využít kryptografie. *Kryptografie* je věda o konstrukci matematických transformací určených k zajišťování důvěrnosti zpráv.

Kromě důvěrnosti zpráv nám kryptografie umožní ověřovat i *autentičnost zpráv*. To je zaručení se za pravdivost informace o původu zprávy, jejím obsahu případně o jejím autorovi.

2.1. HAŠOVÁNÍ

Hašovací funkce slouží k vyjádření téměř libovolně dlouhé zprávy pomocí krátkého řetězce dat h = H(Z). Můžeme tedy říct, že se jedná o kompresní funkci. Základními požadavky jsou jednosměrnost funkce, odolnost vůči modifikaci a kolizi.

Jednosměrností se myslí nemožnost zjistit původní zprávu ze znalosti haše. Odolnost vůči modifikaci si můžeme představit tak, že změnou hašované zprávy Z se musí změnit i výstup hašovací funkce h. A odolnost proti kolizi znamená, že není možné najít dvě zprávy, jejichž výsledný haš by byl stejný.

V důsledku těchto požadavků se například změna jednoho bitu vstupní zprávy projeví nepredikovatelnou změnou na výstupu, která změní v průměru polovinu bitů haše. Délka výstupního haše je závislá na využitém hašovacím algoritmu. MD5 používá haše dlouhý 128b, SHA-1 160b a SHA-2 224, 256, 384, 512b. Na algoritmy MD5 a SHA-1 již byly publikovány možné útoky. Jejich bezpečnost je tedy značně omezena.

V roce 2012 byl standardizován organizací NIST nový algoritmus SHA-3 (Keccak). Byl vybrán i s ohledem na to, aby se algoritmus zcela lišil od SHA-2. Takže i při případném prolomení jednoho z algoritmů by měl druhý zůstat bezpečný. Výsledný haš SHA-3 může mít délky 224/256/384/512b.

Haš je možné využít pro ukládání hesel. Uživatel se přihlásí pomocí svého hesla z kterého je vypočítán haš a porovnán s hodnotou v databázi. Pokud jsou uloženy pouze haše hesel, znemožní to útočníkovi, který by získal data z databáze, odhalení hesel.

Další možné využití je autentizace zpráv. Spolu s přenášenou zprávou je odeslán i vypočítaný haš. Příjemce vypočítá haš z přijaté zprávy a porovná ho s přijatým hašem. Pokud si jsou si haše rovny, můžeme předpokládat, že zpráva nebyla při přenosu nijak modifikována.

Poslední využití hašovaní je při digitálním podepisování. Protože šifrování asymetrickými klíči je poměrně výpočetně náročné, je podpis aplikován pouze na haš podepisované zprávy.

2.2. SYMETRICKÉ KRYPTOSYSTÉMY

Symetrické kryptosystémy jsou využívány pro přeměnu přenášené zprávy Z na nesrozumitelnou posloupnost bitů (kryptogram C) a naopak. Pro šifrování a dešifrování je využit jeden společný klíč pro obě komunikující strany. Jako největší výhodu symetrických kryptosystému můžeme považovat jejich velkou rychlost šifrování a dešifrování. Naopak nevýhodou je distribuce klíčů mezi komunikující strany.

Symetrické kryptosystémy můžeme rozdělit do dvou hlavních skupin. Kryptosystémy mohou být proudové nebo blokové. *Proudové šifry* šifrují zprávu po jednotlivých bitech. Na každý bit zprávy je aplikována pomoci funkce XOR bit s pseudonáhodnou posloupností s_i. Pro

tvorbu posloupnosti s_i se využívá pseudonáhodného generátoru, kterému jako vstup slouží společný klíč. Proudové šifry nejsou tak často využívány jako šifry blokové. Pokud bychom si, ale měli uvést jednoho zástupce, pak jím bude šifra RC4. Obvykle využívá délku klíče mezi 40 až 256 bity.

Blokové šifry nešifrují zprávu po jednotlivých bitech, ale po blocích. Zpráva je rozdělena do stejně velkých částí (64, 128, 192 nebo 256b) a ty potom šifrují pomocí klíče. K šifrování existuje mnoho algoritmů, které využívají jiné metody tvorby kryptogramu. Jedním z nich je Feistelovo schéma. Jeho základním principem je rozdělení bloku na levou a pravou polovinu. Ty jsou pak vzájemně promíchávány s klíčem a mezi sebou. Toto schéma využívají šifry DES a 3DES. Modernější systémy AES využívají nahrazování a obměnu pořadí bitů k šifrování. Konkrétně je využito SP sítě.

AES je v současnosti nejpoužívanější symetrická šifra. Využívá bloky dlouhé 128 bitů a k šifrování využívá klíče o délce 128/192/256 bitů. Dalšími zástupci blokových šifer jsou algoritmy IDEA, DES a jeho vylepšení 3DES. Nevýhodou algoritmu DES je krátký klíč (56b), což je částečně řešeno 3DES. Ten provede dvě šifrování a jedno dešifrování, čímž je délka klíče prodloužena na 112b. Bohužel proces zašifrování zprávy trvá poměrně dlouho.

Existuje několik módů v kterých mohou blokové šifry pracovat. Nejjednodušším módem šifrování je mód Electronic codebook (ECB). Každý blok je samostatně šifrován pomocí sdíleného klíče. Tato varianta není příliš bezpečná, protože lze brzy odvodit šifrovací klíč. Nejčastěji využívaným módem je Cipher-block chaining (CBC). Tento mód využívá XORování nešifrovaného textu s kryptogramem předchozího bloku. Protože první blok nemá žádný předcházející kryptogram, musí zde být využit inicializační vektor (IV). IV většinou bývá náhodné číslo sdílené oběma stranami. Existují ještě módy PCBC, CFB a CTR. Rozdílem oproti CBC je pouze to, která část předchozí zprávy je aplikována na šifrování zprávy následující. V následku toho blokové šifrování šíří chyby. Pokud v některém bloku dojde k chybě, tato chyba se šíří na všechny následující bloky.

2.3. ASYMETRICKÁ KRYPTOGRAFIE

Na rozdíl od symetrické kryptografie využívá asymetrická kryptografie dvojici klíčů. První klíč je soukromý a nesmí být nikdy zveřejněn. Druhý klíč je veřejný a předkládá se protistraně. Z veřejného klíče musí být nemožné odvodit v rozumném čase klíč soukromý, k tomu slouží těžko řešitelné matematické problémy. Asymetrické kryptosystémy jsou v porovnání se symetrickými velmi pomalé. Proto se používají pouze k šifrování krátkých zpráv nebo k podepisování.

Při šifrování si autor zprávy musí opatřit veřejný klíč adresát pomocí, kterého zprávu zašifruje C = E(Z, VK). Jediným, kdo takovouto zprávu může dešifrovat, je majitel soukromého klíče. Adresát kryptogram dešifruje Z = D(Z, VK).

Digitální podpisy slouží k nepopiratelnosti autorství zprávy. Autor digitálního podpisu zašifruje zprávu pomocí svého soukromého klíče C = E(Z,SK). Adresát si důvěryhodným způsobem opatří veřejný klíč autora. Pomocí toho může zprávu dešifrovat Z = D(C, VK). Adresát má ověřeno, že zprávu zašifroval vlastník SK. Protože podepisovat celou zprávu by bylo velmi zdlouhavé a výpočetně náročné, aplikuje se podpis pouze na haš vytvořený ze zprávy Z. Adresátovi se pošle zpráva Z s připojeným podpisem. Příjemce zprávy vypočte haš zprávy Z a porovná ho s dešifrovaným obsahem podpisu. Takovýto postup využívá algoritmus RSA.

Algoritmus RSA je založen na problému faktorizace velkých čísel. VK bývá často volen jako čísla 3, 5, 17, 257 nebo 65537. Tyto čísla se využívají proto, že v binárním vyjádření obsahují pouze 2 jedničky. Což zkrátí dobu šifrování případně dešifrování.

K dohodnutí tajného klíče přes veřejně dostupný komunikační kanál slouží protokol *Diffie-Hellman*. Pokud si strany vymění dostatečně velké prvočíslo p, generátor grupy g a vypočítanou hodnotu (X nebo Y). Pomocí těchto hodnot komunikující strany vytvoří sdílený klíč. Nevýhodou je , že v základní podobě chybí autentizace entit. Tím se stává protokol nachýlným k man in the middle útoku.

Existuje i varianta protokolu *Diffie-Hellman s využitím eliptických křivek*. Princip je stejný jako v předchozím případě. Pouze operace mocnění je nahrazena násobením bodu eliptické křivky. Výsledným sdíleným tajemstvím je bod na křivce. Ten se musí upravit, aby z něho vznikl šifrovací klíč. Hlavní výhodou je využití menších klíčů, při zachování stejné bezpečnosti. Zmenšením šifrovacích klíčů se zvýší rychlost šifrování.

2.4. CERTIFIKÁTY

Certifikáty slouží k ověření identity vlastníka veřejného klíče. Certifikáty poskytuje certifikační autorita, které všichni důvěřují a všichni znají její veřejný klíč. Certifikační autorita vydá certifikát žadateli poté, co ji prokáže svoji identitu I. Certifikát je digitálně podepsaná zpráva pomocí soukromého klíče autority, prokazující, že identita I disponuje veřejným klíčem VK.

Celý systém ověření identity funguje na principu přenesení důvěry. Je možné důvěřovat neznámému certifikátu, který je podepsaný důvěryhodnou certifikační autoritou.

3. Řízení přístupu v počítačových sítích

Počítačové sítě poskytují celou řadu aktiv, ke kterým potřebuje poskytovatel služeb řídit přístup. Řízení přístupu se využívá buď z důvodu důvěrnosti poskytovaných služeb, nebo kvůli vynucení poplatku za využívanou službu.

V počítačových sítích se k řízení přístupu využívají dvě základní skupiny protokolů. První provádí pouze autentizaci a autorizaci (AA protokoly). Skupina druhá přidává k předchozím funkcím ještě sledování využívání síťových zdrojů. Těmto protokolům se říká AAA (z anglického authentication, authorization and accounting).

Řízení přístupu bylo řešeno již v první paketové síti ARPANET. V roce 1984 k tomu byl poprvé použit protokol TACACS. Postupem času přicházeli i další protokoly. Protokol představený roku 1993 s menšími obměnami využívaný až dodnes je KERBEROS. V roce 1997 byl standardizovaný dodnes velmi populární protokol RADIUS. V témže roce firma CISCO inovovala TACACS a tím vytvořila protokol TACACS+. Jelikož všechny tyto protokoly trpěli nedostatky, byl vytvořen DIAMETR. Což je rozšířený následovník protokolu RADIUS.

3.1. AA protokoly

AA protokoly zajišťují centralizovanou autentizaci a autorizaci žadatelů. Tím, že je centralizovaná umožňuje jednoduchou správu uživatelů a jejich přístupových práv. Tyto protokoly neobsahují účtování především z důvodu, že byly vydány v době, kdy centralizované účtování přístupu k službám nebylo vyžadováno.

Z výše jmenovaných protokolů je jediným stále využívaným zástupcem AA protokolů *Kerberos*. Druhým je protokol TACACS. Ten už ale ve většině systémů ustoupil novějším protokolům.

Úkolem AA protokolů je provést autentizaci, její výsledek přenést k autoritě. Autorita rozhodne o právech žadatele a toto rozhodnutí je znovu pomocí AA protokolu přeneseno ke kontroléru, ten pak umožní/odepře přístup ke službám. Konkrétní implementaci si popíšeme na protokolu Kerberos.

3.1.1. Kerberos

Protokol vyvinutý na americké univerzitě MIT. Kvůli využívání blokové šifry DES, byl v počátcích zakázán export protokolu ze Spojených států. Ve verzi V5 specifikované v RFC 4120, je už DES nahrazeno novějším, volně dostupným a mnohem bezpečnějším algoritmem AES.

Protokol využívá čtyři základní entity. *Žadatele* snažícího se přistoupit k *aplikačnímu serveru*, *autentizační server* a tzv. *TGS server*. Protokol Kerberos standardně využívá pro přenos dat protokoly transportní vrstvy TCP a UDP na portu 88.

Žadatel nejdříve vygeneruje haš vytvořený z hesla, čímž si vytvoří klíč K_1. Komunikace mezi entitami je započata odesláním uživatelského jména žadatele (zpráva A) autentizačnímu serveru. Ten prohledá databázi, v které hledá shodu uživatelský jmen. Spolu s uživatelským jménem je uložen i heslo, díky kterému může autentizační server vytvořit stejný klíč 1, jako žadatel.

Autentizační server odešle žadateli dvě zprávy. Zpráva B obsahuje klíč pro komunikaci klient-TGS server (K_2), zašifrovanou klíčem 1. To si můžeme představit jako $B = E(K_2, K_1)$, kde E znamená šifrování algoritmem AES. Druhou zprávou je zpráva C nesoucí

uživatelské jméno klienta, síťovou adresu, životnost tiketu a klíč K_2. To vše zašifrované tajným klíčem TGS serveru (klíč K_3) C= E(ID|| adresa|| doba platnosti|| K_2, K_3).

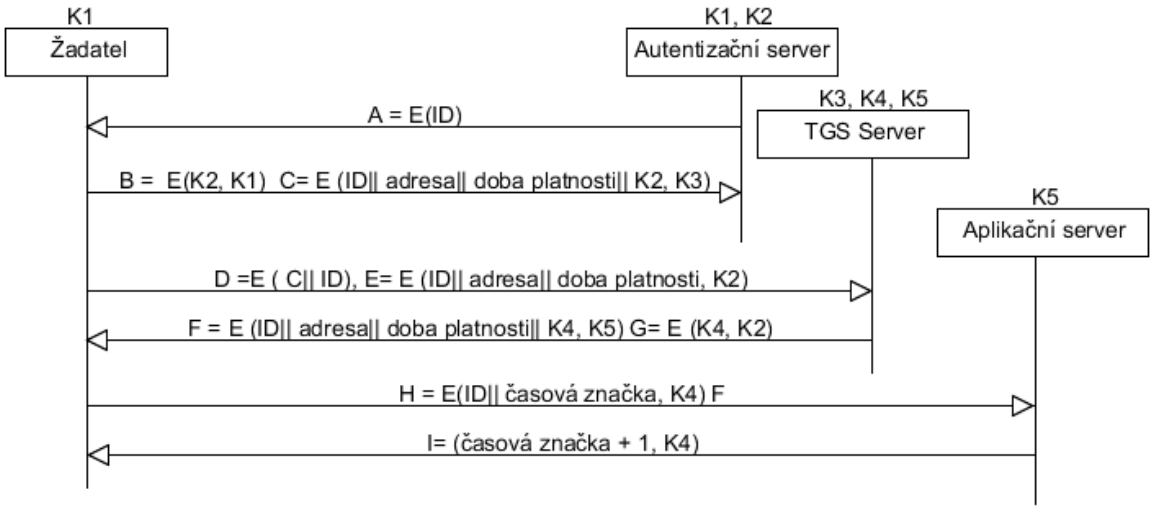
Když klient přijme zprávu B, může ji dešifrovat pomocí svého klíče K_1. Dešifrováním získá klíč K_2= D(B, K_1), kde D znamená dešifrování. Je patrné, že uživatel je schopný zprávu dešifrovat pouze pokud zadal správné heslo. Ziskem klíče K_2 a zprávy C umožní žadateli přejít ke komunikaci s TGS serverem.

Žadatel odešle TGS serveru zprávu D a E. Zpráva D se skládá ze zprávy C a ID služby, k níž žadatel chce získat přístup. Zpráva E obsahuje uživatelské jméno klienta a časovou známku zašifrovanou klíčem K_2.

Jelikož TGS server disponuje klíčem K_3 je schopný dešifrovat zprávu C obsaženou v D, čímž získá K_2 nutný pro dešifrování zprávy E. Tím, že TGS dešifruje obě zprávy, ověří identitu žadatele a může mu odpovědět zprávami F a G. F je složené z uživatelského jména, síťové adresy, doby platnosti a klíče sloužícího pro komunikaci klient – aplikační server (klíč K_4). To celé zašifrované klíčem aplikačního serveru (K_5). F = E(ID|| adresa|| doba platnosti|| K_4, K_5). Zpráva G nese klíč K_4 zašifrovaný klíčem K_2.

Klient je vlastníkem klíče K_2, díky kterému může dešifrovat zprávu G, z které získá K_4. Klíč K_4 využije pro zašifrování uživatelského jména a časové značky, čímž vznikne zpráva H = E(ID|| časová značka, K_4). Zprávu H spolu s dříve přijatou zprávou F odešle aplikačnímu serveru.

Aplikační server je vlastníkem klíče K_5 s jehož pomocí dešifruje zprávu F. Díky dešifrování F, se stane držitelem klíče K_4, který využije pro dešifrování zprávy H. Obdrženou časovou značku inkrementuje, zašifruje klíčem K_4 a odešle zpět žadateli. Žadatel si může ověřit, že je hodnota korektně inkrementována. Pokud ano, je jasné, že komunikuje s aplikačním serverem a může bezpečně využívat jeho služeb.



Obr. 3.1 Průběh protokolu Kerberos

Na příkladu je jasně patrné jak výsledek jedné autentizace slouží jako dokazovací faktor pro autentizaci vůči jiné entitě. Celkově lze říct, že se jedná o decentralizovaný hierarchický protokol skládající se ze tří fází.

3.2.AAA PROTOKOLY

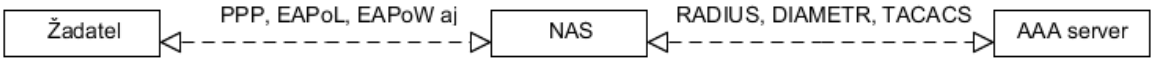
Protokoly jsou určeny pro přenos autentizačních, autorizačních a účtovacích zpráv sítí. Komunikace těmito protokoly probíhá mezi dvěma entitami. *Klientem* a *serverem*. Jako server vystupuje v této komunikaci *AAA server*, který slouží k centralizovanému zajištěni autentizace, autorizace a účtování služeb. Jako klient, ale nevystupuje žadatel o službu, jak by se mohlo na první pohled zdát. Roli klienta vykonává *hraniční přístupový bod* (NAS), skrze který chce žadatel přistupovat. NAS v závěru komunikace působí jako kontrolér, který žadateli umožní přistupovat do sítě.

Rozdíly mezi jednotlivými protokoly jsou pouze minimální. Pokud bychom měli jmenovat největší rozdíl, pak jsou to využité protokoly transportní vrstvy. Jinak princip komunikace je téměř totožný.

Způsob jakým bude prováděna komunikace mezi NAS a žadatelem, není AAA protokoly řešena. K propojení je běžně využíváno klasických technologií ethernet, wi-fi nebo PPP.

3.2.1. AUTENTIZACE AAA PROTOKOLŮ

Nad těmito protokoly je pak prováděna autentizace s využitím protokolů *PAP*, *CHAP* nebo *EAP*. Při autentizaci plní NAS pouze roli prostředníka, jenž překládá autentizační protokol, který využívá žadatel, do protokolu AAA serveru. Níže popsané protokoly jsou, kromě protokolu EAP, určeny pro využívání na PPP



Obr. 3.2 Zapouzdření autentizačních mechanizmů

PAP

PAP (Password Authentication Protocol) je autentizační protokol neposkytující téměř žádné zabezpečení. Data odeslaná PAP jsou odeslaná v otevřeném ASCII formátu. Tím pádem je jakýkoliv útočník, který odposlechne komunikaci, může zachytit. Při případném útoku stačí jen odeslat zachycená data

CHAP

Protokol pro autentizaci *CHAP* neodesílá heslo v otevřené podobě, tak jak PAP ale využívá hašování. Strana vystupující jako autentizátor, vyšle výzvu (Challange) obsahující náhodné číslo a ID sloužící jako počítadlo přenášených zpráv. Žadatel, pomocí přijatého náhodného čísla, ID a sdíleného tajemství (hesla), vytvoří haš h=H(ID|| náhodné číslo|| heslo), který odešle autentizátorovi. Autentizátor si spočítá vlastní haš. Ten může porovnat s hašem přijatým a rozhodnout o výsledku autentizace. Jelikož běžně je využíván hašovací algoritmus MD5, není poskytnutá bezpečnost o moc větší než ta při využití protokolem PAP.

MS-CHAP

MS-CHAP je verze autentizačního protokolu vytvořená firmou Microsoft. Existuje ve dvou verzích MS-CHAPv1 a MS-CHAPv2. Pokud je v současnosti některý z těchto protokolů využíván, pak je to MS-CHAPv2. Bohužel, ale i ten má velké bezpečnostní nedostatky.

K šifrování přenášených údajů využívá šifry DES. Pro šifrování využívá trojice 7 bitových klíčů vytvořených rozdělením haše hesla, které má však pouze 16 bitů. Kvůli tomu musí být 5 bitů třetího klíče doplněno nulami. Kvůli tomu je prolomení třetího klíče otázkou

chvíle. Pro zjištění prvních dvou klíčů je zapotřebí maximálně 23 hodin. Průměrně však okolo 12 hodin.

Díky znalosti všech tří klíčů zná útočník haš hesla, a pokud jednou zaznamenal průběh komunikace, má všechny potřebná data pro útok.

EAP

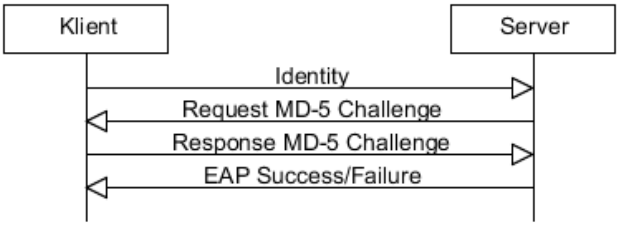
Dále se k autentizaci využívá protokolu *EAP*, popisující formát zpráv. EAP zprávy musí být zapouzdřeny do nějakého jiného protokolu, který jim umožní přenos sítí. Pro zapouzdření od klienta k NAS se využívá zapouzdření do protokolu IEEE 802.1X. Ten umožňuje přenos EAP zpráv skrze ethernet, wi-fi i optické sítě. NAS vyjme EAP zprávy z 802.1X rámce a vloží je do AAA protokolu. Komunikace probíhá formou klient-server. V případě využití u AAA protokolů vystupuje v roli klienta žadatel služby a v roli serveru rozhoduje AAA server.

Protokoly EAP tvoří celou rodinu autentizačních protokolů, kde každý protokol provádí autentizaci jinak. Existují takové, které jsou založeny na znalosti hesla (např. EAP-MD5), vlastnictví předmětu (např. EAP-GTC), vlastnictví soukromého klíče (např. EAP-TLS) nebo kombinace předchozích (např. EAP-IKEv2). Celkově existuje okolo čtyřech desítek EAP protokolů. Některé metody kromě autentizace umožní sjednání tajného klíče, kterým může být šifrována následující komunikace.

V následujících podkapitolách si popíšeme některé z často využívaných EAP autentizačních metod.

EAP-MD5

EAP-MD5 je autentizační metoda přenášející heslo v zašifrované podobě pomocí algoritmu MD5. Algoritmus MD5 je velmi náchylný na slovníkové útoky a proto nepředstavuje dostatečnou bezpečnost. Dalším nedostatkem, kterým EAP-MD5 trpí je pouze jednostranná autentizace. Dojde k ověření identity uživatele, serveru však nikoliv. Tím se metoda stává náchylná k útoku typu Man in the middle. Při využití s AAA protokoly můžeme považovat za nedostatek i to, že neumožní sjednat sdílené klíče, kterými byla následná komunikace šifrována.

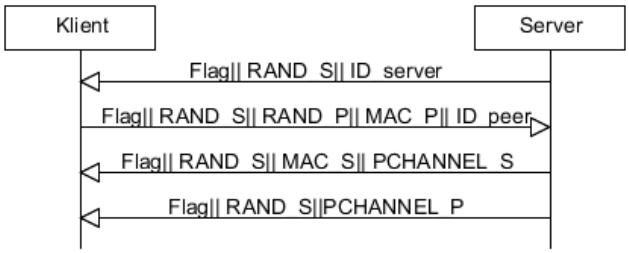


Obr. 3.3 Průběh autentizace EAP-MD5

EAP PSK

EAP PSK slouží k autentizaci komunikujících stran a sjednání klíčů využívaných k symetrickému šifrování následujících dat. EAP PSK je definováno v [2]. Pomocí předem sdíleného hesla (*PSK*) je možné vytvořit dvojici klíčů. Klíč *AK* slouží pro autentizaci obou klientů a klíč *KDK* (key derivation key) je použit pro vytvoření klíče *TEK* (transient EAP key) používaného pro šifrování spojení. Spojení je šifrováno algoritmem AES využívajícím klíče o velikosti 128 bitů.

EAP PSK je určen pro komunikaci na nezabezpečeném kanále jako je 802.11. V případě, že není prozrazeno sdílené tajemství, může být tato metoda autentizace považována za bezpečnou.



Obr. 3.4 Autentizace využívající EAP PSK

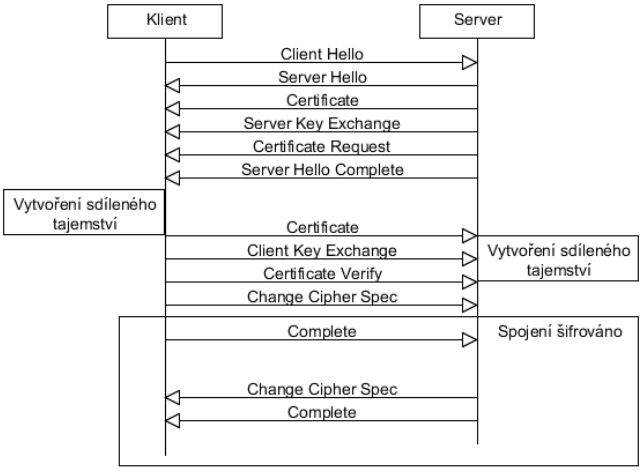
EAP-TLS

EAP-TLS definované v [3] je jednou z nejbezpečnějších způsobů autentizace a sjednání klíčů, jaká vůbec může být. Problém však nastává v nutnosti vlastnictví asymetrického páru klíčů klientem.

Autentizaci započne klient nabídkou jím podporovaných šifrovacích sad ve zprávě ClientHello. Z nabízených šifrovacích sad (obsahujících vybraný algoritmus k vytvoření sdíleného klíče, algoritmus, kterým budou komunikující strany autentizovány, využitý algoritmus pro šifrování symetrickým klíčem a hašovací algoritmus pro ověření integrity přijatých dat) si server jednu vybere. Výběr odešle ve zprávě *ServerHello*. Klientovi server také odešle svůj certifikát ve zprávě *Server Certificate,* dokazující jeho identitu, zprávu Server Key Exchange jež nese informace nutná pro vytvoření sdíleného tajemství (např. Diffie-Hellman veřejný klíč), žádost o certifikát klienta zprávou *Certificate Request* a vše ukončí zprávou *Server Hello Complete*.

Po obdržení zprávy Server Hello Complete, začne vysílat zprávy klient. Nejdříve serveru zašle svůj certifikát. Další odesílanou zprávou je *Client Key Exchange* nesoucí informace, na základě kterých může server vytvořit sdílený klíč. Pokud byl certifikát serveru platný, odešle klient zprávu *Certificate Verify*. Poslední nešifrovanou zprávou je *Change Cipher Spec*. Tato zpráva říká, že veškerá následná komunikace bude šifrována na začátku domluveným algoritmem. Jako klíč k šifrování je využito sdíleného klíče. K ověření, že i server si vytvořil sdílené tajemství, odesílá klient již zašifrovanou zprávu *Complete*.

Server si díky zprávě Client Key Exchange vytvoří sdílené tajemství, díky kterému dešifruje zprávu Complete. Tím si ověřil, že klient disponuje sdíleným klíčem. A jako poslední krok odešle zašifrovanou zprávu Complete zpět.



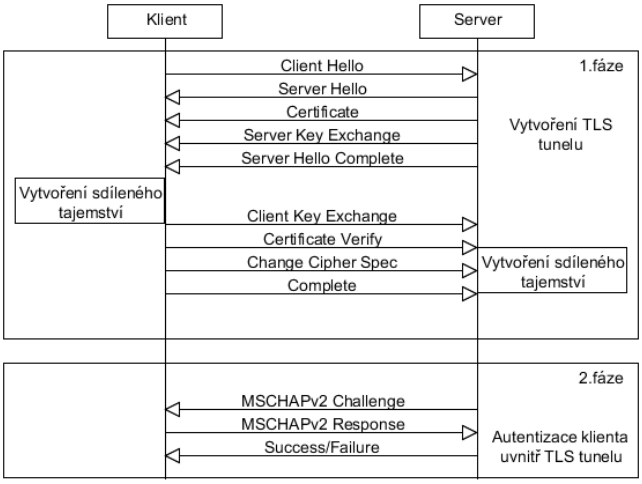
Obr. 3.5 Autentizace EAP-TLS

PEAP

Protected EAP skládá autentizaci ze dvou fází. V první je vytvořen *TLS tunel* stejně jako tomu je v protokolu EAP-TLS. Až na jeden podstatný rozdíl, není vyžadován certifikát na straně klienta. Pomocí certifikátu je autentizován pouze server. Identita klienta je ověřena až po vytvoření bezpečného spojení.

Klient může být autentizován dvěma způsoby. Buď pomocí metody *MS-CHAPv2* nebo využitím autentizačního tokenu.

Tento protokol představuje bezpečnostní kompromis. Poskytuje relativně vysokou bezpečnost a zároveň nenutí klienta k tomu, aby vlastnil dvojici asymetrických klíčů a certifikát.



Obr. 3.6 Autentizace metodou PEAP

3.2.2. RADIUS

Remote Authentication Dial In User Service (*RADIUS*) je protokol využívající transportního protokolu UDP na portu 1812 pro autentizaci a 1813 pro účtování. Ve své základní podobě podporoval autentizaci pouze za použití PAP a CHAP, což ale z důvodů malé bezpečnosti

nedostačovalo. V RFC 3579 byl protokol rozšířen pro využití autentizace EAP, tím že je možné zapouzdřit EAP zprávy přímo do RADIUS protokolu.

RADIUS paket se skládá z hlavičky, která obsahuje druh zprávy (Request, Response apod.), délku celého paketu, *autentikátoru* a identifikátoru paketu. Za hlavičkou následuje obsah samotného RADIUS protokolu. Obsah je složen z *AVP* (Atribute Value Pairs). Což je dvojice složená z typu zprávy a její hodnoty (value). Jako příklad si uvedeme AVP typu 79. To je hodnota přiřazená EAP zprávám. V poli value se tedy bude nacházet EAP zpráva. Zpráva RADIUS protokolu může přenášet neomezené množství AVP. Kvůli 8 bitovém poli určujícímu druh AVP je možné definovat maximálně 255 typů AVP.

Jelikož ne všechny druhy AVP jsou v základním protokolu využity, je možné protokol dále rozšiřovat a vytvářet si nové AVP páry, které umožní dodatečné funkce protokolu.

Obrovskou nevýhodou protokolu je přenášení zpráv v otevřeném formátu. Jediné co RADIUS v základní podobě šifruje, je uživatelské heslo. Heslo je zahešováno spolu s náhodnou hodnotou uvedenou v hlavičce (autentikátor). Všechny ostatní zprávy jako uživatelské jméno, výsledek autorizace apod. jsou pro útočníka, který má možnost odposlechnout komunikaci mezi NAS a RADIUS serverem , volně dostupné.

3.2.1. TACACS+

Terminal access controller access-control plus (*TACACS+*) je proprietární protokol vyvinutý firmou CISCO. Vyhází z původního protokolu TACACS s kterým, ale není zpětně kompatibilní. TACACS+ ke komunikaci využívá spojově orientovaný protokol TCP na portu 49. Díky tomu na rozdíl od RADIUSu může například detekovat ztrátu paketu a přenos zopakovat. Další obrovskou výhodou je šifrování všech přenášených zpráv.

Naopak nevýhodou, kterou TACACS+ trpí je nemožnost autentizace za využití EAP. Pro autentizaci dokáže využít PAP, CHAP a případně MS-CHAP.

3.2.2. DIAMTER

Protokol *DIAMETR* je následovníkem protokolu RADIUS. Liší se především v tom, jakým protokolem je přenášen. Místo UDP využívá spolehlivý transportní protokol *TCP* nebo *SCTP* na portu 3868. Dalším vylepšením je rozšíření hodnoty pro definování AVP typů z 8 bitové hodnoty na 32 bitovou. Tím se značně rozšíří množství využitelných AVP. Typy AVP 0-255 jsou rezervované pro zpětnou kompatibilitu s protokolem RADIUS, tak aby informace přenášené protokolem RADIUS mohli být zapouzdřeny do DIAMETRu.

Další značnou výhodou je možnost zabezpečení protokolu na transportní vrstvě pomocí TLS nebo *IPSEC*. Pokud tedy komunikace probíhá v bezpečném TLS tunelu nebo je-li paket šifrován IPSec protokolem, stává se pro útočníka téměř nemožné odposlechnout DIAMETR zprávy.

Z pohledu autentizace má stejné možnosti jako protokol RADIUS. Může tedy využívat metod PAP, CHAP, MS-CHAP a EAP.

4. BEZKONTAKTNÍ IDENTIFIKACE

Jedna z možností používaných pro řízení přístupu je založena na vlastnictví předmětu, který je schopen bezdrátové komunikace. Nejčastěji se využívají karty, klíčenky, případně i mobilní telefon. Bezkontaktní technologie mají velké množství výhod. Pro komunikaci mezi čtečkou a identifikačním předmětem není potřeba jejich kontakt ani přímá viditelnost, což velice usnadňuje jejich použití. Aby byla možná jednoznačná identifikace, je každý takto komunikující předmět vybaven svým unikátním identifikačním číslem (*UID*). Kontrolér potom může na základě UID rozhodnout o udělení přístupu.

Systém pro bezkontaktní identifikaci osob je používán v široké škále odvětví. Jako příklady můžeme uvést přístup do budov, elektronické občanské průkazy, platební karty atd. Pro takovouto identifikaci je používána technologie označována jako *RFID* a její nástavba *NFC*.

4.1. RFID

Radio frequency identification je způsob identifikace využívající komunikace mezi čtečkou a identifikačním prvkem, RFID štítkem. Princip je jednoduchý. Čtečka vysílá rádiový signál a štítek na něj odpovídá odesláním svého UID, případně obsahem své datové paměti.

Technologie RFID byla patentována v 80. letech 20. století. Hlavním iniciátorem vývoje byla společnost Wal – Mart, která hledala náhradu za čárové kódy. Z počátku bránila rozšíření větší cena, ale s postupem času a rozmachem moderní technologii se rozšířilo i využití RFID. V současnosti je užíváno nejenom pro svůj prvotní účel identifikace zboží, ale i pro identifikaci zvířat, v automobilovém průmyslu, pivovarnictví a zdravotnictví.

Existují dva druhy štítků. *Pasivní štítky* jsou takové, které neobsahují vlastní zdroj elektrické energie. Pro napájení je využíváno elektromagnetických vln, periodicky vysílaných čtečkou. Pokud se pak štítek dostatečně přiblíží k čtečce, je na jeho anténě indukováno napětí. Což vyvolává střídavý elektrický proud, jenž je usměrněn a nabíjí kondenzátor umístěný ve štítku. Poté, co napětí na kondenzátoru dosáhne minimální potřebné úrovně, jsou spuštěny řídící obvody uvnitř štítku a ten začne odesílat odpověď. Odpověď je většinou realizována pomocí dvoustavové amplitudové modulace.

Štítky aktivní na rozdíl od pasivních obsahují vlastní zdroj napájení. Tudíž vzdálenost, na kterou mohou komunikovat je mnohem větší. Kromě identifikace mohou poskytovat i další funkce jako je lokalizace nebo měření teploty. Nevýhodou je větší cena a jejich složitost.

Standardní nosná frekvence užívaná pro přenos je 125 kHz a 13,56 MHz. Okrajově se používají i frekvence 134 kHz a v Evropě 868 MHz. Obecně platí, že čím větší frekvence tím je rychlejší přenos a větší vzdálenost na kterou je možno komunikovat.

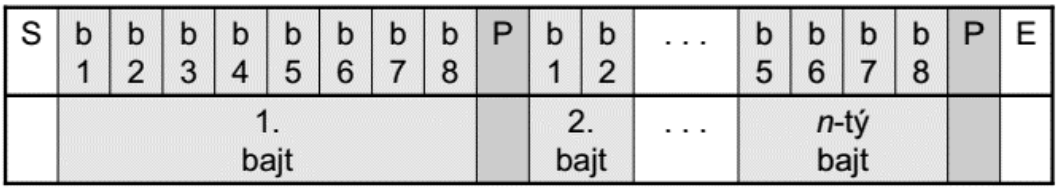
Pro osobní identifikaci je nejčastěji využíván standard identifikačních karet *ISO/IEC 14443*. Byl vytvořen v roce 2001 a definuje dvě mírně odlišné verze. Typ A a typ B, ty se liší způsobem použité modulace a rozdílnými anti-kolizními mechanismy. Skládá se ze čtyř částí, kde první popisuje fyzické charakteristiky, druhá standardizuje frekvenční charakteristiky vysílání, třetí inicializaci a autentizaci a část čtvrtá popisuje přenosový protokol. Využívá pracovní frekvence 13,56 MHz. Maximální dosah je okolo 10 cm při přenosové rychlosti od 106 kb/s do 848 kb/s. Velikost pamětí má značné rozpětí od 64 B až po 64 kB. V oblastech řízení přístupu lze využívat řadu od karet čistě paměťových až po bezpečné mikropočítačové, které mohou obsahovat i kryptografické procesory.

4.1.1. RFID KOMUNIKACE

Při tvorbě standardů ISO/IEC 14443 se výrobci nemohli shodnout na způsobu komunikace mezi kartou a čtečkou a proto vznikli dva odlišné principy komunikace. ISO/IEC 14443 A pro komunikaci od čtečky směrem ke kartě využívá ASK (ampitude shifting key) se 100% hloubkou modulace. Pro zakódování dat se využívá modifikovaný Millerův kód. Millerův kód pro logickou 1 převezme předchozí stav a v polovině intervalu symbolu jej změní. A pro logickou 0 stav ponechá, nachází-li se za logickou 1. Pokud následuje po logické 0, stav změní. Jak je vidět na obrázku. Modifikovaný Milerův kód se využívá proto, aby zajistil napájení kartě. Jeho výhodou je převažující stav H. Rychlost přenosu je 106 kb/s. Přenos od karty zpět ke čtečce je pak realizován pomocí On-Off keying modulace subnosné 848 kHz kódem Manchester a následná zátěžová modulace na nosnou frekvenci 13,56 MHz [4 stránky 106-111] a [5].

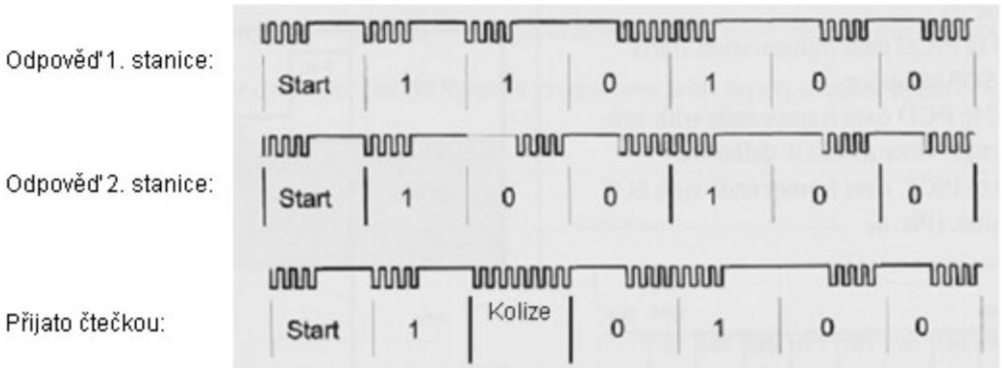
Standard ISO/IEC 14443 B je založen na komunikaci, která používá ASK s 10% hloubkou modulace. Data jsou šifrována pomocí Non return to zero (NRZ) kódu. Rychlost přenosu je stejná jako u standardu A, tedy 106 kb/s. Komunikace v opačném směru je realizována pomocí dvoustavové amplitudové modulace na subnosné frekvenci 847 kHz opět NRZ kódem. Takovýto signál je následně amplitudově modulován na nosnou frekvenci 13,56 MHz.

Data mezi čtečkou a kartou jsou přenášeny po bajtech v datových rámcích. Počet bajtů v rámci není pevně stanoven. Rámec je vždy zahájen *start bitem* rámce S. Následovaným datovými bity b1 až b8. Pro kontrolu je vložen *paritní bit P* daného bajtu. Vytvořený za pomocí operace XOR mezi b1 až b8. Rámec je ukončen *koncovým bitem E*.



Obr. 4.1 Datový rámec RFID

Komunikace je poloduplexní, buď vysílá čtecí zařízení a karta naslouchá nebo opačně. Je možno uskutečnit komunikaci pouze bod-bod. A proto, aby nedocházelo ke kolizím, je ve standardu popsána anti kolizní smyčka. Algoritmus pracuje tak, že terminál vyšle povel *SELECT*. Výzva čtečky je datový rámec, který obsahuje parametr *Prefix* (předpona), což jsou první bity UID hledané stanice. V první výzvě se vysílá Prefix = Ø, což je řetězec o nulové délce. Na každou výzvu odpovídají jen stanice, jejichž UID začíná požadovaným Prefixem. V případě kolize pak čtečka stanovuje Prefix jako první nekolidované bity, které rozšíří o bit "1". Své výzvy čtečka opakuje, dokud kolize nezmizí. Tímto algoritmem čtečka nalezne stanici S, která má ze všech stanic nejvyšší UID. S danou stanicí zahájí komunikaci a po jejím ukončení ji zašle příkaz *HALT*, který způsobí, že tato stanice na antikolizní protokol již nereaguje. Popsaný postup se postupně opakuje se všemi ostatními stanicemi.



Obr. 4.2 Detekce kolize

4.2. NFC

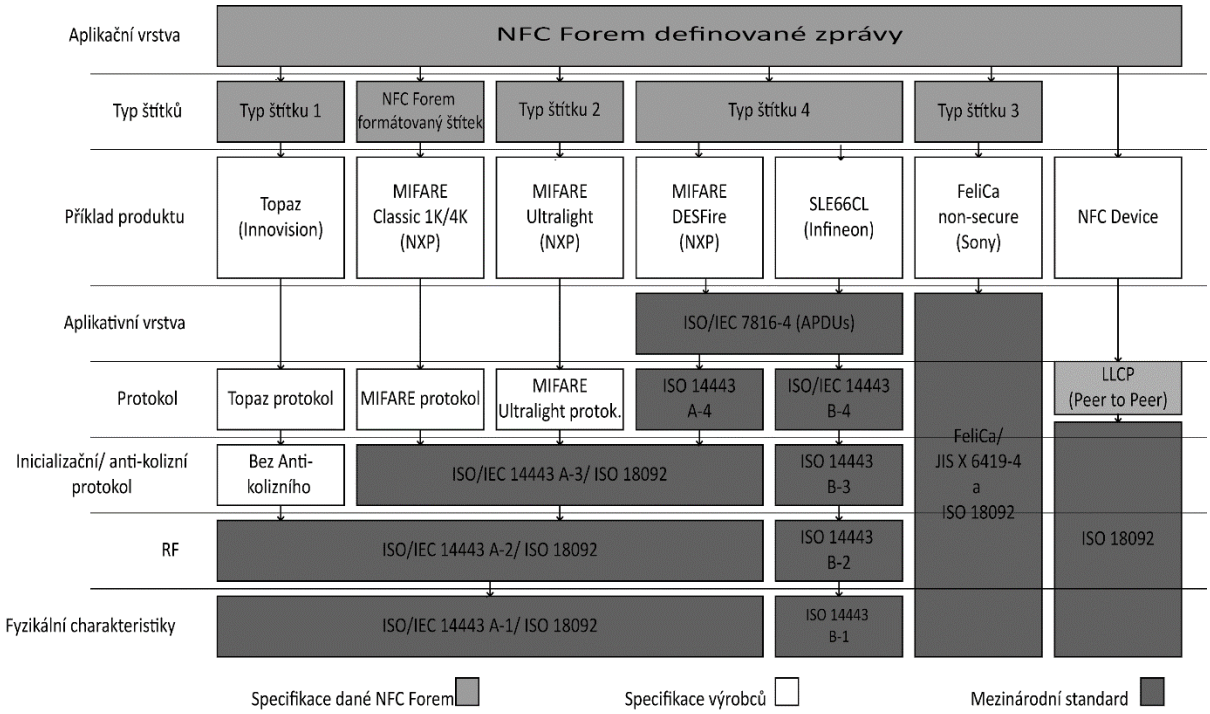
Near field communication zkráceně *NFC* je sada standardů sloužících k bezdrátové komunikaci na malou vzdálenost. Standardy pro NFC jsou definovány neziskovou organizací *NFC Forum* založenou v roce 2004 firmami Nokia, Philips a Sony. V současnosti se do NFC Fora zapojilo okolo 170 firem. NFC je založeno na standardu ISO/IEC 14443 a ISO 18092.

NFC je stejně jako RFID možné využít pro identifikaci osob. Někteří výrobci automobilů dávají možnost nastartovat automobil pomocí telefonu vybaveným NFC. Další použít je pro rychlé spárování zařízení při komunikaci Bluetooth, nebo pro rychlé přihlášení do Wi-Fi sítě. Uplatnění NFC nalezne také v platebních systémech a nákupu lístků na hromadnou dopravu nebo veřejné akce.

V NFC rozeznáváme dva druhy komunikujících prvků, které se od sebe liší tím, zda vytváří vlastní radiofrekvenční pole (RF) pole nebo zda zařízení získává energii z RF pole vytvořeného jiným přístrojem. Pokud je přístroj schopen generovat svoje vlastní RF pole je označován jako aktivní. Zařízení pasivní pro komunikaci využívá RF pole vysílaného prvkem aktivním. Komunikace mezi aktivním a pasivním prvkem již využívala RFID. NFC ale zavádí nový režim, ve kterém může být prováděna komunikace mezi dvěma aktivními prvky. Při komunikaci dvou aktivních prvků (peer-to-peer) vysílá svoje radiofrekvenční pole pouze právě vysílající strana. Vysílající strany se střídají podle toho, jak probíhá komunikace. Pro komunikaci v režimu peer-to-peer je definován protokol *LLCP* (Logical link control protocol), který přibližně odpovídá spojové vrstvě modelu OSI. LLCP řídí aktivaci, sledování a deaktivaci spojení, asynchronní komunikaci a multiplexování. LLCP také poskytuje jak spojově tak nespojově orientovaný přenos.

Je-li uskutečňovaná komunikace mezi aktivní čtečkou a pasivním štítkem mluvíme o režimu čtení – zápis. V tomto režimu aktivní prvek může číst data uložená ve štítku, případně do něj zapisovat, pokud to štítek umožňuje. NFC Forum definovalo čtyři druhy pasivních štítků. Liší se velikostí pamětí, možností uzamykání paměti a standardy na kterých jsou založeny. Tím, že je NFC v podstatě pouze nástavbou RFID existuje možnost zpětné kompatibility se štítky dodržujícími standard ISO/IEC 14443.

Souhrn standardů, na kterých jsou jednotlivé typy NFC komunikace postaveny je vidět na obrázku 2.3. U každého typu je uveden i příklad konkrétního produktu, který reprezentuje daný typ štítku. Zároveň je vidět i postavení NFC standardů na již existujících standardech využívaných RFID.



Obr. 4.3 Sada NFC standardů

NFC Forum popisuje ještě jeden režim komunikace, kterým je emulace čipových karet. V tomto režimu se aktivní zařízení chová jako zařízení pasivní a emuluje činnost karty podle specifikace ISO/IEC 14443. S takto emulovanou kartou může komunikovat NFC nebo RFID čtečka dle režimu čtení – zápis. Emulaci je možné využít například pro sloučení několika identifikačních karet do jednoho mobilního telefonu. Většinou je pro emulaci nutná přítomnost tzv. *Secure Elementu* (SE). Což je bezpečné výpočetní prostředí pro běh aplikací. Toto prostředí je zabezpečeno proti získání dat analýzou čipu případně jeho rozebráním. SE má obvykle formu čipu s integrovanou pamětí, procesorem, rozhraním pro komunikaci a dalšími volitelnými komponenty, například šifrovacím koprocesorem. SE může být integrován přímo v telefonu. Takových zařízení je na trhu zatím jen minimum. Proto se setkáváme se SE integrovaným v SIM kartách. Další možností jak integrovat SE do telefonu podporujícího NFC, je vložení paměťové karty obsahující SE.

Pokud například chcete používat NFC pro bezkontaktní platby je vám vydána, ve spolupráci telefonního operátora a banky, SIM karta disponující SE v kterém jsou uloženy údaje o vašem účtu.

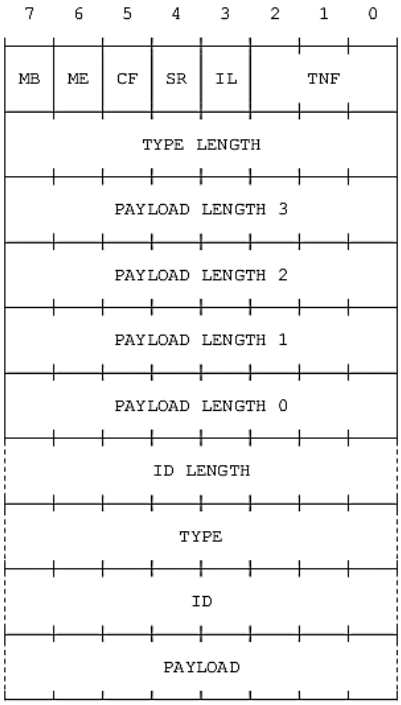
4.2.1. NFC KOMUNIKACE

Fyzické charakteristiky komunikace jsou specifikovány ve standardech *ISO/IEC 18092* (NFCIP-1), *ISO/IEC 21481* (NFCIP-2). Jsou to, ale standardy založené na výše zmíněné komunikaci dle ISO/IEC 14443. NFC a RFID dle ISO/IEC 14443 komunikují na stejné frekvenci (13,56 MHz), využívají stejné kódování (Millerovo modifikované nebo Manchaster). Pokud je přenosová rychlost vyšší než 106 kbit/s nahrazuje se 100% hloubka modulace, hloubkou 10%.

Novinkou v NFC jsou standardy NFCIP. NFCIP-1 popisuje druhy komunikace, které mohou být uskutečněny. Tak jak bylo popsáno výše, jsou to režimy čtení – zápis a peer to peer. Všechna zařízení podporující NFCIP-1 musí operovat na přenosových rychlostech 106,

212 nebo 424 kbit/s. Mezi těmito rychlostmi se zařízení mohou kdykoliv při přenosu přepínat. NFCIP-2 je rozšířením NFCIP-1 a umožňuje vybrat správný komunikační režim.

Co se týče komunikace samotné je v NFC zaveden datový formát zprávy *NDEF* (NFC Data Exchange Format). Je to formát specifikovaný NFC Forem, který je složen z jednoho nebo více NDEF záznamů (NDEF record). Jeho výhodou je možnost zapouzdření libovolných dat. Případně zapouzdření dat o neznámé velikosti.



Obr. 4.4 NDEF záznam

První bajt NDEF záznamu je hlavička složená z Message Begin (MB) a Message End (ME). Když je MB nebo ME 1, říká nám to, že tento záznam je první případně poslední ve zprávě. Dále v hlavičce najdeme Chunk Flag (CF), který indikuje, zda se jedná o rozdělená užitečná data. Po CF následuje Short Record (SR) ten je nastaven jako 1, pokud jsou přenášená data menší než 255 bitů. Tím pádem nemusí být ve zkrácené zprávě čtyři pole PAYLOAD LENGTH, udávající velikost přenášených dat. Ale stačí pouze jedno. Další důležitou součástí hlavičky je Type Name Format (TNF) specifikuje formát dat. Nejpoužívanější hodnotou jsou tzv. Well-Know Type, definované NFC Forem, které umožnují uvádět data ve zkrácené podobě. Což šetří paměť NFC štítku.

Po hlavičce následuje pole TYPE LENGTH udávající délku pole TYPE. ID LENGTH udává délku pole ID. Pole TYPE obsahuje popis přenášených dat a musí respektovat formát uložený v poli TNF. Předposlední údajem v NDEF záznamu je ID. ID je jednoznačný identifikátor zprávy. Tento identifikátor je uveden ale jen u prvního záznamu ve zprávě. U následujících záznamů je prázdný. A na závěr jsou přenášeny samotná data.

4.3. BEZKONTAKTNÍ AUTENTIZACE

Pro autentizaci se v přístupových systémech využívá určitý obsah paměti nebo v horším případě UID karty. UID totiž původně bylo navrženo pouze pro jedinečnou identifikaci karty, využívanou k vyhnutí se kolizím. V mnoha přístupových systémech však slouží přímo k identifikaci osoby, na základě které, je rozhodnuto o udělení přístupu. Bez jakékoliv

propracovanější autentizace. A jelikož komunikace není nijak šifrována, je snadné emulovat UID číslo a získat tak přístup bez velkých potíží [6]

Pokud tedy není bezpečnost zanedbána, jsou pro řízení přístupu využity autentizační protokoly. Těch existuje celá řada a liší se svou bezpečností, ale také požadavky na výbavu karty. U jednoduchých paměťových karet je ověřována pouze informace uložená v paměti. Výhodou takovýchto karet je nízká cena. Se vzrůstající cenou karet, lze pro autentizaci využívat různé formy šifrování komunikace, digitální podpisy a bezpečné ukládání dat. V praxi se karty dělí na:

- Paměťové karty
- Karty vybavené jednoduchou logikou
- Mikroprocesorové karty

Paměťové karty obsahují zpravidla paměť EEPROM a umožňují čtení a zápis informace do paměti. Karty s jednoduchou logikou poskytují možnosti autentizace napevno naprogramované do čipu během výroby. Příkladem takovýchto karet je NXP MIFARE Classic, dříve velmi využívaná pro elektronické jízdenky. Na vrcholu bezpečnosti stojí karty mikroprocesorové, které poskytují nejvyšší možné zabezpečení. Tyto karty jsou vybaveny větší pamětí a mohou být osazeny i kryptoprocesorem zajišťujícím náročné kryptografické výpočty.

Jako UID využívané autentizačními protokoly nemusí být použito unikátní identifikační číslo využívané antikolizním protokolem, ale může to být nějaká předem stanovená hodnota uchovávaná v paměti.

4.3.1. AUTENTIZACE PAMĚŤOVÝCH ČIPOVÝCH KARET

Autentizace paměťových čipových karet je založena na možnosti uzamknutí částí své paměti tak, aby přístup k ní byl podmíněn znalostí jisté tajné informace. Informace využité pro tuto kapitolu jsou čerpány z [7] a [8].

Hash Lock protokoly

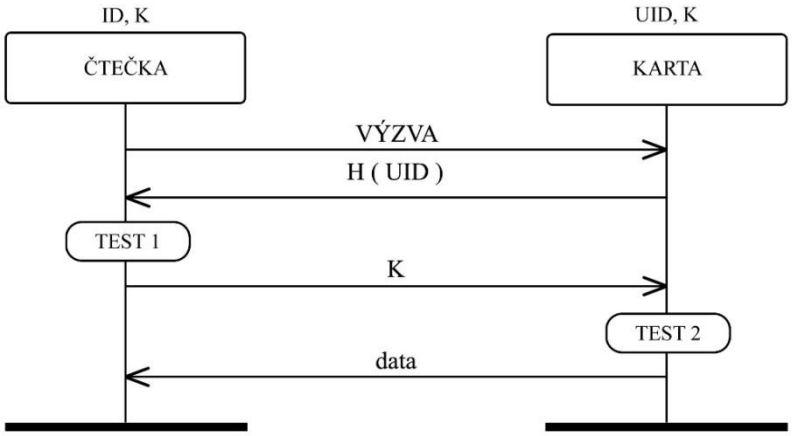
Hash Lock protokoly jsou jedny z nejstarších protokolů pro autentizaci. Tyto protokoly pro svou funkci vyžadují, aby čtečka znala haš UID karty a k ní přidružený tajný klíč. Díky tajnému klíči získá čtečka přístup k obsahu karty. Na základě kterého je schopna identifikovat kartu.

Hash Lock protokol

Protokol, na jehož základě je pak postaveno mnoho dalších protokolů pouze drobně se lišících. Byl vyvinut kolektivem Stephana Weise a byl určen především k tomu, aby neoprávněná čtečka nezískala informace z karty.

Protokol funguje tak, že čtečka nejdříve vyšle výzvu kartě, která na ní odpovídá haší svého UID. Čtečka prohledá databázi systému a najde-li daný haš, tak odešle tajný klíč karty. Karta může otestovat, zda se shoduje přijatý tajný klíč s klíčem uloženým v paměti. Pokud ano je dovršena vzájemná autentizace a karta může odeslat data uzamčená v paměti.

Hash Lock protokol je pouze takovým odrazovým můstkem pro další protokoly. Sám totiž příliš bezpečný není. Pokud útočník zachytí data přenášena v posledním kroku, je schopen kdykoliv tato data využít pro získání přístupu. Problém je i ve chvíli, kdy útočník odchytí jen tajný klíč. Tajný klíč jde totiž využít pro „odemčení“ karty a vyčtení data.

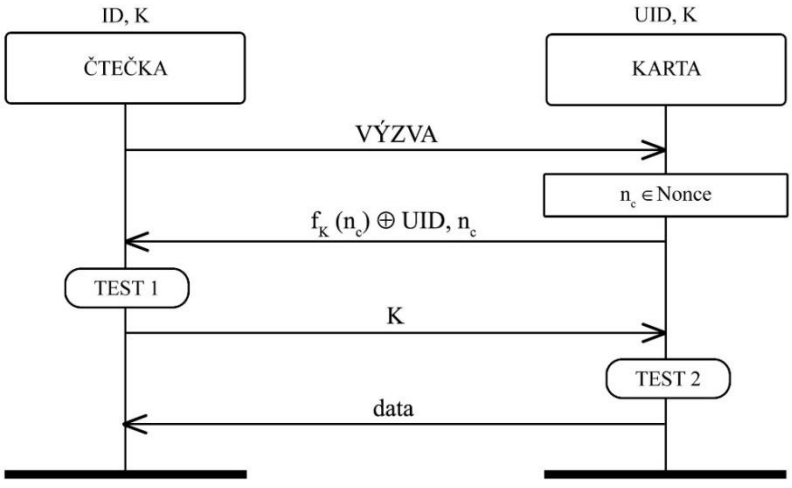


Obr. 4.5 Jednoduchý Hash Lock protokol

Protokol Weis

Jinak také Hash Lock protokol s pseudonáhodnou funkcí. Je protokol postavený na Hash Lock protokolu. Pouze místo haše UID využívá *pseudonáhodnou funkci*. Pseudonáhodná funkce používá náhodná data jako inicializační vektor. Rozšiřuje prostor vstupů na prostor pseudonáhodných výstupů a je považován za silnější kryptografické primitivum než funkce hašovací.

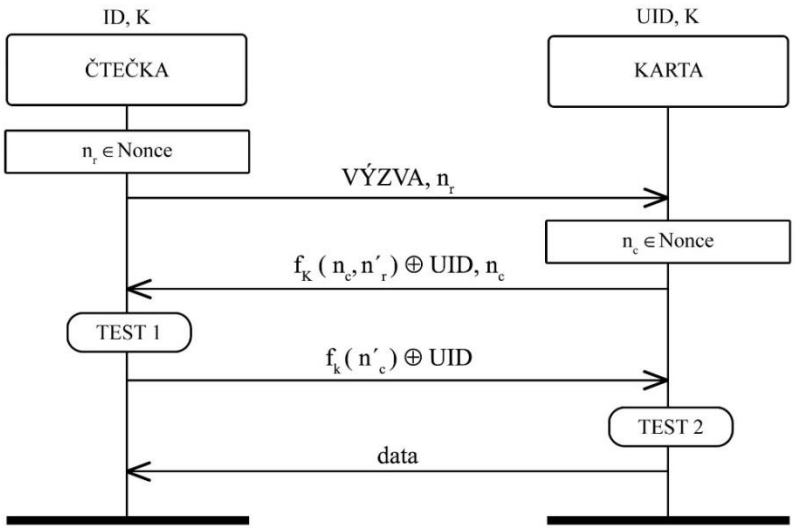
Protokol začíná vysláním výzvy čtečkou. Karta si po obdržení výzvy vygeneruje náhodný řetězec. A odpoví čtečce pseudonáhodnou funkcí aplikovanou na náhodný řetězec a UID karty. Ve zprávě je přiložen i náhodný řetězec. Ten poté čtečka použije na pseudonáhodnou funkci. Na tu následně aplikuje jednotlivá UID z databáze a hledá shodu s přijatou zprávou od karty. V případě nalezení shody si může terminál kartu odemknout, pomocí jejího tajného klíče.



Obr. 4.6 Protokol Weis

Nevýhodou je opět odesílání klíče a dat z karty v otevřeném formátu. Jednoduchému přečtení tajného klíče lze zabránit při využití pseudonáhodné funkce i na straně čtečky. Čtečka tak může použít pseudonáhodnou funkci vytvořenou pomocí náhodného řetězce karty

aplikovaného na tajný klíč. Toto bezpečnostní vylepšení má implementován vylepšený protokol Weis.



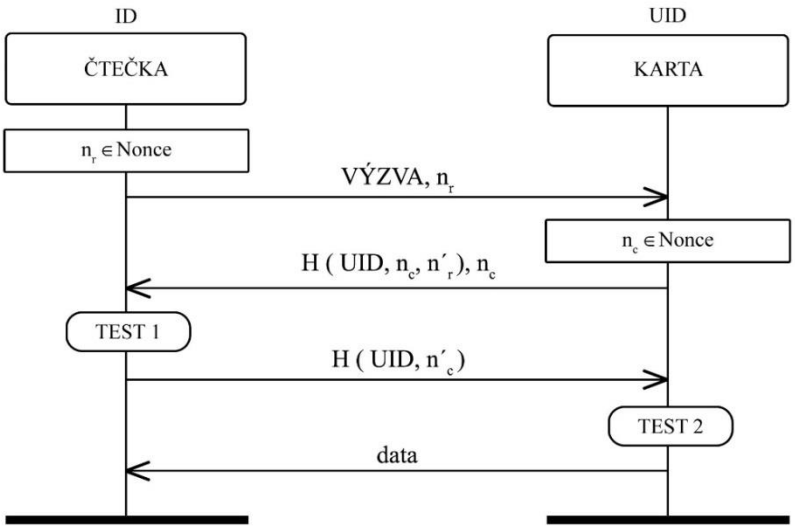
Obr. 4.7 Bezpečnostní vylepšení protokolu Weis

RKKW protokol

Název *RKKW* je odvozen z prvních písmen jmen autorů Rhee, Kwak, Kim a Kwon. Tento protokol je opět založen na protokolu Hash Lock. Hash Lock vylepšuje o zavedení náhodných řetězců, jak na straně čtečky, tak na straně karty.

Autentizace začíná výzvou odeslanou z čtečky společně s jejím náhodným řetězcem. Karta výzvu přijme a vygeneruje si svůj náhodný řetězec. Pomocí obou náhodných řetězců a UID vytvoří haš, jež odešle spolu se svým náhodným řetězcem. Čtečka pak prohledá databázi, kde hledá UID pomocí kterého a obou náhodných řetězců vytvoří haš shodnou s haší přijatou od karty. Aby se prokázala i čtečka kartě odešle čtečka haš UID a náhodný řetězec karty.

Jediným problémem tohoto protokolu je poslední krok a tím je odeslání dat v otevřené podobě. Existuje ale i možnost implementovat tento protokol na kartu podporující šifrování. Tato variant je sice nákladnější, ale zbavuje tento protokol poslední velké bezpečnostní díry.



Obr. 4.8 RKKW protokol

Pravděpodobnostní protokoly

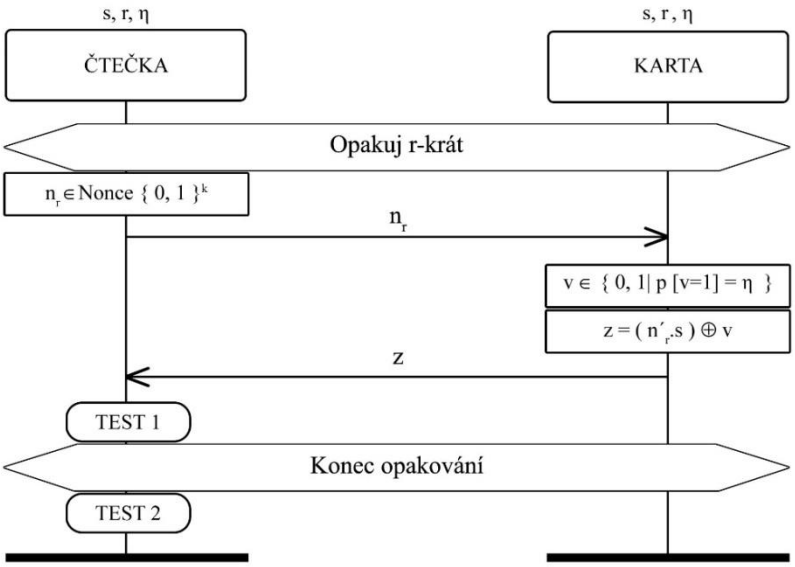
Protože u levných karet není možné použít silné šifrování jako je např. *AES*. Byla vyvinuta rodina protokolů s relativně velkou bezpečností pojmenované *HB pravděpodobnostní protokoly* popsané v [9]. Ty využívají náhodná data (šum) k doplnění zašifrované zprávy s pravděpodobností η. Tím je také zapříčiněno to, že kontrola karty selže s pravděpodobnostní η. Pokud by se útočník snažil dešifrovat zprávu, povede se mu to maximálně s pravděpodobností 1−η. To však ještě neposkytuje silnou ochranu.

Požadovaného zabezpečení se dosáhne opakováním autentizace. Celý proces autentizace je spuštěn *r*-krát. Každá část má vlastní hodnotu η. Po r spuštěních bude karta akceptována, pokud kontrola selže méně než $\eta \cdot r$-krát. Pravděpodobnost, že útočník bude schopný dešifrovat všechny zprávy r je nanejvýš $(1-\eta)^r$. Takže při správně zvolených hodnotách η a *r* je tato šance velmi malá.

HB protokol

Karta a čtečka spolu sdílí náhodný k-bitový tajný klíč *s*. Čtečka vyšle náhodný k-bitový řetězec n_r. Na tuto výzvu karta odpoví binárním skalárním součinem $(n_r' .s)$. Odpověď je jednobitová tzn., že její hodnotu lze uhodnout s 50% pravděpodobností. Díky množství opakování pravděpodobnost klesá na 2^{-r}. Aby bylo zabezpečení ještě větší, je aplikován náhodný bit v, který nabývá hodnoty 1 s pravděpodobností η. Ten je aplikován na odpověď karty $(n_r'.s) \oplus v$. Takto vytvořená odpověď je přijata čtečkou a otestována jestli je rovna hodnotě vypočtené čtečkou. Poté co je zopakováno všech r opakování, uskuteční se druhý test. Karta je akceptována pokud první test selhal méně než $\eta \cdot r$-krát.

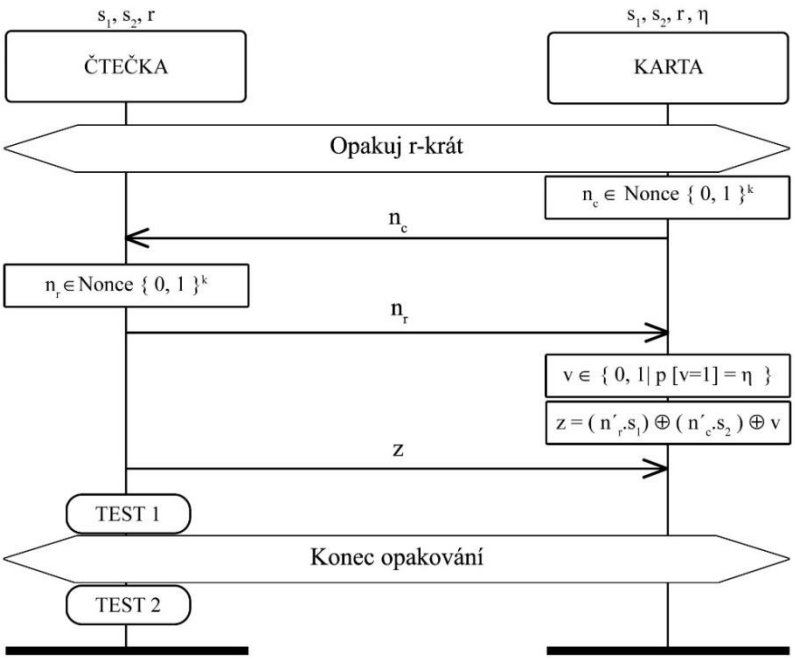
Protokol je odolný vůči útočníkovi, který nemůže libovolně přistupovat ke kartě. Pokud však může útočník kartě zasílat libovolné vstupy a má možnost je dostatečně opakovat. Existuje velká pravděpodobnost, že tajný klíč odhalí. Další možností je zvolení vhodné výzvy (např. lineární kód) a opakovaně tuto zasílat kartě. Po dostatečně mnoha opakováních by mělo být možné odvodit hodnotu $(n_r'.s)$. Po získání k takových hodnot je opět možné odvodit hodnotu klíče pomocí Gaussovy eliminace.



Obr. 4.9 HB protokol

Protokol HB+

Protokol HB+ je vylepšenou verzí protokolu HB. Do protokolu je přidáno náhodné číslo na straně karty a druhý k-bitový klíč. Po výměně náhodných čísel, karta odpovídá čtečce zprávou $z = (n_r' \cdot s_1) \oplus (n_c \cdot s_2) \oplus v$. Čtečka porovná přijatou zprávu s vlastním výpočtem $(n_r' \cdot s_1) \oplus (n_c \cdot s_2)$ a akceptuje, jsou-li výsledky totožné.



Obr. 4.10 HB+ protokol

4.3.2. *AUTENTIZACE KARET VYUŽÍVAJÍCÍ SYMETRICKÉ KRYPTOGRAFIE*

Tyto autentizační protokoly jsou již založeny na silné kryptografii. Většinou již jsou použity pouze na kartách obsahujících mikroprocesor. Najdou se však i výjimky, kdy výrobce

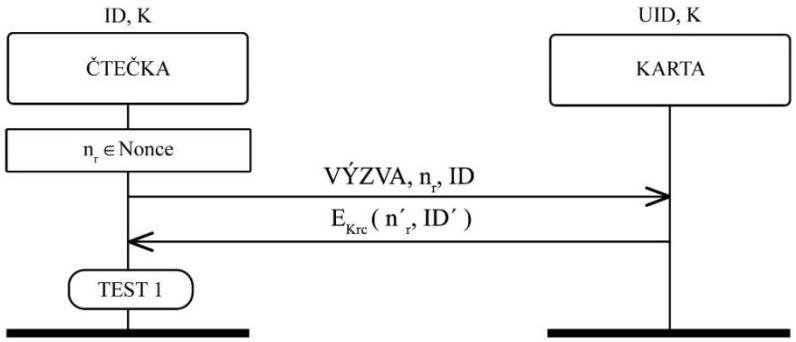
implementuje svoje proprietární řešení do karty s jednoduchou logikou. Příkladem takové karty využívající silné kryptografie je NXP Mifare Classic. V minulosti se však ukázalo, že tato karta má velké bezpečnostní díry.

Standard ISO/IEC 9798-2 definuje šest autentizačních protokolů. Čtyři z nich se zabývají autentizačními mechanizmy mezi dvěma entitami, kde není zapojena žádná třetí důvěryhodná strana. Dva z těchto čtyř se týkají autentizace jednotlivé entity, zatímco druhé dva jsou mechanizmy pro autentizaci vzájemnou. Zbývající mechanizmy vyžadují pro svoji funkci důvěryhodnou třetí stranu pro ustavení společného tajného klíče. V autentizačních mechanismech specifikovaných v této části ISO/IEC 9798 entita, která má být autentizována, potvrzuje svoji identitu prokázáním znalosti tajného autentizačního klíče. Toho je dosaženo tak, že entita použije svůj tajný klíč k zašifrování specifických dat. Zašifrovaná data může dešifrovat každý, kdo sdílí tajný autentizační klíč entity.

V protokolech dle ISO/IEC 9798-2 není pevně definován symetrická šifra. Lze proto užívat šifrování pomocí AES, DES nebo 3DES. Další možností je využití i klíčované jednosměrné hašovací funkce. Při využití hašování zavisí bezpečnost protokolu na použité hašovací funkci. Pokud je však použito přehnaně dlouhých hašů, prodlužuje se doba autentizace. Jako bezpečný a zároveň přiměřeně dlouhý se používá haš SHA-1.

Jednostranná autentizace

Při jednostranné autentizaci čtečka posílá výzvu obsahující náhodné řetězec karty a svůj identifikátor. Karta za použití svého tajného klíče K zašifruje přijatou zprávu a pošle zpět. Čtečka pomocí svého klíče zprávu dešifruje a porovná náhodný řetězec a ID, neliší-li se od těch na začátku odeslaných.



Obr. 4.11 Jednostranně autentizující protokol dle ISO/IEC 9798-2

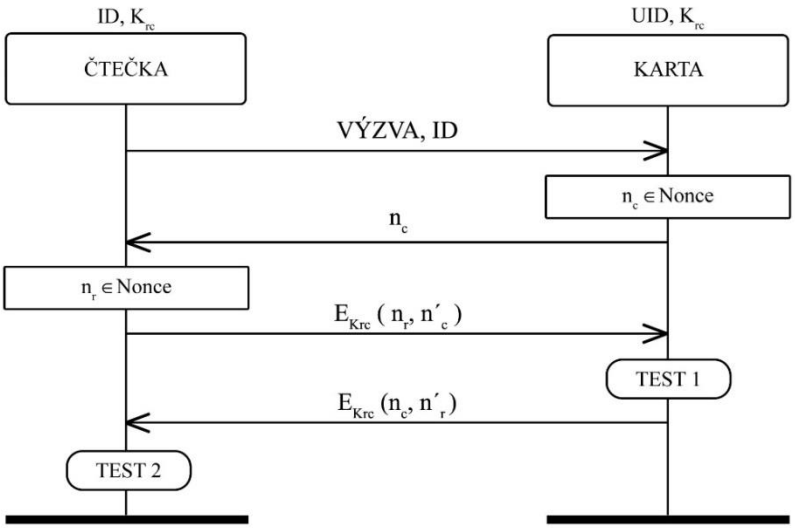
Oboustranně autentizující

Korektně implementovaný protokol zajišťuje bezpečnou vzájemnou autentizaci komunikujících entit.

Protokol začíná vysláním výzvy a ID čtečky směrem ke kartě. Karta odpoví vygenerovaným náhodným řetězcem. Čtečka použije svůj tajný klíč a s jeho pomocí zašifruje svůj náhodný řetězec s tím přijatým od karty. Karta dešifruje přijatou zprávu a zkontroluje, jestli obsahuje její náhodný řetězec. Když je test úspěšný, prohodí pořadí náhodných čísel a znovu zašifruje. Čtečka pak již jen ověří, jsou-li prohozena náhodná čísla. Tím je ověřeno, že karta zná tajný klíč. Bez jeho znalosti by nebylo možné takovou zprávu vygenerovat.

Tuto autentizaci využívá například karta NXP Mifare Classic, dříve využívaná v mnoha městech jako elektronická jízdenka do hromadné dopravy. Týmy expertů, však dokázaly zabezpečení těchto karet prolomit. Byla objevena slabina v pseudonáhodném generátoru

čísel. Ten je schopen generovat pouze 65536 čísel. Což pak dělá z proudové šifry CRYPTO1 společnosti NXP snadný cíl útoků. Na základě odchycené komunikace je možné zpětně získat tajný klíč. [10]



Obr. 4.12 Oboustranně autentizující protokol dle ISO/IEC 9798-2

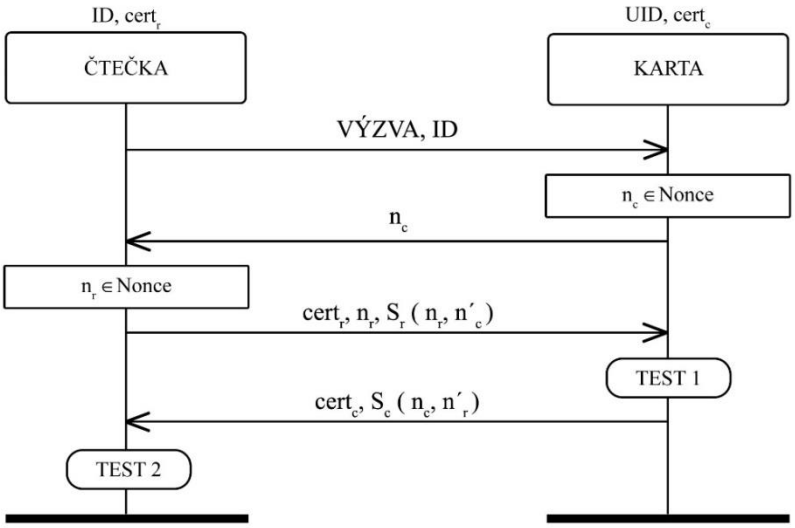
4.3.3. AUTENTIZACE KARET VYUŽÍVAJÍCÍ DIGITÁLNÍCH PODPISŮ

Protokol dle ISO/IEC 9798-3

Propojením vzájemně autentizujícího protokolu dle ISO/IEC 9798-2 a digitálních podpisů vznikne ISO/IEC 9798-3. Rozdílem oproti předchozímu protokolu je využití asymetrické kryptografie.

Do momentu, kdy odesílala čtečka zašifrovanou zprávu se autentizace dle ISO/IEC 9798-3 neliší od autentizace dle ISO/IEC 9798-2. Neodesílá se totiž zpráva zašifrovaná pomocí symetrického šifrování. Místo toho jsou náhodné řetězce zašifrovány soukromým klíčem. Ke zprávě je přiložen náhodný řetězec čtečky a certifikát spojující danou entitu s použitým soukromým klíčem. Certifikát obsahuje i veřejný klíč, díky kterému karta dešifruje zprávu a porovná svůj náhodný řetězec s řetězcem získaným ze zprávy. Karta prohodí náhodné řetězce a zprávu zakóduje svým privátním klíčem. K zašifrovaným řetězcům přiloží svůj certifikát a odešle zpět čtečce. Čtečka po dešifrování zkontroluje, zda náhodné řetězce nebyly změněny. A tím ukončí autentizaci.

Bezpečnost protokolu je závislá pouze na délce použitého klíče. Délka klíče je sice v případě čipových karet značně omezená, ale jinak je tento protokol považován jako bezpečný.



Obr. 4.13 Vzájemná autentizace dle ISO/IEC 9798-3

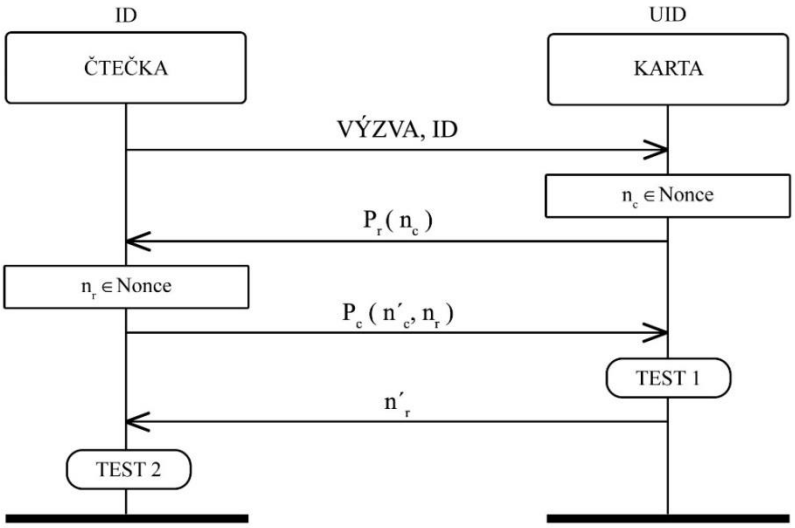
Needham-Schroeder

Needham-Schroeder je protokol poskytující vzájemnou autentizaci dvou entit pomocí asymetrické kryptografie. Jeho výhodou je možnost ustálení session klíčů pro další komunikaci.

Komunikaci opět začíná čtečka odesláním výzvy. Karta vytvoří náhodný řetězec a zašifruje ho pomocí veřejného klíče čtečky. Čtečka zprávu dešifruje a vytvoří svůj náhodný řetězec. Obě náhodné hodnoty čtečka zašifruje veřejným klíčem karty a odešle. Čtečka dešifruje přijatou zprávu. Porovná přijatý náhodný řetězec se svým vlastním. Pokud je shodný odešle náhodné číslo čtečky. Pokud souhlasí přijaté a vygenerovaný řetězec čtečky je autentizace úspěšně ukončena.

Pro případné ustálení klíčů k následné šifrované komunikaci stačí zašifrovat odeslání náhodného čísla v poslední přenášené zprávě. To by znamenalo, že obě komunikující entity disponují dvěma náhodnými čísly, z nichž lze předem definovanou operací např. XOR odvodit session klíč.

Zprávy v tomto protokolu jsou vždy šifrovány veřejným klíčem protistrany. Tzn., že je nutné, aby karta disponovala všemi veřejnými klíči čteček, u kterých se má autentizovat. Což znesnadňuje jednoduchou centralizovanou zprávu klíčů.

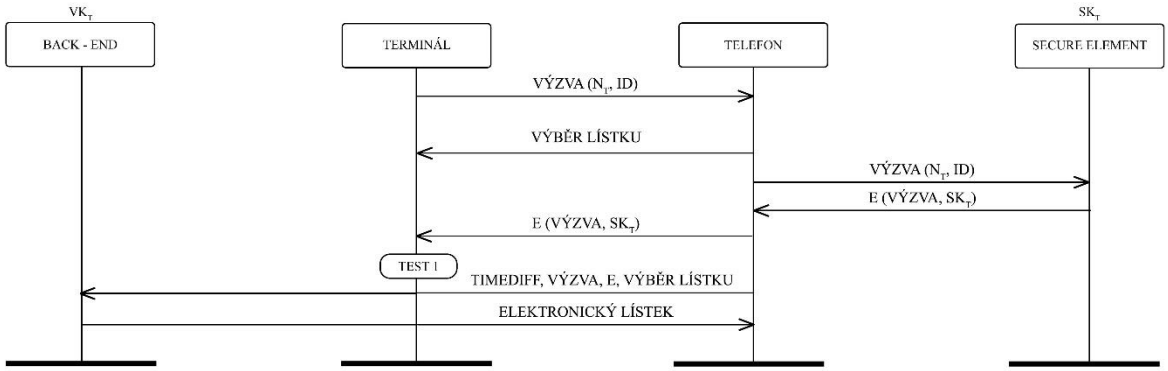


Obr. 4.14 Needham-Schroeder

4.3.4. AUTENTIZACE PŘI NÁKUPU LÍSTKU HROMADNÉ DOPRAVY POMOCÍ NFC

Ve většině velkých měst je možnost platit lístky za hromadnou dopravu pomocí bezkontaktních karet nebo pomocí NFC v telefonu. V Evropě se využívá především karet Mifare a v Asii převažují karty Felica. Bezkontaktní karty většinou využívají asymetrického šifrování dle protokolu 9798 – 3.

V této kapitole bude vysvětlen průběh autentizace pomocí telefonu vybaveného NFC. Kvůli využívání soukromého klíče telefonem, je nutné, aby obsahoval secure element. Secure element umožní bezpečné uložení soukromého klíče a bude v něm probíhat podepisování zpráv. Způsob autentizace, který bude popsán je vyvinut RFID Lab of „La Sapienza“, University of Rome a slouží pro nákup lístků hromadné dopravy v místech, kde se prochází vstupními bránami. Tento systém ke správné funkci potřebuje online propojení terminálu s back-end systémem. Back-end systém má totiž uloženy údaje o stavu uživatelských kont a zároveň spravuje veřejné klíče všech karet.



Obr. 4.15 Autentizace NFC platby hromadné dopravy

Cely průběh autentizace začne přiložením telefonu. Čtečka vyšle telefonu výzvu obsahující náhodné číslo a ID čtečky. Telefon odešle požadavek, v němž je uveden druh klientem požadovaného lístku. Zároveň s tím je v secure elementu podepsán transportním

klíčem výzva s náhodným číslem. Takto zašifrovanou zprávu terminál přijme a pomocí veřejného klíče, který získá z back-end systému. Pokud v dešifrované zprávě najde na začátku vygenerované náhodné číslo, považuje autentizaci za úspěšnou. A přechází na zjišťování stavu konta. Pokud je dostatečný na zaplacení lístku brána je otevřena a člověk může projít. Volitelná fáze autentizace je obdržení kontrolního tokenu (elektronický lístek) od back-end systému. Ten telefon může obdržet na „oplátku" za zaevidování transakce. Zaevidování proběhne tím, že telefon odešle back-end sytému časovou známku, výzvu od terminálu, výzvu podepsanou soukromým klíčem a druhem požadovaného lístku.

4.3.5. AUTENTIZACE ELEKTRONICKÝCH PASŮ

Elektronické pasy jsou doklady obsahující RFID čipy na kterých jsou uloženy informace o držiteli pasu. Aby tyto informace nemohl číst každý a také proto, aby bylo možné ověřit pravost dat, využívá se autentizace.

Paměť elektronických pasů je rozdělena na externě nedostupný segment a segment externě dostupný. V externě nedostupném segmentu je bezpečně uložen soukromý klíč. V externě dostupné paměti jsou ukládána ostatní data určená pro čtení. Tato data jsou strukturována do skupin DG (Data Group). Tyto skupiny jsou označeny *DG1* až *DG16*. A každá z těchto skupin je podepsána soukromým klíčem.

K ověření toho, že data nejsou jakkoliv pozměněna je využito tzv. pasivní autentizace. Pro Českou republiku je uložen ve skupině *DG15* veřejný klíč. Veřejné klíče se předávají mezi státy diplomatickými službami, takže je možné číst i pasy cizích státních příslušníků. Pomocí získaného veřejného klíče je pak možné přečíst obsah jednotlivých datových skupin podepsaných soukromým klíčem. Na dešifrované datové skupiny je použita hašovací funkce. Vypočtené haše jsou pak porovnány s údaji uloženými v pasu. V souboru $EF.SO_D$ jsou totiž uloženy haše všech datových skupin. Pokud se vypočítaná a uložená hodnota neliší, je prokázáno, že data nebyli pozměněny.

Má-li být ověřena autentičnost čipu je potřeba využít aktivní autentizace. Ta je prováděna tak, že čtecí terminál vygeneruje náhodné číslo. To spolu s výzvou zašle pasu, který na náhodné číslo aplikuje svůj soukromý klíč a odešle terminálu. Terminál z buňky *DG15* získá veřejný klíč, s jehož pomocí ověří, jestli je náhodné číslo přijaté od pasu, stejné jako to vygenerované.

4.4. AUTORIZACE BEZKONTAKTNÍCH SYSTÉMU

Všechny autorizační jednotky, které slouží k řízení přístupu, musí obsahovat informace o všech identitách. Každá z identit pak musí mít přiřazen přístupová práva, tj. kam může vstoupit, případně na co má právo. V jednotce musí být uloženo i to jak se daná identita bude prokazovat (dokazovací faktor). Například může být vyžadována kombinace vlastnosti čipové karty a znalost pinu. A pro úspěšnou autentizaci přístupová jednotka musí znát ověřovací faktor. Což jsou informace, pomocí kterých přístupový systém ověřuje pravost dokazovacího faktoru. U čipových karet to zpravidla bývá obsah jejich paměti.

Rozlišujeme dva druhy přístupových systému. První z nich systém decentralizovaný má pro každé přístupové místo svou autonomní přístupovou jednotku. Systém musí obsahovat všechny výše zmíněné informace nutné pro rozhodnutí o udělení přístupu. Problém tohoto řešení je nesnadná správa. Kdy je třeba každou přístupovou jednotku konfigurovat zvlášť. V případě, že střežíme cenná aktiva, musíme udržet kryptografické klíče v tajnosti. Pro jejich zabezpečení je výhodné využití SAM (Secure Access Module) čip. Ten je odolný vůči široké škále útoků a k jeho překonání by dle certifikace EAL 5+ muselo být

vynaloženo 100 000 €. Případně je možné využít ještě bezpečnější HSM (Hardware Security Module).

Druhým z možností řízení přístupu je centralizovaný systém. Tam veškeré rozhodování o udělení přístupu obstarává jedna přístupová ústředna připojená pomocí komunikační sítě k jednotlivým přístupovým bodům. Výhodou je mnohem jednodušší centralizovaná správa. Nevýhodou zase nutnost komunikačního systému.

4.5. SHRNUTÍ

Způsobů jak prokázat svoji identitu pomocí bezkontaktní technologie je obrovské množství. Existují desítky různých autentizačních protokolů, které se liší svou výpočetní náročností a bezpečností.

Pokud by měl být vyzdvižen protokol s největším zabezpečením, zřejmě by to byl protokol vzájemné autentizace využívající asymetrické kryptografie. Bezpečnost je závislá jen na délce šifrovacího klíče. Čím je delší klíč, tím se zvedá bezpečnost ale i nároky na výpočetní výkon. V praxi tedy musíme volit kompromis mezi dostatečností zabezpečení a dobou, jakou proces autentizace zabere.

Karty, které umožňují využívat symetrické a asymetrické kryptografie, jsou ale poměrně drahé. V případě, že potřebujeme dosáhnout poměrně velké bezpečnosti a nechceme vynaložit vysoké náklady na karty, stávají se rozumnou volbou pravděpodobnostní protokoly.

Pokud se však zaměříme na to, jaká je situace v praxi, zjistíme, že v mnoha případech není autentizace nijak zajištěna. Povolení nebo odmítnutí přístupu je udělováno jenom na základě UID. To však není nijak zabezpečeno a útočník ho může lehce zkopírovat.

Autentizační schémata předvedená v této kapitole jsou univerzální, a proto je můžeme bez problému implementovat i do karet kontaktních nebo k autentizaci pomocí NFC. Jak již bylo několikrát zmíněno, jediným omezením využití protokolu se stává výpočetní výkon.

5. BANKOVNICTVÍ

S řízením přístupu se setkáváme i v momentech, kdy jsme v kontaktu s naší bankou. Proces ověření pravosti identity, zkontrolování stavu účtu a rozhodnutí o oprávněnosti požadavku na transakci, je potřeba provést pokaždé, když chceme provést jakoukoliv bezhotovostní transakci. Bezhotovostní platby můžeme provádět pomocí internetového bankovnictví, pomocí kreditní karty nebo můžeme využít i online internetové platby. Manipulace s financemi se stávala terčem útoků již po staletí a bylo potřeba ji dobře chránit. Dříve pro odcizení bylo nutné vykrást banku. Ještě nedávno pro podvod s internetovým bankovnictvím stačilo zjistit uživatelské jméno a k němu přiřazený PIN.

Kvůli bezpečnostním hrozbám je tak k autorizaci prováděné transakce, ve většině případů využívána nějaká forma dvou faktorové autentizace. Výběr autentizačních metod je závislý na mnoha okolnostech. Jedna metoda autentizace se používá pří výběru peněz z bankomatu, jiná při platbách u obchodníků nebo při přístupu do internetového bankovnictví.

5.1. INTERNETOVÉ BANKOVNICTVÍ

Internetové bankovnictví nebo také online bankovnictví je forma kontaktu klienta s jeho bankou. Prostřednictvím webového bankovního portálu je možné obsluhovat svůj účet. Tam můžeme získat informace o zůstatku na účtu, provést jednorázový nebo trvalý platební příkaz.

Zabezpečení komunikace v rámci internetového bankovnictví bývá obvykle řešeno pomocí standardního protokolu *TLS* případně *HTTPS*. Většina českých bank pak pro svou identifikaci používá certifikáty uznané veřejnými certifikačními autoritami, které bývají standardní součástí webového prohlížeče.

5.1.1. AUTENTIZACE INTERNETOVÉHO BANKOVNICTVÍ

Pro ověření identity v internetovém bankovnictví se v současnosti používá více faktorová autentizace. Ale nebylo tomu tak vždy. Do roku 2005 bylo celkem obvyklé, že pro autentizaci stačila znalost identifikačního jména a PINu. V důsledku toho docházelo k podvodům. Proto organizace FFIEC (Federal Financial Institutions Examination Council) vydala doporučení ohledně autentizace v internetovém bankovnictví (Guidance on Authentication in an Internet Banking Environment). V tomto dokumentu je uvedeno, že jednofaktorová autentizace je zcela nevhodná a je příčinou mnoha finančních podvodů a krádeží identity. I přes tato doporučení vyžadovala více faktorovou autentizaci v roce 2007 pouze 4% bank.

Nyní už je ale standard používání více faktorové autentizace pravidlem. Pro autentizaci lze použit:

- Jméno a heslo
- Autentizace pomocí jednorázového kódu poslaného na mobilní telefon
- Certifikát poskytnutý bankou na nechráněném mediu (Softwarový token)
- Hardwarové tokeny

Nejběžněji se setkáme s kombinací, kdy se uživatel autentizuje pomocí zadání uživatelského jména a hesla. Následně je mu zaslán tzv. *OTP* (One Time Password) na jeho mobilní telefon. OTP je platný jen pro jednu autentizaci a pak se stává neplatným. Takže i v případě, že je tento kód útočníkem odchycen, není mu v podstatě k ničemu.

Další z možných metod jsou jednorázové autentizační kódy vydávané ve formě tabulky, na které je uvedeno číslo kódu a kód. Při autentizaci kontrolér vyzve klienta číslem požadovaného kódu. Klient si vyhledá v tabulce kód s požadovaným pořadovým číslem a ten zašle kontroléru. Kódy jsou stejně jako v případě SMS zpráv jednorázové a proto po vyčerpání všech kódů z tabulky musí dojít k distribuci tabulky nové. Tak častá nutnost výměny není u autentizačních tabulek s opakovanými autentizačními kódy. Ty využívají tzv. grid karty, což jsou plastové karty o velikosti platební karty. Na její povrch je natištěná tabulka, která je rozdělena na množství řádků a sloupců označených souřadnicemi. Při autentizaci zašle kontrolér souřadnice nějaké buňky tabulky a klient mu pak musí zaslat kód, který je v uvedené buňce. Tato autentizace je o něco méně bezpečná, než předchozí jednorázové kódy.

Pokud ale uživatel vyžaduje bezpečnější přístup ke svému účtu, většinou si musí připlatit. Zaručení bezpečnějšího přihlášení bývá realizováno pomocí hardwarového nebo softwarového tokenu. Pod pojmem softwarový token si můžeme představit soubor obsahující soukromý šifrovací klíč případně generátor OTP uložený přímo na zařízení, kterým přistupujeme do internetového bankovnictví. Pro použití tokenu je nutné zadat PIN, který je ověřován na vzdáleném severu. Výhodou tohoto tokenu je výrazně nižší cena oproti tokenu hardwarovému. Tento způsob má ale i nevýhody. Pokud útočník získá kontrolu nad počítačem, z kterého se klient přihlašuje, může softwarový token zkopírovat. Občas se navíc najde slabina v některém z těchto softwarových tokenů, jako tomu například bylo v roce 2012 u softwarového generátoru jednorázových hesel *RSA SecureID*. Pomocí reverzního inženýrství byly objeveny informace, které se používají pro generování pseudonáhodných čísel v knihovnách Windows. Se znalostí těchto informací je pak možné generovat stejná čísla, jaká generuje RSA SecureID.

Hardwarový token má mnoho podob. Pro identifikaci uživatele může být použita čipová karta, na které je uložena dvojice privátní, veřejný klíč a certifikát. Pro zvýšení bezpečnosti bývá přístup k autentizačním údajům ještě chráněn PINem. Za vydání čtečky pro čtení paměťových karet sloužících pro autentizaci a autorizaci si například ČSOB účtuje 500 Kč. Soukromý a veřejný klíč společně s certifikátem mohou být uloženy i na flash disku. U této varianty není potřeba u počítače žádné další přídavné zařízení, protože USB zdířku obsahuje již většina počítačů. Přístup k údajům bývá také chráněn PINem.

Způsob autentizace vyvinutý společností MasterCard je nazýván *CAP* (Chip Authentication Program). Tato technika byla převzata později i firmou VISA, která ji přejmenovala a používá ji pod názvem *DAP* (Dynamic Passcode Authentication). Prakticky se jedná o autentizační kalkulátor, který generuje autentizační odpověď na základě platební karty klienta, PINu platební karty a popřípadě nějaké výzvy od banky. Kalkulátor CAP může pracovat ve třech režimech. V prvním případě pouze, vygeneruje jednorázové heslo, podobně jako výše popsaný SecureID. V druhém případě je uživateli předložena bankou výzva, kterou je nutné pomocí klávesnice zadat do kalkulátoru. Na jejím základě je pak vygenerován autentizační kód. A poslední možností je, že výzva není náhodně generována, ale je použit některý z údajů o transakci. Jako je číslo transakce, celková částka transakce apod. Kalkulátor následně může vytvořit kód sloužící k autentizaci uživatele.

5.1.2. AUTORIZACE

V roli kontroléru, který rozhoduje o oprávněnosti provedení transakce, vystupuje sám majitel účtu. Ten musí rozhodovat, zda transakce je oprávněná či nikoliv. Pro autorizaci se využívají stejné mechanismy jako pro autentizaci.

Nejčastěji se u běžných účtů setkáme s autorizací transakce pomocí jednorázového hesla poslaného v SMS zprávě. SMS zpráva obsahuje údaje o transakci a heslo s omezenou časovou platností. Jako autorizační metoda může být ale použit i soukromý klíč a příslušný podpisový certifikát uložený buď na počítači, nebo na čipové kartě.

5.2. KREDITNÍ PLATBY

Při jakékoliv bezkontaktní platbě v obchodě za pomocí platební karty, ať už kontaktní nebo bezkontaktní se setkáme se specifikací *EMV*. V roce 1999 se společnosti VISA International, MasterCard International a Europay International rozhodli vytvořit společnost EMVCo LLC, která měla na starosti sjednocení společných aktivit těchto firem. EMVCo tak postupně začalo tvořit specifikace pro vzájemnou interoperabilitu platebních karet a terminálu od různých výrobců.

EMV pro čipové karty je postaveno na základě řady standardů ISO 7816. Bezkontaktní platby využívají standardu ISO 14443. EMV specifikace pro kontaktní platby je rozdělena na 4 knihy. Ty jsou průběžně aktualizovány. Současná verze 4.3 byla vydána v listopadu roku 2011. Kniha první popisuje požadavky kladené při použití nezávislé čipové karty na rozhraní terminálu. Najdeme v ní především podrobné informace o mechanických a elektrických charakteristikách čipových karet a terminálů. Popisuje i souborový systém a příkazy, kterých je možno využít. V knize druhé je popsána bezpečnost a správa klíčů využívaných pro autentizaci. V třetí knize je specifikován aplikační protokol. Poslední z knih definuje požadavky kladené na držitele karty, obsluhu a rozhraní nabyvatele.

Pro bezkontaktní platby definovalo EMV specifikaci EMV Contactless nyní ve verzi 2.3. Tato specifikace je rozdělena na oddíly A-D, přičemž oddíl C je ještě rozdělen na 5 částí. Kniha A je pojmenována Architektura a obecné požadavky. Jejím obsahem jsou charakteristiky karet a terminálů. V knize B je popsáno objevení a vyjednání aplikace, kterou podporují obě strany komunikace. Kniha C je rozdělena na 5 částí, kdy každá z nich popisuje specifikaci jednoho z jader využívaných pro komunikaci. A poslední kniha D popisuje minimální funkce vyžadované na kartě i terminálu pro zajištění správného chodu. Shrnutí všech knih je dostupné v [11].

5.2.1. AUTENTIZACE PLATEBNÍ KARTY

Pro autentizaci platební karty je využíváno digitálních podpisů. Generování digitálního podpisu pro celou zprávu by bylo výpočetně velmi náročné a proto se podepisuje pouze část dat. Pomocí algoritmu SHA-1 se spočítá 20-ti bajtový haš kód originální zprávy. Zároveň je zpráva rozdělena na dvě části. Z leva ze zprávy je odděleno X bajtů (LeftMSG) na základě kterých je tvořen reprezentativní řetězec, který se pomocí soukromého klíče podepíše. Řetězec je tvořen jako „6A“ + „LeftMSG“ + haš + „BC“. Kde 6A a BC jsou konstanty. Celý řetězec je dlouhý X + 22 bajtů (20 bajtů je dlouhý haš a 2 bajty mají konstanty). Příjemci je odeslána reprezentativní řetězec a zbytek zprávy.

Pro ověření podpisu je nejdříve ověřena délka podpisu v bajtech. Poté se tento řetězec dešifruje pomocí veřejného klíče. Tím získá příjemce reprezentativní řetězec, který lze rozdělit na jednotlivé složky, z kterých byl vytvořen. Zkontrolují se hodnoty konstant a spojí se LeftMSG se zbytkem zprávy. Z takto vytvořené zprávy je spočítán haš, který je možno porovnat s hašem obsaženým v přijatém řetězci. Pouze pokud všechna ověření proběhla v pořádku, je možné považovat zprávu za pravou.

K autentizaci dat uložených na kartě jsou využívány tři mechanizmy:

- Statická autentizace dat (SAD)
- Dynamická autentizace dat (DDA)
- Kombinovaná dynamická autentizace dat (CDA)

Pro autentizaci držitele karty je využívána znalost PINu. V dřívějších dobách se využíval i podpis držitele karty. PIN je možné ověřovat buď offline. Je ověřována informace pouze mezi kartou a klávesnicí, na které je PIN zadáván. K tomu je využito šifrování dat za pomoci asymetrického páru klíčů. Terminál nejdříve získá od karty odpovídající veřejný klíč a z klávesnice dostane PIN v podobě textového řetězce. Dále požádá kartu o vygenerování náhodného čísla, které spolu s PINem zašifruje pomocí veřejného klíče. Vzniklou šifru zašle kartě. Čipová karta tato data ověří s použitím soukromého klíče. Pokud jsou údaje ověřovány online, neprovádí kontrolu dat karta, ale banka vydávající kartu zákazníkům. Více se problematikou zajímá [12].

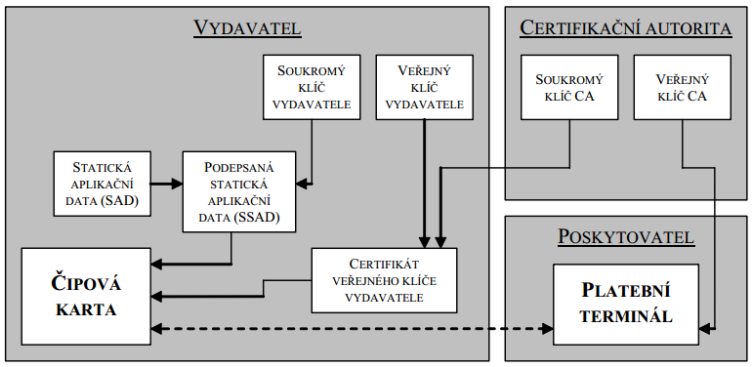
Statická autentizace dat

Metoda statické autentizace dat potvrzuje pravost dat uložených na kartě. Veškerá autentizace je prováděna terminálem. Tato autentizace neřeší problém s duplikováním karty. Pro SAD je nutná existence certifikační autority a vydavatele. A oba musí disponovat dvojicí veřejného a soukromého klíče. Certifikační autorita je nutná pro ověření identity vydavatele karty.

Autentizace proběhne tak, že data uložená na kartě spolu s identifikátorem hašovací funkce jsou podepsány soukromým klíčem vydavatele karty. V paměti karty je uložen i certifikát veřejného klíče vydavatele karty, příznak identifikující certifikační autoritu a typ použitého algoritmu, exponent a modulo veřejného klíče.

Terminál nejdříve ověří pomocí veřejného klíče certifikační autority certifikát veřejného klíče vydavatele karty. Veřejné klíče certifikačních autorit musí mít terminál uloženy v paměti. Pokud je certifikát platný a v pořádku, je možné z něj získat veřejný klíč vydavatele karty. Díky veřejnému klíči karty je možné ověřit, zda data byla opravdu podepsána vydavatelem karty. Postup vytvoření digitálního podpisu je popsán výše.

Dle specifikací EMV musí být terminál schopný uložit 6 veřejných klíčů certifikačních autorit. Pro digitální podpis je využíván algoritmus RSA a jako hašovací funkce je využívána SHA-1.



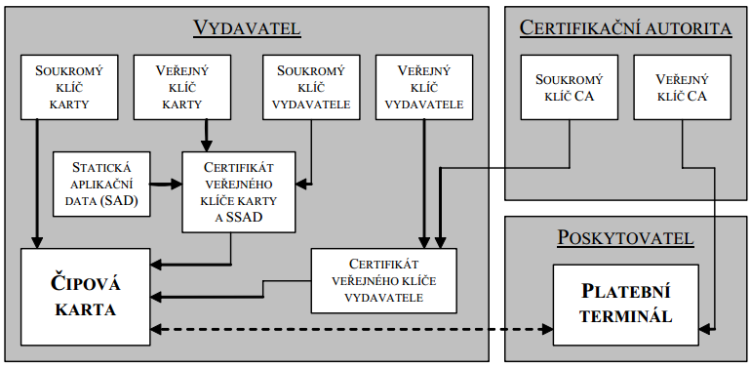
Obr. 5.1 Statická autentizace dat – schéma rozmístění klíčů

Dynamická autentizace dat

Výhodou *dynamické autentizace* je přítomnost dalšího páru RSA klíčů, který není z karty kopírovatelný. Díky tomu není možné takovou kartu duplikovat. Případná padělaná karta neobsahuje soukromý klíč karty, a proto není schopna korektní komunikace s čtečkou.

Prvotní autentizace proběhne stejně jako v případě SAD. Po dešifrování certifikátu veřejného klíče karty získá veřejný klíč vydavatele. S jeho pomocí pak ověří autenticitu dat. A navíc oproti SAD získá veřejný klíč karty. Terminál následně zašle kartě příkaz „*INTERNAL AUTHENTICATE*“ následovaný seznamem datových elementů, které jsou očekávány od karty jako odpověď. Součástí tohoto seznamu je i náhodné číslo vygenerované terminálem. Karta jako odpověď odešle odpověď společně s náhodným číslem. Tato odpověď je digitálně podepsána soukromým klíčem karty. Terminál si je dříve získaným veřejným klíčem přečte a porovná náhodné číslo.

Tato autentizace je už odolná proti kopírování karty, ale riziko zůstává, pokud útočník zasáhne do komunikace. Například pokud terminál požádá kartu o potvrzení transakce a karta se rozhodne transakci odmítnout, je možné odpověď karty změnit. Princip útoku je takový, že podvržená karta obsahuje dva čipy. Kromě běžného, ještě jeden, který zprostředkovává výměnu informací mezi terminálem a originálním čipem. Tuto výměnu pak může libovolně měnit.



Obr. 5.2 Dynamická autentizace dat – schéma rozmístění klíčů

Kombinovaná dynamická autentizace

Kombinovaná dynamická autentizace (CDA) také vyžaduje podepisování přenášených dat. Pro ověření pravosti přenášených dat je využíváno tzv. aplikačního kryptogramu. Ten umožňuje detekovat podvržené zprávy v případě, že se útočník pokouší o útok typu Man In The Middle. Aplikační kryptogram (AC) je 8 - bajtový autentizační kód zprávy (Message Authentication

Code – MAC). V podstatě je to haš vytvořený jednosměrnou funkcí a tajným autentizačním klíčem. Tajný klíč je vygenerován pro každou komunikaci jedinečný. Je odvozen z parametrů, jako je veřejný klíč karty, čítač transakcí a kódu pobočky. Kryptogram by měl obsahovat data obsahující hrazenou částku, kód země terminálu, výsledek ověření certifikátu karty, kód měny, datum transakce, typ transakce, náhodné číslo, čítač transakcí.

Samotných aplikačních kryptogramů existuje jen pár druhů. Pokud karta odmítá provést transakci, odesílá aplikační autentizační kryptogram (AAC). V případě, že karta transakci akceptuje, odešle transakční certifikát (TC) nebo kryptogram s požadavkem o autorizaci (ARQC). TC je odesílán v případě offline autorizace transakce. ARQC je použit v případě, kdy je vyžadována online autorizace transakce. Terminál provede ověření podpisu a haše. Pokud je vše v pořádku, tak se v případě TC transakce provede, anebo je v případě ARQC odeslána k online ověření transakce k vydavateli karty.

5.3. ONLINE PLATBY

Pro jednoduché a rychlé nakupování na internetu je možné platit pomocí údajů z platební karty. Hlavní výhody takového typu transakce je to, že platba je provedena okamžitě a celý proces se obejde bez jakýchkoliv poplatků za převod. K zaplacení zboží pak stačí pouze několik údajů:

- Číslo platební karty
- Jméno držitele karty
- Datum expirace karty
- Verifikační kód

Kromě přímých plateb existuje i možnost placení přes zprostředkovatele. Mezi společnosti, které takové služby poskytují, můžeme řadit PayPal, Moneybookers a PaySec. Jsou to elektronické peněženky, pomocí nichž je také možná online platba.

5.3.1. 3D SECURE

3-D Secure (Three Domain Secure) blíže popsaný v [11] je protokol, který vyvinula společnost VISA. Ta ho ve svých kartách pojmenovala Verified by Visa. Protokol brzy převzala i druhá velká společnost na poli platebních karet, která tento protokol využívá pod názvem MasterCard SecureCode. Obě tyto implementace jsou představovány ve spojení s EMV jako prostředek k bezpečnějšímu provádění online transakcí.

K vytvoření protokolu 3-D Secure vedl společnost VISA masivní nárůst objemu finančních prostředků, které byly převáděny online transakcemi. A tím se začala rozšiřovat záporná stránka věci. Množily se podvodné transakce. Počet podvodných transakcí byl dokonce mnohonásobně vyšší, než počet finančních podvodů provedených tváří v tvář. Řešením těchto problémů se ukázal protokol 3-D Secure založený na technologii XML.

Celý koncept tohoto systému je postaven na propojení autorizace finanční transakce s online autentizací držitele karty. Autorizace a autentizace je vykonávána na základě tří domén. Konkrétně to jsou domény vydavatele a nabyvatele, které jsou propojeny komunikační doménou. Každá z nich v sobě obsahuje několik samostatných entit.

Doména vydavatele

Doména vydavatele je zodpovědná za správu přihlašování majitelů platebních karet podporujících 3-D Secure platby a autentizaci držitele karty při nákupu.

Prvním subjektem vystupujícím v online platbě je držitel karty. Pro uskutečnění platby musí bance, která bude přijímat platbu, poskytnout informace o kartě (číslo karty, datum expirace). A pro autorizaci celé transakce musí přepsat OTP přijatý v SMS zprávě.

Jako prostředník přenosu informací mezi držitelem karty a internetovým obchodem funguje webový prohlížeč držitele karty. Kromě obchodu komunikuje i se serverem pro kontrolu přístupu.

Server pro kontrolu přístupu ověřuje, zda konkrétní číslo platební karty má povolené 3-D Secure platby. Zároveň také provádí autentizaci na základě znalosti údajů o kartě. Sever pro kontrolu přístupu je součástí vydavatelské finanční instituce. Což je banka, která vydala platební kartu a je registrovaným členem asociace Visa nebo MasterCard. Tato instituce poskytuje data o platebních kartách severům karetních asociací (v případě společnosti Visa poskytuje data Visa Directory Serveru). Další činností takové banky je registrace platebních karet do systému 3-D Secure.

Doména nabyvatele

Doména nabyvatele je tvořena především z obchodníka a jeho banky. Je především zodpovědná za proces, který vede k prokázání identity nabyvatele a zpracování transakcí mezi bankami. Dalším úkolem je ověřit, že nabyvatel jedná s bankou, se kterou má sepsanou obchodní smlouvu.

V roli obchodníka vystupuje webová aplikace, která se stará o zpracování celého průběhu transakce. Od vyžádání si autentizace držitele karty po výměnu autorizačních zpráv s nabyvatelskou bankou. Na úplný závěr zpracuje informaci o úspěšnosti (případně neúspěšnosti) celé transakce. Výsledek celé transakce pak zobrazí držiteli karty v podobě internetové stránky.

Banka obchodníka je finanční instituce, která je registrovaným členem asociace Visa nebo MasterCard. Určuje, zda je obchodník vůbec způsobilý k účasti v 3-D Secure transakcích. Od obchodníka přijímá požadavky k autorizaci a následně je předává autorizačnímu systému (v případě společnosti Visa je to VisaNet).

Komunikační doména

Komunikační doména slouží pro přenos zpráv mezi vydávající a nabyvatelskou doménou. Pro přenos využívá běžné bezpečnostní protokoly jako je SSL/TLS spojení. Entity komunikační domény jsou závislé na konkrétní implementaci 3-D Secure. Pokud je to implementace společnosti Visa, tvoří komunikační doménu adresářový server, komerční certifikační autorita, Visa certifikační autorita, server historie autentizací a VisaNet.

Visa adresářový server je dotazován obchodníkem, zda dané číslo karty podporuje 3-D Secure platby. Pokud ano, adresářový server komunikuje se serverem pro řízení přístupu ve vydavatelské doméně a zjišťuje možnosti autentizace dané karty. Takto zjištěné informace vrací obchodníkovi.

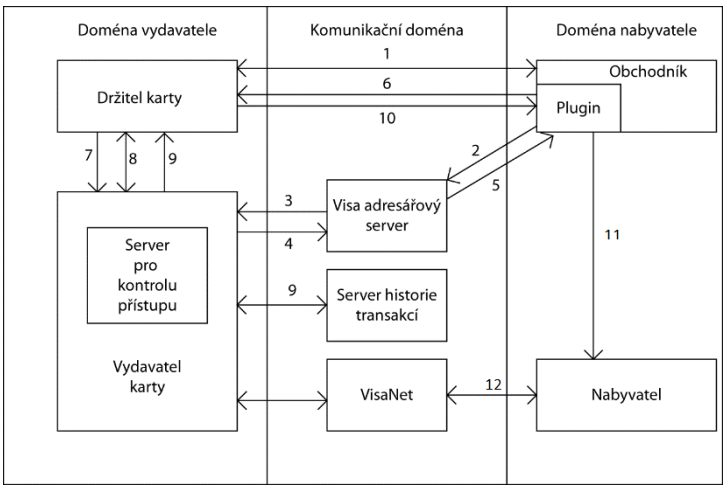
Komerční certifikační autorita umožňuje bezpečnou komunikaci tím, že generuje SSL/TLS klientské a serverové certifikáty. Podobně funguje i Visa certifikační autorita. Ta generuje certifikáty pro jednotlivé entity v 3-D Secure. Například podpisový certifikát nebo Visa kořenový certifikát.

Server historie autentizací slouží pro ukládání pokusů o autentizaci, ať už úspěšné, či neúspěšné. Informace získává od serverů pro kontrolu řízení přístupu. Uložená data jsou k dispozici jak doméně vydavatele, tak doméně nabyvatele a má sloužit pro řešení případných sporů.

Zprostředkovatelem transakcí mezi vydavatelskou a nabyvatelskou bankou je VisaNet. VisaNet také ukládá data týkající se veškerých transakcí, které se v systému vyskytly.

Průběh transakce

Finanční transakce uskutečněná pomocí 3-D Secure je tvořena z množství po sobě jdoucích kroků, znázorněných na obrázku 5.3.



Obr. 5.3 Průběh transakce v 3-D Secure sytému

1. Držitel platební karty si na internetových stránkách obchodníka, zapojeného v 3-D Secure, vybere zboží a uvede informace o své platební kartě.
2. Plugin na serveru obchodníka pošle číslo karty a další informace týkající se částky, měny apod. na Visa adresářový server.
3. Visa adresářový server vyhledá server kontroly řízení přístupu a dotáže se ho na definované způsoby autentizace pro danou kartu.
4. Server kontroly řízení přístupu odpoví nadefinovaným způsoby autentizace, které jsou nutné k autentizaci karty.
5. Visa adresářový server informaci o možnostech autentizací pošle na plugin internetového obchodu.
6. Plugin vyšle požadavek serveru pro kontrolu přístupu, na provedení autentizace držitele karty.
7. Server pro kontrolu řízení přístupu obdrží autentizační požadavek.
8. Na základě zjištěných autentizačních metod (znalost hesla, pinu, data expirace apod.) pro danou kartu je provedena autentizace.
9. Výsledek autentizace je odeslán pluginu obchodníka a zároveň je proveden záznam na serveru historie autentizací.
10. Plugin obdrží výsledek autentizace a zároveň zkontroluje digitální podpis připojený k této zprávě.
11. Obchodník zpracuje výměnu autorizačních zpráv s nabyvatelskou bankou.
12. Nabyvatelská banka pošle autorizační požadavek VisaNet, který transakci zpracuje.

6. ADRESÁŘOVÉ SYSTÉMY

Servery systémů řízení přístupu mohou obsahovat textové soubory se seznamy žadatelů, kterým bude umožněn. Pokud by však v systému bylo využito více serverů, nastal by problém s centralizovanou správou uživatelů a jejich práv.

Tento problém odpadá s využitím adresářového serveru nebo databáze. V adresářovém serveru můžeme uchovávat od jednoduchých uživatelských hesel až po certifikáty. Což nám poskytne všechny potřebná data pro dokončení jakéhokoliv typu autentizace. Kromě dokazovacího faktoru klienta můžeme mít uloženy i jeho přístupová práva. Čímž nám adresářový server obstará i autorizaci.

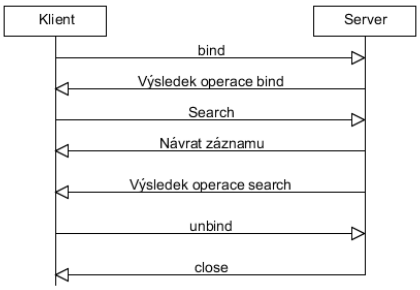
Jedním z často využívaných protokolů sloužících pro ukládání a přístup k datům uloženým na adresářovém serveru, je LDAP (Lightweight Directory Access Protocol). Informace pro následující kapitolu se opírají o tuto práci [13].

6.1. LDAP

LDAP je protokol optimalizovaný pro vyhledávání položek uložených na serveru, nikoliv pro časté zapisování. Data jsou uspořádávány do stromové struktury, která při správné implementaci umožňuje rychlé vyhledávání.

LDAP je zjednodušenou verzí protokolu X.500 a původně byl určen pouze pro komunikaci s ní. Postupem času se ale vyvinul v samostatný celek. Protokol X.500 kvůli své robustnosti využívají pouze ty největší telekomunikační společnosti. To LDAP se rozšířil mnohem více, našel si své místo ve spoustě větších firem.

Styl komunikace u LDAPu je klient-server. V případě řízení přístupu vystupuje v roli klienta autentizátor. Autentizátor obdrží od žadatele služby nějaký dokazovací faktor jehož pravost potřebuje ověřit. Proto odešle dotaz na LDAP server, který prohledá svou datovou strukturu a odpoví záznamem uloženým v databázi. Díky tomu je autentizátor schopný provést rozhodnutí o autentizaci klienta. Po provedení autentizace může následovat další dotaz na žadatelova práva.



Obr. 6.1 Průběh dotazu na LDAP server

Mezi klientem (např. RADIUS server) a LDAP serverem je navázáno TCP spojení a příkazem bind vykonána autentizace. O výsledku autentizace server klienta informuje. Klient si pak může vybrat dotaz Search (vyhledej) nebo Compare (porovnej). Podle zadaného dotazu prohledá server databázi a odešle výsledky svého hledání zakončené návratovou hodnotou. Po obdržení výsledků klient oznámí ukončení spojení zprávou unbind, načež server spojení uzavře.

7. INTEROPERABILITA PROTOKOLŮ

Interoperabilita je schopnost protokolů vzájemně spolupracovat, poskytovat si služby a dosáhnout vzájemné součinnosti. Dosáhnout nějaké vysoké úrovně interoperability mezi v současnosti využívanými protokoly řízení přístupu moc nelze. Protokoly pro svoji funkci totiž využívají různé druhy a formáty zpráv. Komunikace probíhají na jiných vrstvách referenčního modelu ISO/OSI. A především využívají pouze jediný přenosový scénář. Takže i přesto, že protokoly využívají principiálně podobných mechanizmů, ve své základní podobě spolupracovat nejsou schopny.

Neschopnost spolupráce protokolů se stává velmi limitující v momentě, kdy potřebujeme propojit více systému řízení přístupu dohromady. Proto vznikly snahy o vytvoření protokolu pro jednotnou implementaci různých metod řízení přístupu k aktivům. Jeden takový protokol navrhl doc. Ing. Karel Burda, CSc. nazvaný *ACP* (Access Control Protocol) definovaný v [14]. Je to univerzální dvoustranný protokol umožňující v rámci tzv. transakce dohodnout se na požadovaných aktivech a autentizační metodě. Po sjednání autentizační metody, autentizaci provést a v případě úspěchu doručit přístupové parametry.

7.1.ACP

Téměř každé zařízení v síti disponuje nějakými aktivy, ke kterým potřebuje žadatel získat přístup. V případě počítače je to jeho výpočetní výkon. Přístupové karty a telefony obsahují dokazovací faktor. I ten můžeme považovat jako aktivum v momentě, kdy ho potřebujeme dále využít k prokázání identity. Přístupové body nám svým rozhodnutím umožní nebo odepřou přístup k datům. A nakonec servery mají jako své aktivum určité poskytované služby. Je tedy jasné, že téměř každé zařízení musí být schopno rozhodovat o umožnění přístupu žadatele.

Kvůli tomu se jasně nabízí možnost implementovat do každého prvku sítě autonomní systém řízení přístupu (*AAA portál*). Ten v sobě zahrnuje funkci autority, kontroleru, autentizátoru a případně může i účtovatele. Autorita rozhoduje o povolení přístupu, kontrolér přístup umožňuje, autentizátor provádí autentizaci protistrany a svou vlastní. Účtovatel pokud je obsažen vede evidenci o přístupech k aktivům.

Dříve by byl velkým problémem výpočetní výkon nutný k běhu portálu například na čipové kartě. Výpočetní výkon však postupně stoupá a tak už by neměl být problém implementace portálu i do takových zařízení jako je mobilní telefon nebo přístupová karta. V budoucnu by se měly stát portály přímo součástí jádra operačních systému daného prvku.

Pro komunikaci mezi portály by se mělo využívat speciálního protokolu řízení přístupu (ACP – „Access control protocol“). Zprávy ACP jsou použitelné na všech síťových vrstvách referenčního modelu ISO/OSI. Primárně je pro jejich transport zamýšlen protokol TLS, EAPoL a rozhraní USB jeho však možné přenášet i přes ISO/IEC 14443. To například umožňuje bezproblémovou komunikaci mezi autentizačním tokenem připojeným k USB počítače, s přístupovým bodem sítě.

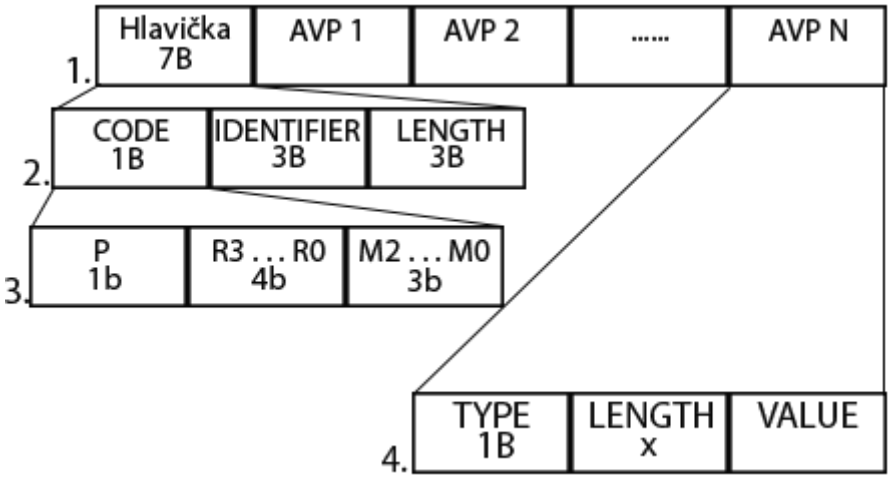
Protokol ACP je navržen jako dvoustranný, tzn. probíhá komunikace mezi portály *žadatele* a *poskytovatele*. V případě, že je nutné využít více prvků, lze využít sekvenčního zřetězení více běhů protokolu ACP nebo vložením dalšího běhu protokolu do již probíhajícího běhu. Sekvenční přístup se dá přirovnat k protokolu KERBEROS. Funguje totiž tak, že v jednom běhu protokolu získá dokazovací faktor, který mu poslouží v následujícím kroku. Vložení dalšího běhu si můžeme představit jako protokol RADIUS. Žadatel komunikuje

s prvkem, který inicializuje další běh ACP k jinému prvku. Typicky to může být žadatel komunikující s přístupovým bodem. Přístupový bod pak předává potřebná data přístupovému bodu.

7.1.1. ZPRÁVY ACP

Každá zpráva (1. řádek obrázku 7.1)protokolu ACP se skládá ze *záhlaví* a z N *attribute-value pairs* (AVP), kde N určuje počet AVP. V hlavičce (2. řádek) se nachází pole *IDENTIFIER*, které rozlišuje jednotlivé transakce probíhající portálem. Pole *LENGTH* určí celkovou délku zprávy. Poslední pole v hlavičce se dále dělí (3. řádek) na bit P, rezervní bity R0 – R3 a bity určující druh zprávy M2 – M0. Bit P určuje, zda-li se jedná o zprávu ACP (P=1) nebo zprávu protokolu EAP (P=0). Zprávy ACP jsou totiž téměř totožné se zprávami protokolu EAP.

Bity M2 – M0 určují druh zprávy. Protokol ACP definuje šest typů zpráv. K zahájení transakce je určena zpráva *START* (M = 000). K sjednání parametrů transakce jsou určeny zprávy *OFFER* (M = 001) a *SPECIFICATION* (M = 110). Pro autentizaci komunikujících stran je využita zpráva *REQUEST* (M = 101) a *RESPONSE* (M = 010). Celá transakce je nakonec uzavřena zprávou *FINISH* (M = 111). Kromě identifikace typu zprávy je pomocí bitu M0 určen i směr přenášené zprávy. Zpráv s M0 = 0 je zpráva putující od žadatele k poskytovateli a naopak zpráva s M0 = 1 je přenesena od poskytovatele k žadateli.



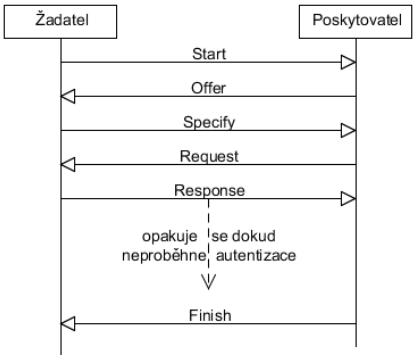
Obr. 7.1 Formát zpráv ACP

Attribute-value pairs je dvojice tvořená druhem zprávy (TYPE) a dat samotných (VALUE), jak je vidět na obrázku řádek 4. Délka pole VALUE může být buď kratší než 256B, pak hovoříme o krátkém AVP (SAVP). Tyto AVP jsou vhodné pro přenos krátkých bloků dat, jež by mohly využívat zařízení s omezeným výpočetním výkonem. Hodnota TYPE SAVP je v rozsahu 0 - 127. Dlouhé AVP (LAVP) má maximální délku 65 536B a je určeno pro přenos dlouhých bloků dat. Hodnota TYPE je v rozmezí 128 – 191. Posledním druhem je kontejnerové AVP (CAVP), které může obsahovat různá SAVP, LAVP případně i CAVP. Jediným limitem je maximální délka zprávy, která nesmí přesáhnout 2^16 B (65 536B).

7.1.2. AUTENTIZACE ACP

Autentizační metody a parametry nutné k autentizaci jsou sjednány během výměny zpráv OFFER a SPECIFICATION. Zprávou OFFER nabízí poskytovatel jím podporované metody autentizace a data k autentizaci nutné. Ve zprávě SPECIFICATION si pak žadatel volí jednu

z nabízených metod. V návrhu ACP jsou definovány dvě metody autentizace. První metodou je autentizace využívající některý z protokolů EAP. Tato autentizační metoda má hodnotu AVP TYPE = 32 a její hodnota (VALUE) specifikuje využitý EAP (např. pro EAP-TLS je to 13). Druhou metodou popsanou v ACP je „Lokální autentizační metoda“ jejíž AVP TYPE = 33. Její obsah je zcela ponechán na správci lokality.



Obr. 7.2 Průběh protokolu ACP

Konkrétní předpis pro výměnu zpráv protokolu ACP a stanovení obsahu zpráv se nazývá varianta protokolu ACP. Ve výchozím stavu využívá ACP variantu jednoduchých odpovědí. Tato varianta umožňuje snížení počtu přenášených zpráv. Pokud jsou ve zprávě START uvedeny potřebné informace, může poskytovatel vynechat zprávy sjednávání parametrů a přejít přímo k autentizaci případně rovnou ke zprávě FINISH. Kromě defaultní varianty existují ještě varianty globální a lokální. Globální varianta musí být definována nějakým RFC

Lze si však vytvořit i *vlastní lokální variantu protokolu*. V takové lokální variantě si může tvůrce definovat určité typy AVP dle vlastních potřeb. Rezervované typy SAVP jsou 96 - 127, LAVP jsou v rozsahu 176 – 191 a CAVP v rozsahu 240 – 255. Dané rozsahy typů AVP jsou tedy závislé na použité variantě protokolu.

7.2. NÁVRH KOMPLEXNÍHO AUTENTIZAČNÍHO PROTOKOLU

Jednotný autentizační protokol schopný autentizovat zařízení od čipových karet s poměrně malým výpočetním výkonem až po autentizaci žadatele přistupujícímu k serveru s důvěrnými daty zřejmě neexistuje. Předpokladem takového protokolu je možnost využití všech dostupných kryptogfarických primitiv. Pro zařízení s malým výpočetním výkonem by pro základní autentizaci bylo možné využívat hašovacích funkcí. Při větším výpočetním výkonu zařízení by bylo možné využívat symetrické kryptografie. A pro autentizaci s největším stupněm zabezpečení by se využívalo asymetrické kryptografie.

Autentizaci by měl začínat poskytovatel služby zprávou OFFER obsahující jím uznávané autentizační metody. Ty budou uváděny v seznamu CIPHER_SUITE. Vhodné by bylo přenést v úvodní zprávě i náhodné číslo poskytovatele. Náhodné číslo pak může být využito při výpočtu klíče spojení. Pokud poskytovatel nabízí metodu využívající digitálních podpisů, ve zprávě musí být obsažen blok SIGNATURE_AND_HASH_ALGORITHM, jehož součástí jsou dva seznamy HASH_ALGORITHM a SIGNATURE_ALGORITHM. Ty nabízejí poskytovatelem uznávané podpisové a hašovací algoritmy. Pokud poskytovatel nabídne v některé šifrovací sadě sjednání tajného klíč za pomocí eliptických křivek, je zapotřebí, aby zpráva OFFER byla rozšířena o zprávu NAMED_CURVE jež poskytuje listinu názvů podporovaných eliptických křivek. Parametry jednotlivých křivek jsou uvedeny dokumentu [15 stránky 26-40] nebo

[16]. Předem by mělo být definováno, který způsob pojmenování bude využit. Zda-li jména dle SECG, ANSI X9.62 nebo NIST. Parametry křivek jsou ekvivalentní, jména křivek se však liší podle toho, jakou organizací byly definovány. Druhým rozšířením při využití Eliptických křivek by mělo být EC_POINT_FORMAT. Ten definuje formát bodů eliptických křivek. Může být buď bez komprese, kdy je ponechán původní formát dat. Případně data mohou projít kompresí. Souřadnice x zůstavují beze změny, ale souřadnice y obsahuje pouze jeden bit. K tomu lze využít metody ANSI_X_962_COMPRESSD_PRIME nebo ANSI_X_962_COMPRESSED_CHAR2.

```
struct {
       CIPHER_SUITE list šifrovacích sad;
       RAND_PRO;
       IDENTITY_PRO;

       Case Eliptické křivky:

              NAMED_CURVE list názvů eliptických křivek;
              EC_POINT_FORMAT list podporovaných formatů eliptických křivek;

       Case Podpis

              SIGNATURE_ALGORITHM list podpisových algoritmů;
              HASH_ALGORITHM list podporovaných hašů;
}OFFER
```

7.2.1. AUTENTIZACE HAŠOVÁNÍM

Autentizace hašováním nastane v případě, že si žadatel vybere z nabízených sad některou využívající hašování sdíleného tajemství. Jednocestnou funkci můžeme aplikovat buď pouze na sdílené heslo, případně na heslo s přijatou výzvou. Jako výzvu můžeme využít hodnotu RAND. Pro hašování se bude využívat hašovacích funkcí MD5 (délka obrazu zprávy 128b), SHA-1 (160b), SHA-2 (224, 256, 384, 512b) i nejnovějšího standardu SHA-3 (224, 256, 384, 512b).

Aby poskytovatel mohl rychleji vyhledávat v databázi, je žádoucí, aby žadatel odeslal svoji identitu v poli IDENTITY. Touto hodnotou může být např. UID karty, uživatelské jméno, email.

Autentizace nevyužívající výzvy provede pouze ověření identity žadatele poskytovateli. Kdežto autentizace hašující i výzvy umožní oboustrannou autentizaci. Žadatel totiž vytvoří hash z přijaté hodnoty RAND_PRO, jím vygenerované hodnoty RAND_SUP a sdíleného klíče (h = H(K), kde K = (RAND_PRO, RAND_SUP, secret)). Pro ověření identity poskytovatele, stačí pak vytvořit haš z RAND_SUP a sdíleného tajemství secret. Výsledný haš bude přenášen v poli HASH_VALUE.

Po nabídce OFFER tedy bude následovat zpráva SPECIFY. Jejímž obsahem bude:

```
Case CIPHER SUITE autentizace hašováním{
       CIPHER_SUITE;
       IDENTITY_SUP;
       HASH_VALUE ;
       Case hašování výzvy
              RAND_SUP
}SPECIFY
```

Poskytovatel po přijetí zprávy ověří hodnotu z pole HASH_VALUE. Pokud haš odpovídá hodnotě uložené u dané identity v databázi je autentizace žadatele úspěšná. Pokud se vyžaduje oboustranná autentizace je třeba přenosu ještě jedné zprávy. Zpráva REQUEST je odeslána poskytovatelem a obsahuje jím vypočítaný haš.

```
Case CIPHER SUITE autentizace hašováním {
      HASH_VALUE;
}REQUEST
```

Díky této zprávě může být ověřena identita poskytovatele. Tím je autentizace dokončena a může proběhnout zbytek protokolu řízení přístupu.

7.2.2. VYUŽITÍ PŘEDEM SDÍLENÉHO KLÍČE K SYMETRICKÉMU ŠIFROVÁNÍ

V momentě, kdy nejsme tolik omezeni výpočetním výkonem, a obě strany disponují sdíleným tajemstvím, můžeme využít symetrického šifrování jak pro autentizaci, tak pro následně probíhající komunikaci. Přestože by šlo pro šifrování využít onoho sdíleného klíče, nebylo by to příliš bezpečné řešení. Během autentizace si tedy musíme odvodit klíče daného spojení.

To by šlo provést podobně tomu, jak jsou klíče odvozeny v protokolu EAP PSK. Algoritmem AES bychom zašifrovali předem dohodnutý řetězec (např. string „0“) klíčem sdíleného tajemství. Vzniklý řetězec bychom rozšířili a následně rozdělili na dva klíče (AK a KDK). Klíč *AK* bychom využili k vzájemné autentizaci. Klíč *KDK* by potom sloužil k vytvoření klíče *TK*, sloužícího pro šifrování následné komunikace.

Pokud si žadatel vybere z kryptografických sad autentizaci a vytvoření šifrovacích klíčů pomocí symetrické kryptografie, musí vypadat komunikace následovně. Na zprávu OFFER odpoví žadatel zprávou SPECIFY. Ta obsahuje pole svého náhodného čísla RAND_SUP. Dále bude obsahem zprávy SPECIFY pole HMAC sloužící k autentizaci žadatele. Jeho obsahem je vypočítaná pečeť P. Pečeť P = Q (Z, AK) se vytvoří zašifrováním zprávy Z algoritmem AES pomocí odvozeného klíče AK a následným zahešováním výsledku pomocí hašovací funkce nabídnuté ve zprávě OFFER. Kde Z jsou zřetězená pole IDENTITY_PRO ‖ IDENTITY_SUP ‖ RAND_PRO ‖ RAND_SUP. Poskytovatel provede vlastní výpočet pečeti P, kterou pak porovná s přijatou pečetí P’. Tím, že klíč AK by měl být schopný odvodit pouze majitel předem sdíleného klíče, může poskytovatel považovat při rovnosti P = P’ identitu žadatele za ověřenou.

```
Case CIPHER SUITE předsdílený klíč{
      CIPHER_SUITE;
      HASH_ALGORTIHM
      IDENTITY_SUP;
      HMAC ;
      RAND_SUP;
}SPECIFY
```

Následuje zpráva REQUEST, která umožní autentizaci poskytovatele, tím že obsahuje také pole HMAC. Tentokrát v něm je obsažená pečeť vytvořena ze zřetězených hodnot IDENTITY_PRO ‖ RAND_PRO opět zašifrovaných pomocí klíče AK a výsledek zahešován. Tím by byla obstarána oboustranná autentizace. V případě požadavku na další šifrované spojení je nutné ještě ověřit, zda obě komunikující strany disponují klíčem TK, sloužícím k šifrování veškerých následně přenášených dat v transakci. Pokud budeme opět využívat mechanizmus podobný EAP PSK, TK odvodíme tak, že zašifrujeme RAND_SUP klíčem KDK odvozeným na začátku.

Pole PCHANNEL_PRO obsahuje zašifrovanou zprávu pomocí klíče *TK*. Zpráva v sobě nese informaci o výsledku autentizace. Druhou součástí je TAG, jehož obsahem je MAC zašifrované zprávy.

```
Case CIPHER SUITE předsdílený klíč{
      HMAC
      PCHANNEL_PRO
                        Zašifrovaný výsledek autentizace;
                        TAG;
}REQUEST
```

V posledním kroku prokáže znalost klíče *TK* i žadatel. Znalost prokáže odesláním zprávy RESPONSE jejíž součástí bude pole PCHANNEL_SUP. Obsahem pole bude opět zašifrovaný text s výsledkem autentizace a k němu přiložená MAC.

```
Case CIPHER SUITE předsdílený klíč{
      PCHANNEL_SUP
                        Zašifrovaný výsledek autentizace;
                        TAG;
}REQUEST
```

7.2.3. AUTENTIZACE S VYUŽITÍM ASYMETRICKÉ KRYPTOGRAFIE

Problémem asymetrických kryptosystémů je pomalé šifrování a dešifrování. Podle testu uvedenému v [17] je šifrování stejně velkých souborů pomocí AES v průměru čtyři krát rychlejší než pomocí RSA. Proto se jejich silné zabezpečení využívá pouze pro sjednání klíčů sloužící symetrickým kryptosystémům nebo pro podepsání zpráv. Ke sjednání klíčů můžeme využít tří kryptosystémů. *RSA*, *Diffie-Hellman* a *Diffie-Hellman s využitím eliptických křivek*. Návrh této části využívá mechanismů využívaných protokolem EAP-TLS [3].

RSA

Při využití RSA výměny klíčů poskytovatel nabídne žadateli svůj veřejný klíč. Pomocí tohoto veřejného klíče zašifruje žadatel nějaké sdílené tajemství. Jediným kdo takovou zprávu může dešifrovat, je poskytovatel pomocí svého soukromého klíče.

Pokud k výměně klíčů využijeme metodu RSA, je třeba, aby zpráva OFFER obsahovala kromě nabídky šifrovací sady využívající RSA výměny klíčů i pole SIGNATURE_ALGORITHM spolu s HASH_ALGORITHM. Na takovéto zahájení autentizace odpoví žadatel zprávou SPECIFY, která pouze vybírá šifrovací sadu a podpisové algoritmy.

```
struct{
      CIPHER_SUITE;
      IDENTITY_SUP;
      RAND_SUP;
      SIGNATURE_ALGORITHM;
      HASH_ALGORITHM;
}SPECIFY
```

Jakmile poskytovatel přijme zprávu SPECIFY vyžadující autentizaci za pomoci RSA, odešle zprávu RESPONSE. Jejím obsahem bude nejméně jeden certifikát. Certifikát poskytovatele bude uveden jako první. Za ním pak mohou následovat certifikáty certifikačních autorit. V certifikátu musí být veřejný klíč poskytovatele, který lze využívat pro šifrování. Certifikát musí obsahovat podpis vytvořený pomoci algoritmů definovaných ve zprávě SPECIFY.

V případě, že vybraná autentizační metoda potřebuje ověřit i identitu žadatele je přidáno do zprávy REQUEST pole CERT_REQ. To slouží k vyzvání žadatele, aby do další zprávy přiložil svůj certifikát. RSA veřejný klíč musí být využitelný pro podepisování zpráv.

```
Case CIPHER SUITE RSA{
      CERT_PRO;
      CERT_REQ;
}REQUEST
```

Žadatel díky přijatému certifikátu disponuje veřejným klíčem poskytovatele. Pomocí tohoto klíče je schopný šifrovat zprávy, které bude schopný dešifrovat pouze poskytovatel. To lze využít k vytvoření společného klíče.

Žadatel si vygeneruje dostatečně dlouhé náhodné číslo, které bude využito pro další odvozování klíčů (Pre master secret). Z náhodného čísla vytvoříme kryptogram C = (N, VK_P). Ten můžeme přenést, aniž by ho mohl kdokoliv, kromě poskytovatele, dekódovat.

V případě, že si poskytovatel vyžádal certifikát, musí ho žadatel do své zprávy přiložit.

```
Case CIPHER SUITE RSA{
      CERT_SUP;
      ENCRYPTED_PRE_MASTER;
}RESPONSE
```

Diffie-Hellmanova výměna klíčů

Diffie-Hellman (DH) protokol umožní dohodnout tajný klíč přes nezabezpečený kanál, aniž by tajný klíč byl přenášen po lince. Případný útočník by k zjištění společného tajemství musel vyřešit problém diskrétního logaritmu. K tomu aby obě strany byly schopny vypočítat společné tajemství, je pouze nutné sdílet dostatečně velké prvočíslo p, generátor g a veřejný klíč protistrany.

Pokud je v nabídce šifrovací sada, která k ustálení klíčů využívá DH, musí byt použit následující průběh komunikace. Jako klasicky v první zprávě SPECIFY vybere žadatel šifrovací sadu, a pokud jsou nabízeny, algoritmy využité k podepisování zpráv.

```
 Struct {
      CIPHER_SUITE;
      IDENTITY_SUP;
      RAND_SUP;
      SIGNATURE_ALGORITHM;
      HASH_ALGORITHM;
}SPECIFY
```

Po přijetí této zprávy odpoví poskytovatel svým certifikátem, který obsahuje podle předchozí dohody buď RSA nebo DSS podpisový klíč. Podpisový klíč bude sloužit k podepsání přenášených parametrů sloužících pro DH výpočty. Existuje i varianta, kdy se komunikující strany dohodli na tom, že přenášené parametry nemusí být podepsány (SIGNATURE_ALGORITHM a HASH_ALGORITHM = NULL). Tato varianta autentizace ale nezabraňuje útoku typu man in the middle.

Ve zprávě si může poskytovatel vyžádat certifikát žadatel za pomoci pole CERT_REQ do kterého uvede jím uznávané certifikáty, podpisové metody a název certifikační autority.

```
Case CIPHER SUITE D-H {
       CERT_PRO;
       CERT_REQ;
       KEY_PRO {
       case DH_anon {
              dh_p;
              dh_g;
              dh_VK_PRO;}

       case DH_RSA, DH_DSS

              digitálně podepsané{
                           dh_p;
                           dh_g;
                           dh_VK_PRO;}}
}REQUEST
```

K úspěšnému výpočtu sdíleného tajemství potřebujeme ještě přenést veřejný klíč žadatele. Ten pro zvolení veřejného klíče musí využít hodnoty zvolené poskytovatelem uvedené v poli KEY. Veřejný klíč žadatele může být buď obsažen přímo v certifikátu (RSA_fixed_DH, DSS_fixed_DH), pak se do pole KEY_SUP zapíše řetězec implicit, který značí uložení VK v certifikátu. Pokud ale je odeslaný certifikát určený k podpisům vytvoří se digitální podpis veřejného klíče a ten se odešle.

```
Case CIPHER SUITE D-H: {
       CERT_SUP;
       KEY_SUP {
       case CERT_SUP = RSA_fixed_DH, DSS_fixed_DH {
              implicit;}

       case CERT_SUP = RSA_sign, DSS_sign
              digitálně podepsané{
                     dh_VK_SUP;}}
}RESPONSE
```

Po takto proběhlé komunikaci disponují obě komunikující strany dostatečnými znalostmi k tomu, aby mohli vypočítat totožný sdílený klíč. Jak je z předcházejících zpráv vidět, klíč nebyl nikdy přenesen a tudíž ani nemohl být odposlechnut.

Diffie-Hellmanova výměna klíčů s využitím eliptických křivek

Diffie-Hellmanova výměna klíčů s využitím eliptických křivek (EC DH) místo problému řešení diskrétního logaritmu využívá problému sčítání bodů eliptické křivky. Výhodou kryptografie využívající eliptických křivek oproti jiným asymetrickým kryptografiím, je mnohem menší klíč při zachování stejné bezpečnosti. V dokumentu NSA [18] se uvádí, že stejné bezpečnosti dosahují klíče RSA o velikosti 2048b jako klíče EC o velikosti 233b.

V případě, že poskytovatel nabídne možnost šifrovací sady využívající ECDH je nutné do zprávy OFFER nabídnout seznam podporovaných křivek NAMED_CURVE a jejich kompresní formát EC_POINT_FORMAT.

Odpovědí na zprávu OFFER bude podobná jako u předchozích metod autentizace. Tentokrát pouze rozšířená o výběr NAMED_CURVE a EC_POINT_FORMAT.

```
Struct {
       CIPHER_SUITE;
       IDENTITY_SUP;
       RAND_SUP;
       SIGNATURE_ALGORITHM;
       HASH_ALGORITHM;
       NAMED_CURVE;
       EC_POINT_FORMAT;
}SPECIFY
```

Po vybrání šifrovací metody a eliptické křivky, nad kterou bude prováděn výpočet, následuje odeslání podpisového certifikátu (ECDH_ECDSA nebo ECDH_RSA). Dále je odeslán klíč obsahující parametry eliptické křivky. V případě, že byla vybrána konkrétní eliptická křivka pojmenovaná dle [16], stačí do klíče vložit pouze název křivky. Parametry těchto křivek jsou již předem definovány. Pokud se ale využívá některého kompresní formát, musí být přeneseny všechny parametry.

Pokud byl zvolen kompresní formát explicit_prime je třeba přenést prvočíslo p definující těleso. Koeficienty a a b specifikující křivku E, bod G na eliptické křivce, jeho řád n a kofaktor h.

Pokud se využívá kompresního formátu explicit_char2. Stejně jako v předchozím se přenáší a, b, E, G, n a h. Místo prvočísla p je ale nutné přenést stupeň charakteristického dvourozměrného pole. Dále buď k nebo k1, k2, k3 podle vybrané varianty typu základu.

```
 Case CIPHER SUITE EC D-H {
       CERT_PRO;
       CERT_REQ;
       KEY_PRO { digitálně podepsané{
              EC_VK_PRO
              case NAMED_CURVE {
                     NAMED_CURVE}

              case explicit_prime
                     {prime_p (p);
                     ECCurve (a, b);
                     ECPoint (G);
                     Order (n);
                     Cofactor (h);}

              case explicit_char2
                     {m;
                     ECCurve (a, b);
                     ECPoint (G);
                     Order (n);
                     Cofactor (h);}
                     ECBasisType
                            case trinomial{
                                   k;}
                            case trinomial{
                                   k1;
                                   k2;
                                   k3;}}}
}REQUEST
```

Poslední zprávou nutnou k ustálení klíčů je zpráva RESPONSE, jež v sobě nese certifikát žadatele, byl-li poskytovatelem vyžádán. Pokud byl odeslán certifikát obsahující

veřejný klíč použitelný nad sjednanou eliptickou křivkou, obsahuje pole KEY_SUP hodnotu implicit. Pokud je využit pouze podpisový certifikát, podepíše se zvolený veřejný klíč a odešle v poli KEY_SUP.

```
Case CIPHER SUITE EC D-H {
      CERT_SUP;
      KEY_SUP {
      Case CERT_SUP = ECDSA_fixed_ECDH, RSA_fixed_ECDH {
            implicit;}

      case CERT_SUP = ECDSA_sign
            digitálně podepsané {
                  EC_VK_SUP;}}
}RESPONSE
```

Po takto proběhlé komunikaci disponují obě strany vypočítaným sdíleným tajemstvím, jež se může použít pro tvorbu klíčů sloužících symetrické kryptografii.

Symetrické šifrování

Po úspěšné autentizaci a dohodnutí se na sdíleném klíči je možné přejít k symetrickému šifrování dalšího spojení. Vypočítaný klíč je třeba rozšířit. Toho můžeme dosáhnout buď stejně jako u před sdíleného klíče. Případně můžeme využít *pseudonáhodné funkce*, které bude sloužit jako vstup sdílené tajemství, RAND_SUP a RAND_PRO. Data generujeme tak dlouho, dokud nemáme zajištěno dostatečné množství materiálu pro všechny potřebné klíče a inicializační vektory. Měly by být vytvořeny dva klíče pro šifrování spojení (každý pro šifrování jednoho směru komunikace), dva klíče pro vytváření MAC k ověření autentičnosti a dva inicializační vektory. Pomocí šifrovacího klíče je zašifrován výsledek autentizace a vypočítán MAC zprávy, který se přenáší k protistraně ve zprávě PCHANNEL.

Následná komunikace může být zabezpečena buď blokovými, nebo proudovými šiframi. A jejich klíč by měl být těžce zjistitelný.

7.2.4. ZPRÁVY VYUŽITÉ PROTOKOLEM

Popis jednotlivých zpráv byl rozepsán na předchozích stranách. V této kapitole si shrneme všechny použité zprávy protokolem. Ten musí být schopen přenést následující zprávy:

Tab. 7.1 Souhrn využitých zpráv navrhnutého autentizačního protokolu

Zpráva	Stručný popis
CIPHER_SUITE	Nabídka/výběr šifrovací sady
RAND_PRO, RAND_SUP	Náhodné číslo poskytovatele/ žadatele
IDENTITY_PRO, IDENTITY_SUP	Identita poskytovatele/ žadatele
HASH_VALUE	Haš vytvořený z nějakého tajemství
MAC_PRO, MAC_SUP	Slouží k ověření znalosti MAC klíče
PCHANNEL_SUP, PCHANNEL_PRO	Slouží k ověření znalosti šiforvacího klíče
CERT	Obsahuje list certifikátů, kde se jako první nachází klíč samotné entity. Teprve poté je následován klíči CA
CERTREQ	Požadavek na protistranu, aby předložila certifikát
KEY_PRO, KEY_SUP	Přenos parametrů sloužících pro výpočet klíče
HASH_ALGORITHM	Nabídka/výběr hašovacích sad sloužících pro vytvoření digitálních podpisů
SIGNATURE_ALGORITHM	Nabídka/výběr algoritmů, jež se budou využívat pro podepisování zpráv

7.3. IMPLEMENTACE VYTVOŘENÉHO PROTOKOLU DO ACP

Abychom mohli výše popsaný protokol implementovat do ACP, musíme nejdříve vytvořit variantu protokolu, která nám umožní si definovat vlastní AVP. Takovou lokální variantu si můžeme pojmenovat AUTENTIZACE. Její použití iniciujeme nabídkou AVP typu 35 (Lokální varianta protokolu) s přenášenou hodnotou AUTENTIZACE ve zprávě OFFER.

Kromě využití lokální varianty protokolu musíme uvést, že budeme využívat i lokální autentizační protokol. Toho docílíme odesláním AVP typu 33 (Lokální autentizační metoda) s hodnotou VNITRNI_AUTENTIZACE.

V následující tabulce si uvedeme všechny AVP, které budeme využívat. Některé je možné využít z již definovaných, jiné si musíme vytvořit.

Tab. 7.2 AVP využívané navrženým autentizačním protokolem

Název	Význam	AVP	Typ Value
NAME_PRO_L	AVP schopné přenést IDENTIY_PRO	4	Text
NAME_SUP_L	AVP schopné přenést IDENTITY_SUP	5	Text
PMS	Pre Master Secret	49	String
HMAC	Pole pro přenos pečeti MAC_SUP a MAC_PRO	50	String
CIPHER_SUITE	Nabídnutá/vybraná šifrovací sada	96	String
RAND_PRO	Náhodné číslo, vytvořené poskytovatelem	97	String
RAND_SUP	Náhodné číslo vytvořené žadatelem	98	String
NAMED_CURVE	Nabídnutá/vybraná eliptická křivka	101	String
EC_POINT_FORMAT	Formát využité eliptické křivky	102	String
SIGNATURE_ALGORITHM	Nabídnutý/vybraný podpisový algoritmus	103	String
HASH_ALGORITHM	Nabídnutý/vybraný hašovací algoritmus	104	String
HASH_VALUE	Zahešovaná hodnota (např. heslo)	105	String
CERT_REQ	Požadavek k vydání certifikátu protistranou	106	String
DH_P	Přenáší hodnotu p pro výpočet D-H	107	String
DH_G	Přenáší hodnotu g pro výpočet D-H	108	String
DH_VK	Veřejný klíč D-H	109	String
KEY	Oznamuje, přítomnost veřejného klíče v certifikátu	110	String
EC_VK	Veřejný klíč využitý DH-EC	111	String
PRIME_P	Prvočíslo definující těleso	112	String
EC_CURVE	Zřetězené konstanty a a b.	113	String
EC_POINT	Bod G na eliptické křivce	114	String
ORDER	Řád bodu na eliptické křivce	115	String
COFACTOR	Kofaktor h	116	String
M	Stupeň charakteristického pole	117	String
EC_BASIS_TYPE	Trinomial/ Pentamonial	118	String
K	Exponent K při Trinomiálním vyjádření EC	119	String
K1	Exponent K_1 při Pentanomiálním vyjádření EC	120	String
K2	Exponent K_2 při Pentanomiálním vyjádření EC	121	String
K3	Exponent K_3 při Pentanomiálním vyjádření EC	122	String
CERT	Obsahuje certifikáty dle X.509	160	String
PSS	Obsahuje digitální podpis řetězce	163	String

PCHANNEL	Slouží k ověření klíče protistrany	176	String
SIG	Kontejner obsahující nějakou hodnotu AVP spolu s podpisem PSS	226	-
KEY_VALUES	Kontejner nesoucí AVP nutné pro výpočet sdíleného tajemství	240	-
VK	Kontejner přenášející AVP veřejný klíč	241	-

Pro přenos hodnot nutných k výpočtu sdíleného tajemství (např. DH_P, DH_G, PRIME_P apod.) budeme využívat KEY_VALUES. Druhým využívaným kontejnerem je VK. To nese kombinaci KEY (implicit/explicit) a je-li vyžadován i VK.

Pro většinu využitých zpráv by měla svou velikostí stači SAVP s velikostí 2048b. Výjimku, pro kterou jsem zvolil LAVP jsou certifikáty, které v případě přiložení několika certifikátů mohou dosáhnout větší velikosti. Jako LAVP bylo definováno i PSS nesoucí zprávu spolu s podpisem. Z důvodu velikosti byl mezi LAVP zařazeno i PCHANNEL nesoucí zašifrovaný řetězec prokazující znalost klíče.

7.4. SHRNUTÍ

Takto vytvořené *AVP* by měli být schopny vyměňovat všechny potřebné informace pro úspěšný průběh širokého spektra autentizací. Protokol ACP s implementací této lokální autentizační metody by byl schopný pracovat jak s paměťovými kartami, tak se zařízeními v počítačové síti.

Druh autentizace by byl plně závislý na nabídce podporovaných algoritmů poskytovatelem. V závislosti na bezpečnosti provedené autentizace by poskytovatel mohl upravovat i přístupová práva udělená žadateli.

8. KLADY A ZÁPORY PROTOKOLŮ

Na následujících řádcích budou shrnuty dříve vyjmenované klady a zápory různých mechanismů využívaných při řízení přístupu.

Základem každého systému řízení přístupu je autentizace. Autentizační mechanizmy můžeme rozdělit podle toho, jakou kryptografickou metodu využívají k zašifrování dokazovacího faktoru, tak aby nebyl schopný dokazovací faktor získat případný útočník. Nejjednodušší způsob skrytí dokazovacího faktoru je vypočítání haše dokazovacího faktoru. Další variantou jak skrýt dokazovací faktor je šifrování celé probíhající komunikace za pomocí některé ze symetrických šifer. Poslední využívaným způsobem přenosu dokazovacího faktoru je za pomoci asymetrické kryptografie.

8.1. HAŠOVÁNÍ

Autentizaci s využitím hašování využívají v této práci například protokoly RKKW, EAP-MD5 nebo Hash lock. U těchto autentizačních protokolů je velkou výhodou, když nehešují pouze dokazovací faktor, ale zřetězí k němu i náhodný řetězec. Odesílaný haš komunikačním kanálem se tak neustále mění a znesnadňuje to i případný útok. V případě, kdy haš vytváří obě komunikující strany, můžeme docílit i velice žádané vzájemné autentizace.

Bezpečnostním problémem se stává prolomení některých hašovacích algoritmů. Jako potenciálně nebezpečné se v současné době dají považovat algoritmy MD5 a SHA-1. V základní podobě útoku hrubou silou by bylo potřeba k prolomení MD5 2^112 provedených výpočtů. Na základě publikovaných útoků se toto číslo snížilo na 2^51 provedených výpočtů. Ke zcela bezpečnému přenosu dokazovacího faktoru by tedy bylo ideální využívat protokolů SHA-2 a SHA-3. Po skončení oficiální podpory Windows XP by již nemělo nic bránit masivnímu nasazení SHA-2. V základní podobě totiž Windows XP nasazení SHA-2 nepodporovalo.

Autentizace s využitím starších hašovacích algoritmů bude ještě delší dobu využívána u přístupových karet. Při řízení přístupu paměťovými kartami se velmi oceňuje malá výpočetní náročnost.

8.2. SYMETRICKÁ KRYPTOGRAFIE

Zástupci autentizačních metod, využívajících symetrické kryptografie, jsou protokoly EAP PSK a protokoly dle ISO/IEC 9798-2. K ověření identity komunikující protistrany stačí to, že je schopná komunikovat pomocí sdíleného šifrovacího klíče.

K udržení šifrovacího klíče v tajnosti před případným útočníkem je nutné, aby došlo k oboustranné autentizaci. Pokud totiž proběhne pouze jednostranná autentizace, tak jak je stanoveno v ISO/IEC 9798 může být využito útoku typu *chosen plaintext attack.* Tento útok se zakládá na tom, že útočník může posílat žadateli libovolné výzvy, které žadatel zašifruje a odešle zpět. Útočník tak získá svůj vlastní (nešifrovaný) text a k němu odpovídající zašifrovaný text, pomocí něhož může získat šifrovací klíč.

Další podmínkou k bezpečné autentizaci je dostatečně dlouhý klíč. Tato podmínka se zatím vztahuje pouze k protokolu DES, který má délku klíče 64 bitů, z toho pouze 56 efektivních, což se považuje jako nedostačené.

Autentizace s využitím symetrické kryptografie je masivně využívána přístupovými kartami MIFARE. Ty jsou nasazovány v mnoha městech (Praha OpenCard, Londýn Oyster apod.) k nákupu lístků na hromadnou dopravu nebo přístupu do knihoven.

Díky využití techniky postranních kanálů, nejsou v bezpečí ani tyto karty. Vědci z německé univerzity Ruhr, publikovali v roce 2011 útok na kartu MIFARE DESFire (využita jako OpenCard), schopnou šifrovat algoritmy AES a 3DES. Získání klíče netrvá déle než 7 hodin a to s vybavením za cenu okolo 50 000Kč.

8.3. ASYMETRICKÁ KRYPTOGRAFIE

Asymetrickou kryptografii využívají protokoly v několika podobách. V případě přítomnosti certifikační autority je možné využívat digitálních certifikátů nesoucích veřejné klíče. Pomocí veřejných klíčů se následně komunikace šifruje a jediným, kdo může zprávy dešifrovat, je vlastník privátního klíče náležejícího k veřejnému klíči. Tohoto postupu využívá vzájemná autentizace popsaná v ISO/IEC 9798-3.

V případě zašifrování celého průběhu autentizace asymetrickými šiframi bychom zřejmě dosáhli největší bezpečnosti, ale také zároveň by autentizace trvala neúnosně dlouho. Z toho důvodu bývá asymetrické kryptografie využito ke sjednání klíče následně využitého pro symetrické šifrování. Asymetrická kryptografie je také často využívaná pro ověření autentičnosti zprávy, pomocí podepsání privátním klíčem.

Jakmile autentizace nevyužívá certifikační autority a v systému není nikdo, kdo by garantoval identitu komunikujících stra, hrozí útok typu Man in the middle. Při tomto typu útoku útočník zachytává probíhající komunikaci a modifikuje odesílané veřejné klíče tak, aby byl schopný svým privátním klíčem dešifrovat komunikaci.

Pokud je využita asymetrická kryptografie na přístupových kartách, často se využívá výhody asymetrické kryptografie s využitím eliptických křivek. Ty totiž ke stejné bezpečnosti jakou poskytuje například RSA, potřebují značně kratšího klíče, což umožní urychlit šifrování a množství přenášených dat.

8.4. AUTORIZACE

Jakmile proběhne ověření identity, musí být přístupovým systémem vyhledány přístupová práva daného žadatele, tak aby autorita mohla rozhodnout o umožnění přístupu. Databáze uživatelů, jejich dokazovací faktory a přístupová práva se ukládají buď do textových souborů, na LDAP servery případně do různých databází. Výhodou LDAP a databázových serverů je to že k nim může přistupovat několik systémů řízení přístupu současně a tak se centralizuje správa uživatelů.

Samotná autorizace probíhá tak, že autoritou je vyslán dotaz obsahující identitu žadatele získanou autentizací směrem k serveru. Server prohledá databázi, a pokud nalezne shodu, odešle autoritě přístupová práva přiřazené danému žadateli. Díky přijatým informacím autorita rozhodne a instruuje bránu, zda umožnit nebo odepřít přístup.

9. VÝUKOVÝ SOFTWARE

Součástí práce je výukový software, sloužící studentovi pro správné a jednoduché pochopení protokolů řízení přístupu. Výukový materiál je spustitelný ve webovém prohlížeči, tudíž při nahrání na některý ze serverů by měl být snadno dostupný na jakémkoliv počítači.

Celá tato práce slouží jako zdroj materiálu pro webové stránky. Čím však stránky práci rozšiřují, je obsah názorných animací průběhu protokolů řízení přístupu. Výhodou animace je názorné vyobrazení přenášených dat, na kterých je možné popsat konkrétní výhody a nevýhody protokolů. Kromě samotných animací by měl výukový materiál přispět k pochopení principů využívaných při řízení přístupu.

Na závěr každé kapitoly se nachází test, díky kterému si student může ověřit nastudované znalosti. Testové otázky je možné libovolně modifikovat díky jejich uložení v *XML souboru*.

Materiál využívá kombinace kódu jazyka HTML spojeného s CSS a názorné animace jsou vytvořeny v grafickém vektorovém programu Flash. Interaktivity animací je dosaženo za pomocí jazyka *ActionScript 3.0*. ActionScript (AS) je objektově orientovaný programovací jazyk vycházející z jazyka JavaScript. AS ve verzi 3.0 umožňuje několikanásobně rychlejší zkompilování kódu oproti verzi 2.0. Další výhodou verze 3.0 je lepší práce s XML soubory. Práce s XML soubory je využita v softwaru při načítání testových otázek.

Animace a testy jsou do materiálu umístěny ve formátu .swf. K tomu, aby mohl být tento formát korektně zobrazen, je vyžadován Flash Player. Navíc je nutné aby Flash Player byl ve verzi vyšší než 8. To je zapříčiněno využitím AS 3.0, který je podporován až od verze Flash Player 9.

Výukový materiál je spustitelný souborem index.html umístěným na přiloženém CD ve složce Access_control. Otevřením souboru index ve webovém prohlížeči se dostaneme na titulní stranu výukového materiálu jak je vidět na obrázku A.1 umístěného v příloze A.

Z úvodní stránky je možné přejít do jednotlivých kapitol (Řízení přístupu, Biometrika, Přístupové karty a NFC, Bankovnictví a Počítačové sítě). Kde si čtenář bude moci prohlédnout přenášená data, průběh protokolů apod. Na obrázku A.3 je konkrétně znázorněna stránka pojednávající o protokolu Needham-Schroeder.

Po prostudování každé kapitoly si může student ověřit svoje nabyté znalosti krátkým testem, jak je vidět na obrátku A.2.

Korektní zobrazení stránek bylo ověřeno na operačních systémech Linux, MacOS a Windows. Při rozlišení větším než 1024x768 se nevyskytovali žádné problémy u prohlížečů, které využívá 98% uživatelů dle [19]. Konkrétně Internet Explorer od verze 8, Chrome, Mozilla Firefox, Opera a Safari.

10. ZÁVĚR

Protokolů řízení přístupu je celá řada. V této práci některé reprezentativní systémy byly vybrány tak, aby se dali demonstrovat klady a zápory, které obsahují.

Tím, že řízení přístupu je využíváno v mnoha oblastech, práce se dělí do kapitol. První kapitola přímo zaměřená na protokoly řízení přístupu pojednává o počítačových sítích. Kde hlavními zástupci protokolů je Kerberos, TACACS+, Radius a Diametr. Další kapitoly rozebírají využití přístupových karet, NFC zařízení a na závěr kapitola popisující řízení přístupu v bankovnictví.

S tím jakou požadujeme bezpečnost od jednotlivých systémů, se odvíjí i jejich cena. Na počátku zavádění systému řízení přístupu si tedy musíme položit otázku, jaké bezpečnosti potřebujeme dosáhnout. To by se mělo odvíjet od škody, kterou by potenciální útočník mohl napáchat v případě oklamání přístupového systému. Podle toho jsme pak schopni přibližně odhadnout, jaké zabezpečení by mělo být nasazeno.

Základní variantou autentizace je využívání haše pro přenos heslo. To při použití slabších hašovacích algoritmů (MD5, SHA-1) nezaručuje bezpečnost systému. Protože na tyto haše už byly publikovány útoky. V případě hašování algoritmy SHA-2 a SHA-3 by měl být obsah pro útočníka nedešifrovatelný. Je ale samozřejmostí, že vytvoření těchto hašů vyžaduje větší výpočetní výkon.

Varianta, která by měla poskytnout větší bezpečnost oproti hašování hesla, je šifrování celé komunikace s využitím symetrických kryptosystémů. Pokud obě komunikující strany dokážou udržet v tajnosti své šifrovací a dešifrovací klíče, neměl by mít útočník šanci komunikaci dešifrovat. Za předpokladu, že není využívaná šifra s nedostatečně dlouhým klíčem, jakou je například DES. Šifra DES již ale většinou byla nahrazena novými algoritmy, které takovými nedostatky netrpí.

V případě, kdy komunikující strany mají přístup k certifikační autoritě a disponují dvojicí asymetrických klíčů, může být použita autentizace s využitím certifikátů. Autentizaci s využitím certifikátů lze považovat za nejbezpečnější, není však v praxi příliš zastoupena. Což je dáno tím, že žadatelé (přístupové karty, uživatelské počítače) málokdy disponují certifikátem zaregistrovaným u certifikační autority. Disponuje-li certifikátem pouze jedna komunikující strana, lze využít kombinací autentizace. Kdy se například server prokazuje svým certifikátem a žadatel hašem svého hesla.

Trendem při řízení přístupu je oboustranná autentizace, což spolu s šifrováním komunikace a odesíláním pečetí zpráv minimalizuje možnosti útoků typu Man in the middle. Moderní přístupové systémy kromě autentizace a autorizace jsou schopny zaznamenávat počet přístupů a žadatelem využívané služby do databáze.

Za jeden z přínosů práce lze považovat studie zpráv, využívaných různými autentizačními protokoly, na základě kterých jsem popsal autentizační protokol schopný obsáhnout širokou škálu autentizačních metod. Protokol byl navrhován s ohledem jeho možné implementace do protokolu ACP, který by řešil problém interoperability protokolů řízení přístupu využívaných v současnosti.

Hlavním úkolem práce byl výukový materiál spustitelný ve webovém prohlížeči. Ten je přiložen na CD spolu s prací. Jeho cílem je ukázání a popsání konkrétních výhod a nedostatků jednotlivých protokolů přímo na příkladech. Díky využití interaktivních animací a testů se snaží čtenáři co nejvíce představit problematiku těchto protokolů

LITERATURA

[1] BURDA, K. Řízení přístupu v počítačových sítích [Řízení přístupu v počítačových sítích]. 2012, č. 2, s. 11 [cit. 0527]. Dostupné z: http://www.elektrorevue.cz/cz/clanky/komunikacni-technologie/40/rizeni-pristupu-v-pocitacovych-sitich/

[2] The EAP-PSK Protocol: A Pre-Shared Key Extensible Authentication Protocol (EAP) Method [The EAP-PSK Protocol]. 2007, č. 1 [cit. 0526]. Dostupné z: http://tools.ietf.org/html/rfc4764

[3] The EAP-TLS Authentication Protocol [The EAP-TLS Authentication Protocol]. 2008, č. 1 [cit. 0526]. Dostupné z: http://www.ietf.org/rfc/rfc5216.txt

[4] BURDA, K. a I. STRAŠIL. *Zabezpečovací systémy* [Zabezpečovací systémy].. Brno: Vysoké učení technické, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2012, 187 s. s.. ISBN 978-80-214-4441-6.

[5] media.blackhat. [Don't stand so close to me] In: *Don't stand so close to me: An analysis of the NFC attack surface* [online]. 2012 [cit. 1217. Dostupné z: https://media.blackhat.com/bh-us-12/Briefings/C_Miller/BH_US_12_Miller_NFC_attack_surface_Slides.pdf

[6] Access server FELD ČVUT. [Klonování RFID čipů na přístupových kartách] In: *Klonování RFID čipů na přístupových kartách* [online]. 2012 [cit. 1204. Dostupné z: http://access.feld.cvut.cz/view.php?cisloclanku=2012070003

[7] Challenge-Response Based RFID Authentication Protocol for Distributed Database Environment. In: *Lecture Notes in Computer Science* [Challenge-Response Based RFID Authentication Protocol for Distributed Database Environment]. Second International Conference. Boppard, Germany: Springer, 2005, s. 79-93 [cit. 0527].

[8] VOMÁČKA, J. *Řízení přístupu pomocí čipových karet* [Řízení přístupu pomocí čipových karet]. Brno: 2011. Diplomová práce. Masarykova univerzita.

[9] Information Security Group. [HB and Related Lightweight Authentication Protocols for Secure RFID Tag/Reader Authentication] In: *HB and Related Lightweight Authentication Protocols for Secure RFID Tag/Reader Authentication* [online]. 2006 [cit. 0527. Dostupné z: http://www.avoine.net/rfid/download/papers/Piramuthu-2006-collecter.pdf

[10] KALLA, T. *Bezpečnost mifare karet* [Bezpečnost mifare karet]. Brno: 2012 [cit. 1204]. Bakalářská práce. Masarykova univerzita. Dostupné z: http://is.muni.cz/th/324974/fi_b/Bakalarska_prace_final.pdf

[11] ZAORALOVÁ, L. *Systém elektronických plateb* [Systém elektronických plateb]. Brno: 2009 [cit. 1215]. Diplomová práce. Masarykova univerzita. Dostupné z: http://is.muni.cz/th/99184/fi_m/DiplomovaPrace.pdf

[12] KRHOVJÁK, V. LORENC a V. MATÝÁŠ. Autentizace a autorizace finančních transakcí [Autentizace a autorizace finančních transakcí]. 2007, **XVIII** (č. 1) [cit. 1216]. Dostupné z: http://www.ics.muni.cz/zpravodaj/articles/561.html

[13] BENÁK, K. *Použití adresářových služeb v informačních systémech* [Použití adresářových služeb v informačních systémech]. Praha: 2004. Diplomová práce. České vysoké učení technické v Praze [cit. 0527]. Dostupné z: http://ldap.benak.net/diplom.pdf

[14] Access Control Protocol (ACP). *Access Control Protocol (ACP): draft-kaaps-acp-01*. Brno: 2011 [cit. 2452014]. Dostupné z: http://tools.ietf.org/html/draft-kaaps-acp-01

[15] Mathematical routines for the NIST prime elliptic curves [Mathematical routines for the NIST prime elliptic curves]. 2010, č. 1, s. 44s [cit. 0524]. Dostupné z: http://www.nsa.gov/ia/_files/nist-routines.pdf

[16] SEC 2: Recommended Elliptic Curve Domain Parameters [SEC 2: Recommended Elliptic Curve Domain Parameters]. 2000, č. 1 [cit. 0524]. Dostupné z: http://www.secg.org/collateral/sec2_final.pdf

[17] PADMAVATHI, B. a R. KUMARI. A Survey on Performance Analysis of DES, AES and RSA Algorithm along with LSB Substitution … [A Survey on Performance Analysis of DES, AES and RSA Algorithm along with LSB Substitution …]. 2013, č. 1 [cit. 0526]. Dostupné z: http://ijsr.net/archive/v2i4/IJSRON120134.pdf

[18] National Security Agency. [The Case for Elliptic Curve Cryptography] In: *The Case for Elliptic Curve Cryptography* [online]. 2009 [cit. 0526. Dostupné z: http://www.nsa.gov/business/programs/elliptic_curve.shtml

[19] W3SCHOOLS. W3Schools. [Browser Statistics] In: *Browser Statistics* [online]. 2014 [cit. 0526. Dostupné z: http://www.w3schools.com/browsers/browsers_stats.asp

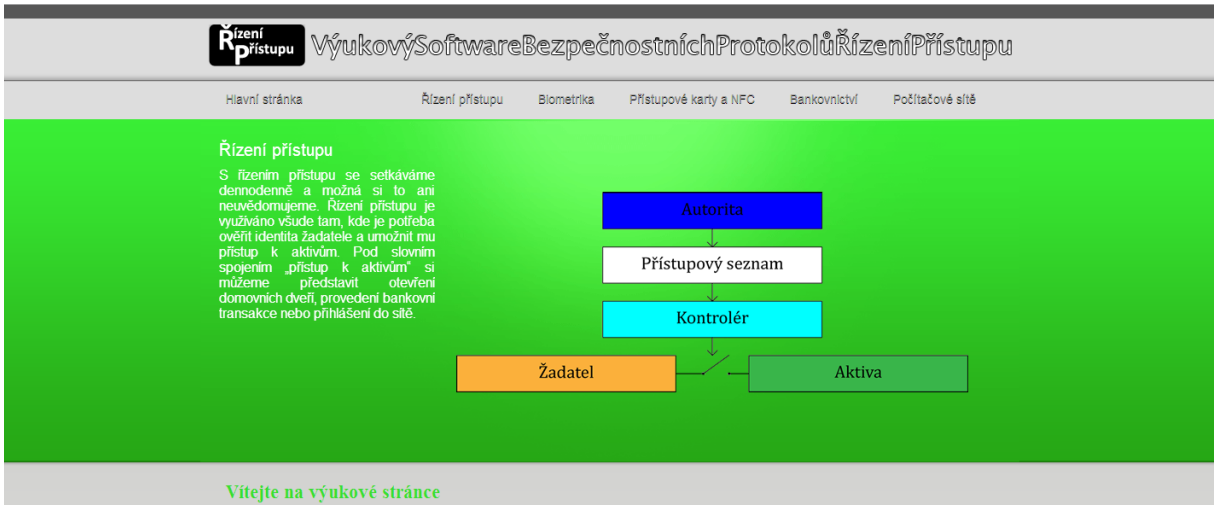
SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

3DES	Triple Data Encryption Standard
AA	Autentizační a autorizační
AAA	Autentizační, autorizační a účtovací
AAC	Application Authentication Cryptogram
AC	Application Cryptogram
ACP	Access Control Protocol
AES	Adavanced Encryption Standard
ANSI	American National Standards Institute
ARPANET	Advanced Research Projects Agency Network
ARQC	Kryptogram s požadavkem na autorizaco
ASCII	American Standard Code for Information Interchange
AVP	Attribute–value pair
CAP	Chip Authentication Program
CAVP	Container Attribute–value pair
CBC	Cipher-block chaining
CDA	Combined Dynamic Data Authentication
CFB	Cipher feedback
CSS	Cascading Style Sheets
CTR	Counter
DAP	Data Authentication Pattern
DDA	Dynamic Data Authentication
DES	Data Encryption Standard
DG	Data Group
DH	Diffie Hellman
DSS	Digital Signature Standard
EAL+	Evaluation Assurance Level
EAP	Extensible Authentication Protocol
EAPoL	EAP over LANs
EAPoW	EAP over Wireless
EC	Eliptic Curve
ECB	Electronic codebook
EEPROM	Electrically Erasable Programmable Read-Only Memory
EMV	Europay, MasterCard and Visa
FFIEC	Federal Financial Institutions Examination Council
GTC	Generic Token Card
HMAC	Keyed-hash Message Authentication Code
HSM	Hardware Security Module
HTML	HyperText Markup Language
HTTPS	Hypertext Transfer Protocol Secure
CHAP	Challenge Handshake Authentication Protocol
IDEA	International Data Encryption Algorithm
IEEE	Institute of Electrical and Electronics Engineers
IKEv2	Internet Key Exchange version 2
IPSEC	IP security
KDK	Key Derivation Key
LAVP	Long Attribute–value pair
LDAP	Lightweight Directory Access Protocol
LLC	Logical Link Control
LLCP	Logical link control protocol
MAC	Message Authentication Code
MB	Message Begin

MD5	Message-Digest algorithm
ME	Message End
NAS	Network Acces Server
NDEF	NFC Data Exchange Format
NFC	Near field communication
NFCIP	Near Field Communication -- Interface and Protocol
NIST	National Institute of Standards and Technology
NRZ	Non Return to Zero
NSA	National Security Agency
OTP	One Time Password
PAP	Password authentication protocol
PCBC	Propagating cipher-block chaining
PEAP	Protected EAP
PIN	personal identification number
PPP	Point to Point Protocol
PSK	Pre Shared Key
RADIUS	Remote Authentication Dial In User Service
RC4	ARCFOUR
RF	Radio Frequency
RFC	Request For Comments
RFID	Radio Frequency Identification
RKKW	Rhee, Kwak, Kim and Won protocol
RSA	Rivest, Shamir, Adleman protokol
SAD	Static Authentication Data
SAM	Secure Access Module
SAVP	Short Attribute–value pair
SCTP	Stream Control Transmission Protocol
SE	Secure Element
SECG	Standards for Efficient Cryptography Group
SHA-1	Secure Hash Algorithm 1
SHA-2	Secure Hash Algorithm 2
SHA-3	Secure Hash Algorithm 3
SR	Short Record
SSL	Secure Socket Layer
TACACS	Terminal Access Controller Access-Control System
TC	Transaction Certificate
TCP	Transmission Control Protocol
TEK	Transient EAP Key
TGS	Ticket-Granting Server
TK	Transient Key
TLS	Transport Layer Security
TNF	Type Name Format
UDP	User Datagram Protocol
UID	Unique Identification
USB	Universal Seriál Bus
XML	Extensible Markup Language

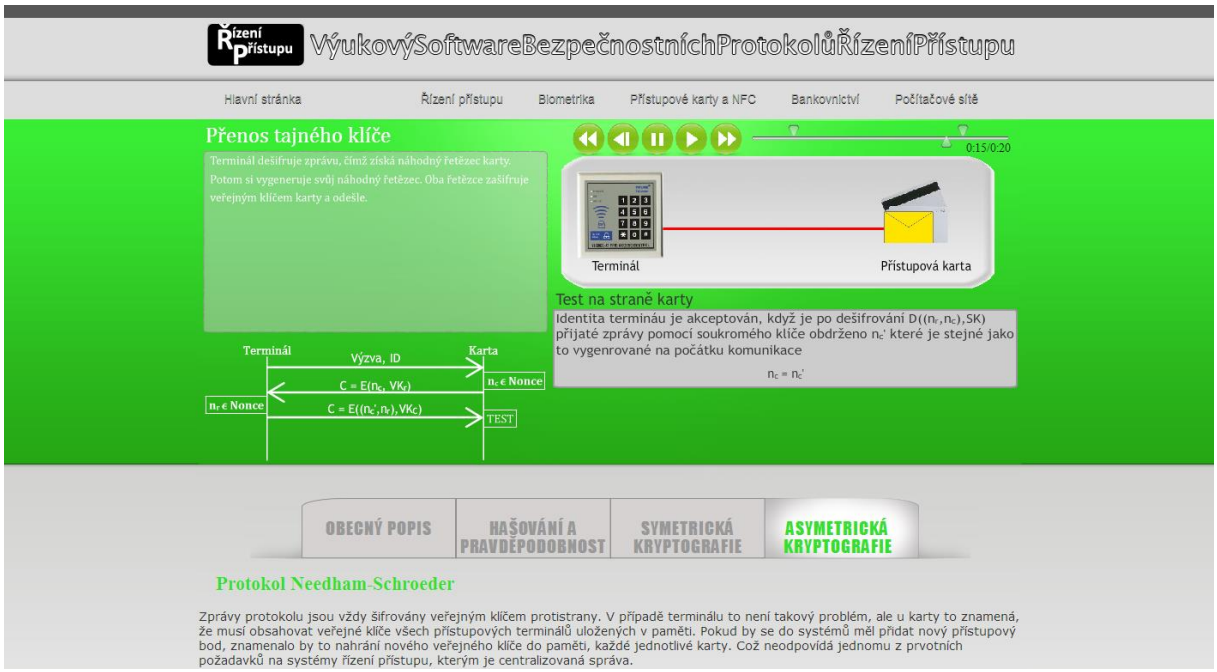
SEZNAM PŘÍLOH

A. SNÍMKY VÝUKOVÉHO MATERIÁLU

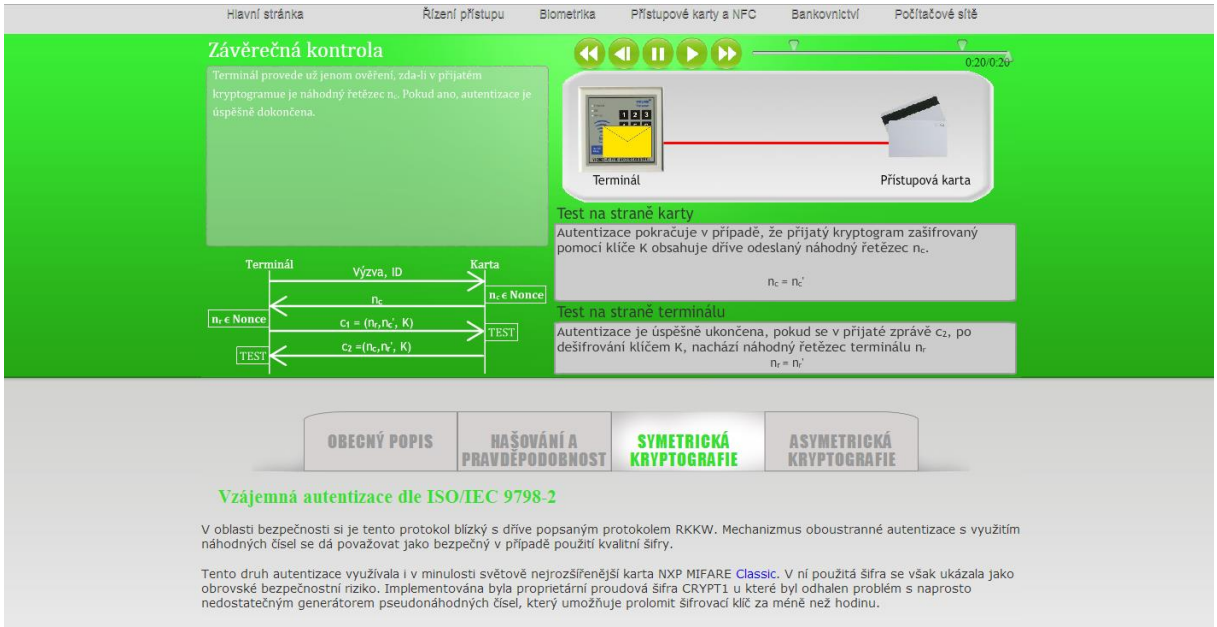


Obr. A.1 Titulní strana výukového materiálu

Obr. A.2 Test vědomostí za kapitolou řízení přístupu



Obr. A.3 Stránka popisující protokol Needham-Schroeder



Obr. A.4 Stránka pojednávající o oboustranné autentizaci dle ISO/IEC 9798-2

B. Obsah přiloženého CD

1. /bp_Kopulety_146866.pdf – text bakalářské práce ve formátu .pdf
2. /Vyukovy_material/ - složka obsahující zdrojové kódy, animace ve formátu .swf a obrázky nutné pro běh výukového materiálu.
3. /Flash/ - složka obsahující veškeré nezkompilované animace ve formátu .fla