UNIVERZITA PARDUBICE
Fakulta elektrotechniky a informatiky

Návrh a konfigurace vzdáleného přístupu na koncové prvky

Jáchym Krasek

Diplomová práce
2013

Univerzita Pardubice
Fakulta elektrotechniky a informatiky
Akademický rok: **2012/2013**

ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení:	**Bc. Jáchym Krasek**
Osobní číslo:	**I11391**
Studijní program:	**N2646 Informační technologie**
Studijní obor:	**Informační technologie**
Název tématu:	**Návrh a konfigurace vzdáleného přístupu na koncové prvky**
Zadávající katedra:	**Katedra softwarových technologií**

Zásady pro vypracování:

Cílem práce je navrhnout a realizovat zabezpečený vzdálený přístup z veřejné sítě internet do lokální zabezpečené LAN. Autor představí problematku zabezpečení vzdálených přístupů s důrazem na principy protokolu IPsec a technologií VPN. Dále bude podrobně představena problematika hardwarových a softwarových certifikátů a jejich ověřování. V implementační části autor navrhne a v laboratorních prostředí realizuje a podrobně zdokumentuje metody vzdáleného přístupu ze simulované veřejné sítě internet do zabezpečené sítě LAN umístěné za NAT s dvoufaktorovým ověřením uživatele a jeho autentizací.

Rozsah grafických prací:

Rozsah pracovní zprávy:

Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam odborné literatury:

[1] Luhový Karel, seriál o VPN, 2003, soubor internetových článků dostupných (březen 2007) na http://www.svetsiti.cz/view.asp rubrika=Tutorialy&temaID=219&clanekID=220
[2] Virtual Private Network Consorcium, http://www.vpnc.org
[3] Scott C., Wolfe P., Erwin M: Virtual Private Networks, druhé vydání, O'Reilly, 1999, ISBN 1-56592-529-7
[4] International Organization for Standardization, http://www.iso.org
[5] Thomas M. T.: Zabezpečení počítačových sítí bez předchozích znalostí, CP Books, 2005, ISBN 80-251-0417-6
[6] Peterka Jiří, Principy počítačových sítí, dostupné (duben 2007) na http://www.spsest.sk/spse/jpeterka_siet/f_pri.htm
[7] Schneier B. a Mudge, Cryptanalysis of Microsoft's Point-to-Point Tunneling Protocol (PPTP), 1998, dostupné (duben 2004) na http://www.schneier.com/paper-pptp.html
[8] Schneier B. a Mudge, Cryptanalysis of Microsoft's PPTP Authentication Extensions (MS-CHAPv2), 1999, dostupné (duben 2007) na http://www.schneier.com/paper-pptpv2.html
[9] Daler Ivan, IPSec Howto, 2006, dostupné (duben 2007) na http://www.ipsec-howto.org/ipsec-howto_cz.html
[10] The Internet Engineering Task Force, http://www.ietf.org
[11] Hladík Radek, OpenVPN jednoduše, 2004, dostupné (duben 2007) na http://www.root.cz/clanky/openvpn-vpn-jednoduse/

Vedoucí diplomové práce: **Ing. Soňa Neradová**
Katedra softwarových technologií

Datum zadání diplomové práce: **31. října 2012**
Termín odevzdání diplomové práce: **17. května 2013**

prof. Ing. Simeon Karamazov, Dr.
děkan

L.S.

prof. Ing. Antonín Kavička, Ph.D.
vedoucí katedry

Prohlášení autora

Prohlašuji, že jsem tuto práci vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Souhlasím s prezenčním zpřístupněním své práce v Univerzitní knihovně.

V Pardubicích dne 10. 05. 2013 Jáchym Krasek

Poděkování

Na tomto místě bych rád poděkoval paní Ing. Soně Neradové za vedení diplomové práce a také za její rady, inspiraci a diskuze nejen při vypracování této diplomové práce. Rád bych také poděkoval Univerzitě Pardubice za možnost zpracování této práce a své rodině, za možnost studovat.

Anotace

Tato diplomová práce se zabývá virtuálními privátními sítěmi se zaměřením na IPsec. Práce shrnuje teoretické poznatky o IPsec frameworku a všech jeho součástech a zahrnuje také návrh možné konfigurace pro vzdálený přístup na koncové prvky s využitím zařízení Cisco

Klíčová slova

VPN, IPsec, Authentication Header, Encapsulating Security Payload, bezpečnost, vzdálený přístup, konfigurace, Cisco

Title

Design and Configuration Remote Access for End Devices.

Annotation

This thesis deal with virtual private networks and it’s focused on IPsec. The first part of the thesis is focused on theoretical description of private networks and all parts of IPsec framework. The second part addresses practical configuration of remote access on Cisco devices.

Keywords

VPN, IPsec, Authentication Header, Encapsulating Security Payload, security, remote access, configuration

Obsah

Seznam zkratek

VPN	Virtual Private Network
IANA	Internet Assigned Numbers Authority
PSN	Packet Switched Network
EDI	Electronic Data Interchange
ISP	Internet Service Provider
ATM	Asynchronous Transfer Mode
MPLS	Multiprotocol Label Switching
PDU	Protocol Data Unit
L2TP	Layer 2 Tunneling Protocol
SSL	Secure Sockets Layer
QoS	Quality of Service
BGP	Border Gateway Protocol
AH	Authentication Header
ESP	Encapsulating Security Payload
POP3	Post Office Protocol
SMTP	Simple Mail Transfer Protocol
IMAP	Internet Message Access Protocol
SSH	Secure Shell
SA	Security Association
RFC	Request for Comments
IKE	Internet Key Exchange
DSCP	Differentiated Services Codepoint
MAC	Message Authentication Code
HMAC	Hash Message Authentication Code
DES	Data Encryption Standard
3DES	Triple Data Encryption Standard
AES	Advanced Encryption Standard
SEAL	Software-Optimized Encryption Algorithm
NIST	National Institute of Standards and Technology
EFF	Electronic Frontier Foundation
CTR	Counter
CBC	Cipher-block Chaining
MD5	Message-Digest Algorithm
SHA	Secure Hash Algorithm
AAA	Autentizace, Autorizace a Accounting (účtování)
EAP	Extensible Authentication Protocol
CA	Certifikační autorita
CRL	Certificate Revocation List
DH	Diffie–Hellman

Seznam obrázků

Seznam tabulek

Úvod

Vzdálený přístup na koncové prvky je poměrně rozsáhlou problematikou, která se řeší již mnoho let. Dříve bylo nutné pro bezpečný vzdálený přístup, například při spojení dvou firemních poboček, využívat samostatnou pronajatou linku.

Takové řešení bylo nejen velmi nákladné, ale také neefektivní a přenos dat byl velmi pomalý. Jen minimum firem si navíc mohlo dovolit pronajmout samostatnou vysokokapacitní linku pro své potřeby. Při potřebě vzdáleně propojit více poboček, dosahovaly náklady na pronájem linek velmi vysokých částek. Vzdálený přístup zaměstnanců do firemní sítě byl dříve v podstatě nemyslitelný.

Změnu v této oblasti přinesl až rozvoj širokopásmového Internetu a virtuálních privátních sítí (VPN). Dostatečně rychlé připojení je dnes dostupné velmi snadno a virtuální privátní sítě zajišťují dostatečně bezpečné spojení.

Tato diplomová práce se v první kapitole zabývá obecnou problematikou virtuálních privátních sítí, bezpečností a možností pro nasazení VPN. Druhá kapitola je věnována technologii IPsec, což je jedna z možností, jak lze virtuální privátní sítě vytvářet s využitím veřejné infrastruktury.

V dalších pěti kapitolách jsou podrobně rozebrány jednotlivé součásti IPsec frameworku. Osmá kapitola se věnuje navazování IPsec tunelu a protokolu IKE. Poslední kapitola je věnována praktické konfiguraci IPsecu v síťových laboratořích Univerzity Pardubice s využitím zařízení od společnosti Cisco.

Práce shrnuje všechny potřebné teoretické i praktické znalosti, které musí administrátor mít, pokud se rozhodne implementovat vzdálený přístup pomoci frameworku IPsec.

1 Úvod do virtuálních privátních sítí

Virtuální privátní síť (VPN) je dle definice VPN konsorcia privátní datová síť, která využívá veřejné telekomunikační infrastruktury. Pomocí speciálních tunelovacích a bezpečnostních protokolů zajišťuje soukromí a bezpečnost přenášených dat.

Hlavním účelem VPN je poskytnout společnostem či jednotlivcům stejné možnosti, jaké nabízí pronajaté nebo vlastní linky, ale za nižší cenu a s použitím veřejné infrastruktury. Společnosti dnes využívají VPN jak pro vybudování extranetu a rozsáhlých intranetových infrastruktur, tak pro vzdálený přístup.

Dříve, než se Internet stal takřka univerzálním médiem, se skládaly virtuální privátní sítě z jednoho či více okruhů pronajatých od poskytovatele. Každý pronajatý okruh fungoval jako samostatný kabel kontrolovaný zákazníkem. Se správou připojení mu přitom obvykle pomáhal poskytovatel. Základní myšlenkou však bylo to, že zákazník může využívat tuto linku stejně jako fyzickou infrastrukturu ve své lokální síti.

Soukromí poskytované tímto typem VPN bylo založeno pouze na tom, že poskytovatel ujistil zákazníka, že nikdo jiný nebude využívat stejný okruh. To dovolovalo zákazníkovi vyžívat na okruhu vlastní IP adresaci a vlastní bezpečnostní politiky.

Pronajatý okruh využíval jeden či více přepínačů u poskytovatele a každý z přepínačů mohl být kompromitován někým, kdo chtěl získat přístup k provozu na privátním okruhu. Zákazník musel poskytovateli důvěřovat, že zajistí integritu okruhu a vyhne se odposlouchávání dat. Tento typ VPN sítě je tedy označován jako **důvěrná (trusted) VPN.**

V době kdy začal být Internet stále populárnějším komunikačním médiem i pro firmy, začala být bezpečnost naléhavým problémem jak pro zákazníky, tak pro poskytovatele. Protože důvěrná (trusted) VPN nezajišťuje žádnou reálnou bezpečnost, začali výrobci síťových zařízení pracovat na protokolech, které zajistí šifrování dat v počáteční stanici, jejich přenos po Internetu a dešifrování v koncové stanici.

Přenášená data se chovají jako v tunelu mezi dvěma sítěmi. Přestože útočník může data vidět, nedokáže je přečíst a ani modifikovat, aniž by si toho příjímací stanice všimla a neodmítla modifikovaná data. Sítě využívající šifrování se nazývají **bezpečné (secure) VPN.**

Bezpečná VPN síť také může fungovat jako součást důvěrné VPN sítě a dohromady tak tvoří třetí typ – **hybridní VPN**. Zabezpečené části hybridní VPN mohou být řízeny zákazníkem (například využitím VPN zařízení na obou stranách připojení) či poskytovatelem služby, který nabízí důvěrnou část hybridní VPN.

1.1 Důvěrná VPN

Firmy využívají důvěrnou VPN, pokud potřebují vědět, že jejich data se pohybují přes skupinu cest se specifikovanými vlastnostmi a jsou kontrolovány jedním ISP či konfederací ISP.

Zákazníkům to umožňuje v síti využít vlastní IP adresaci a případně řídit i samotné směrování. Zákazník musí plně důvěřovat poskytovateli, že dodrží stanovené cesty v sítí a zajistí ochranu před nedůvěryhodnými osobami, které nemohou do sítě nijak zasahovat.

Zákazník nezná a ani si nemůže ověřit cesty, kterými důvěrná VPN prochází a vše je tak plně v rukou poskytovatele.

1.1.1 Požadavky na důvěrnou VPN

VPN konsorcium definuje pro každý typ VPN několik požadavků. Pro důvěrnou VPN jsou požadavky stanoveny takto:

- pouze důvěryhodný poskytovatel VPN připojení může vytvářet a modifikovat cesty ve VPN síti, nikdo jiný,
- pouze důvěryhodný poskytovatel VPN připojení může měnit, injektovat a mazat data a řídit datový tok ve VPN síti,
- směrování a IP adresace použitá v důvěrné síti musí být stanovena předtím, než je vytvořena samotná VPN síť.

1.1.2 Technologie pro vybudování důvěrné VPN

Moderní poskytovatelé připojení nabízí různé typy důvěrných VPN. Lze je rozdělit na VPN pracující na druhé síťové vrstvě a třetí síťové vrstvě.

Technologie pracující na druhé síťové vrstvě jsou:

- ATM okruhy,
- Frame Relay okruhy,
- Přenos rámců druhé síťové vrstvy pomocí MPLS popsaný v RFC 4761.

Na třetí síťové vrstvě pracuje technologie:

- MPLS s omezenou distribucí směrovacích informací skrze BGP, jak jej popisuje RFC 4364.

1.2 Bezpečná VPN

Hlavním důvodem, proč společnosti vyžadují důvěrnou VPN, je přenos citlivých informací po Internetu bezpečnou cestou. Všechna data procházející bezpečnou VPN jsou šifrována a i když jsou zachycena, nemůže je útočník přečíst. Bezpečné VPN také dávají uživatelům jistotu, že data nebyla útočníkem modifikována.

1.2.1 Požadavky na bezpečnou VPN

VPN konsorcium stanovuje pro bezpečnou VPN celkem tři požadavky:

- všechna data procházející bezpečnou VPN musí být šifrována a autentizována (musí být známa identita subjektu),
- bezpečnostní nastavení musí souhlasit na všech stranách VPN. Bezpečná VPN nabízí jeden čí více tunelů a každý tunel má dva koncové body. Administrátoři těchto koncových bodů se musí shodnout na bezpečnostních vlastnostech tunelu,

- nikdo mimo VPN síť nemůže měnit bezpečnostní nastavení. Pro útočníka musí být nemožné změnit bezpečnostní nastavení, například nastavit slabší šifrování či ovlivnit volbu šifrovacího klíče.

1.2.2 Technologie pro vybudování bezpečné VPN

Za bezpečnou VPN lze podle VPN konsorcia považovat virtuální privátní síť vyžívající níže uvedené technologie:

- IPsec,
- IPsec uvnitř L2TP (popisuje jej RFC 3193),
- SSL 3.0 nebo TLS s šifrováním.

1.3 Hybridní VPN

Bezpečná a důvěrná VPN mají mnoho odlišných vlastností. Bezpečná VPN zajišťuje bezpečnost dat, ale negarantuje průchozí cesty. Na druhou stranu důvěrná VPN zajišťuje průchod dat přes stanovené cesty spolu s QoS (Quality of Service), ale neumí zajistit ochranu před snoopingem či záměnou dat.

A právě tyto slabiny odstraňuje hybridní VPN. Typickou situací pro použití hybridní VPN je firma, která již vlastní důvěrnou VPN a některé části sítě vyžadují zabezpečení. Firmám nic nebrání ve vybudování bezpečné VPN nad vlastní důvěrnou VPN a někteří výrobci zařízení nabízí systémy, které explicitně podporují vytváření hybridních VPN sítí.

1.3.1 Požadavky na hybridní VPN

U hybridní VPN musí být především jasné hranice bezpečné VPN pracující uvnitř důvěrné VPN. Hybridní VPN je totiž bezpečná VPN vybudovaná v rámci důvěrné VPN sítě.

Příkladem může být, když jedno oddělení firmy provozuje vlastní bezpečnou VPN skrze firemní důvěrnou VPN síť. Pro každý pár adres v hybridní VPN musí být administrátor schopen jednoznačně říci, zda je či není provoz mezi těmito dvěma uzly součástí bezpečné VPN.

1.3.2 Technologie pro vybudování hybridní VPN

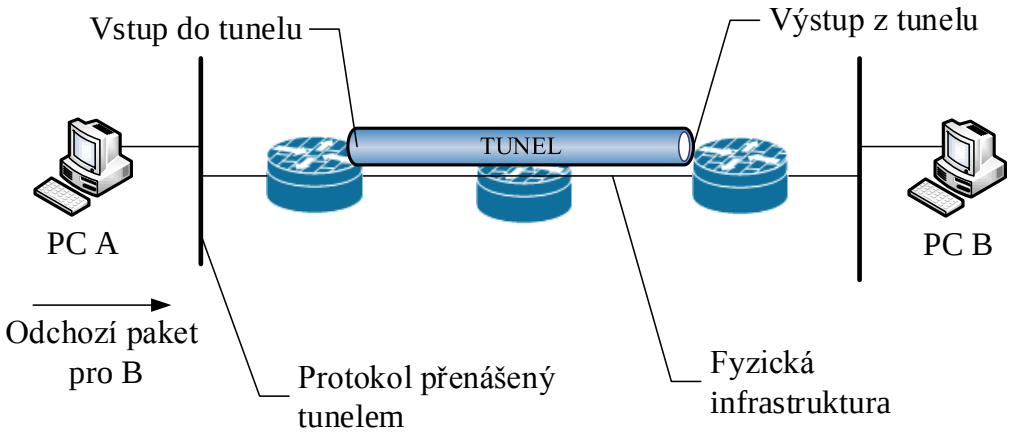
Hybridní VPN v sobě kombinuje vlastnosti důvěrné a bezpečné VPN, stejně to platí u použitých technologií. Podle VPNC je hybridní VPN každá podporovaná bezpečná VPN technologie pracující na libovolné podporované důvěrné VPN technologii.

1.4 Tunelování

Virtuální privátní sítě jsou většinou založeny na vytváření tunelů na veřejné infrastruktuře. Princip síťového tunelování spočívá v tom, že PDU (Protocol Data Unit - datové prvky protokolu) jednoho protokolu jsou zapouzdřena do PDU stejného či jiného protokolu a poté jsou přenášena danou sítí.

Přidáním doplňujícího záhlaví se vlastně schová původní PDU, která se tak stává pro přenosovou síť nečitelnou. Díky tomu lze například přenášet data jednoho protokolu přes

nekompatibilní síť (například při tunelování IPv6 v IPv4), obcházet administrativní omezení sítě (zakázané protokoly) a také poskytnout zabezpečenou komunikaci přes nezabezpečenou síť.



Obrázek 1- Princip tunelování

1.4.1 Tunelování podle síťových vrstev

Nejčastěji je tunel tvořen na druhé a třetí síťové vrstvě. Podle Pužmanové[1] lze dělit tunelování takto:

- na třetí (síťové) vrstvě ISO/OSI modelu,
- na druhé (spojové) vrstvě ISO/OSI modelu.

Na třetí síťové vrstvě probíhá tunelování formou zapouzdřování IP datagramů do jiných datagramů (například IP v IP, IPv6 v IPv4) a jako bezpečnostní mechanismus zde nejčastěji figuruje IPsec.

Na druhé síťové vrstvě probíhá tunelování pomocí tunelování rámců, přičemž existují dva typy:

- dobrovolné (voluntary),
- povinné (compulsory).

V případě dobrovolného tunelování leží správa VPN tunelu na klientovi, který sám a tedy dobrovolně naváže spojení s VPN serverem přes veřejnou infrastrukturu. V případě povinného (compulsory) tunelování iniciuje VPN spojení přístupový server (NAS, network access server) bez přičinění VPN klienta. O VPN spojení se tedy nestará samotný klient, ale jeho správa leží na bedrech poskytovatele.

[1] PUŽMANOVÁ, Rita. Moderní komunikační sítě od A do Z. 2. akt. vydání. Brno: ComputerPress, 2006, s. 284.

1.4.2 Protokoly pro tunelování

Protokolů pro tunelování existuje celá řada, pracují na různých síťových vrstvách a poskytují rozlišné služby. V této části jsou zmíněny pouze protokoly nejčastěji používané pro VPN tunelování.

Jako tunelovací protokol lze použít:

- Generic Routing Encapsulation (GRE),
- Layer Two Tunneling Protocol (L2TP),
- IPsec.

V podstatě lze říci, že tunelovat lze vždy a všechno, neboť pojem tunelování není přesně definován. Příkladem takového přístupu je například HTTP tunelování. To funguje na velmi jednoduchém principu. Klient naváže se serverem HTTP tunel a všechna data všech protokolů posílá přes tento tunel. Server tyto data vyjme z tunelu a odešle je dále. Z pohledu firewallu to tedy vypadá, že klient komunikuje pouze HTTP protokolem a žádným jiným.

Protože klient ale balí všechny ostatní protokoly do HTTP, komunikuje i ostatními protokoly. Tímto způsobem se dá například obejít zabezpečení na firewallu, které různé protokoly zakazuje.

1.5 Rozdělení VPN podle technologií

Technologií pro virtuální privátní sítě existuje velmi mnoho, nicméně pouze některé jsou podporovány VPN konsorciem.

Konkrétně jde o tyto technologie:

- ATM a Frame Relay okruhy,
- MPLS na druhé síťové vrstvě,
- MPLS na třetí síťové vrstvě (MPLS/ BGP VPN),
- IPsec,
- L2TP/IPsec,
- SSL 3.0 či TLS s šifrováním.

Dále lze VPN vytvořit například pomocí protokolů a technologií:

- SSTP (Secure Socket Tunneling Protocol),
- DTLS (Datagram Transport Layer Security),
- MPPE (Microsoft Point-to-Point Encryption),
- GRE (Generic Routing Encapsulation),
- LANE (LAN Emulace),
- MPOA (Multiprotocol Encapsulation over ATM).

Tato práce se zaměřuje především na IPsec a ostatní technologie podporované VPNC jsou zmíněny pouze okrajově.

1.5.1 ATM a Frame Relay okruhy

Jak již bylo zmíněno v úvodní části, privátní sítě se dají vytvářet pomocí ATM či Frame Relay okruhů, nicméně nejde o VPN ve smyslu provozu na veřejné infrastruktuře, ale především o vytvářeních vlastních privátních propojení mezi pobočkami.

1.5.2 VPN pomocí MPLS

MPLS (Multiprotocol Label Switching) je mechanismus pro přepínání paketů pomocí speciálních značek (labelů). Díky MPLS je L3 hlavička kontrolována pouze jednou (při vstupu do MPLS domény) a poté již probíhá směrování na základě značek.

Směrovače LSR (Label Switching Router) tedy nemusí zkoumat celé své směrovací tabulky a místo toho předávají datagramy na základě svých individuálních tabulek značek.

Protože značky se v případě MPLS umisťují mezi hlavičku druhé a třetí vrstvy, je možné, je využit jako mechanismus pro tunelování v případě MPLS VPN. MPLS VPN se používá pro site-to-site VPN, pro vzdálený přístup se nehodí. Podle Pužmanové[2] je MPLS VPN zajímavá především výkonností přenosu (rychlé přepínání na základě značek) a možností regulovat provoz (traffic engineering). Díky QoS dokáže zajistit také požadovanou kvalitu služeb.

MPLS VPN zajišťuje poskytovatel služeb (v České republice například České Radiokomunikace) a to buď na druhé síťové vrstvě (L2VPN), nebo na třetí síťové vrstvě (L3VPN).

L2VPN jsou založeny na konceptu tzv. pseudowire (mechanismus emulace přenosových služeb přes PSN) a **L3VPN** jsou založeny na tunelech se zapouzdřením GRE nebo na MPLS/BGP (RFC 2547). MPLS/BGP využívá jak přepínání značek, tak směrovací protokol BGP.

1.5.3 IPsec a L2TP/IPsec

IPsec využívá tunelování na třetí síťové vrstvě a je podrobně rozebrán v kapitole 2. IPsec je možné používat buď samostatně, nebo v kombinaci s protokolem L2TP, čímž vzniká L2TP tunel chráněný prostřednictvím IPsecu.

Tunelovací protokol L2TP slouží pro přenášení PPP rámců skrze síť s přepínáním paketů (případně jiných sítí). Neposkytuje ale dostatečnou ochranu důvěrných dat a proto je potřeba bezpečnost řešit pomocí protokolů vyšších vrstev.

L2TP/IPsec VPN

Spolupráce IPsecu a L2TP protokolu v IP sítích je podrobně popsána v RFC 3193, spolupráce v dalších ne-IP sítích je popsána v dalších RFC dokumentech. V případě L2TP/IPsec jsou L2TP pakety zabaleny do IPsecu.

[2] PUŽMANOVÁ, Rita. Moderní komunikační sítě od A do Z. 2. akt. vydání. Brno: ComputerPress, 2006, s. 295.

Pro zabezpečený přenos musí L2TP poskytovat autentizaci, integritu a ochranu proti replay útokům pro kontrolní pakety a integritu a ochranu proti replay útokům pro datové pakety. Může také ověřovat důvěrnost paketů. Posledním, co L2TP musí, je poskytnutí škálovatelného managementu klíčů.

Aby mohl protokol splnit tyto požadavky, musí implementovat IPsec spolu s ESP protokolem, který chrání jak kontrolní, tak datové pakety. Transportní mód musí být podporován a je nejčastěji používaný, tunelovací mód se vyskytuje pouze u některých implementací.

Navazování tunelu v případě L2TP/IPsec lze shrnout do 4 kroků:

1. Vytvoření bezpečnostních asociací pro IPsec pomocí IKE protokolu,
2. navázání IPsec spojení (obvykle ESP v transportním módu),
3. vytvoření L2TP tunelu mezi uzly.

Provoz L2TP je poté chráněn pomocí IPsecu.

1.5.4 VPN pomocí SSL a TLS

SSL (Secure Sockets Layer) a TLS (Transport Layer Security) jsou kryptografické protokoly, které pracují mezi aplikační a transportní vrstvou a poskytují zabezpečení a autentizaci jednotlivých uzlů. TLS vychází z SSL, protokoly však nejsou kompatibilní. V následující části je pojmem SSL VPN myšlena také TLS VPN.

SSL a TLS zajišťují autentizaci zpráv, jejich důvěrnost a integritu. Existují tři verze SSL protokolu a dvě TLS protokolu:

- SSL 1.0 (již se nepoužívá),
- SSL 2.0 (není doporučováno),
- SSL 3.0,
- TLS 1.1 (SSL 3.1),
- TLS 1.2.

Z tohoto seznamu také jasně plyne vazba mezi SSL a TLS, které se vyvinulo z třetí verze SSL protokolu a někdy je také označováno jako SSL 3.1.

Existují tři typy SSL VPN:

- Web SSL (bez použití klienta),
- tenký klient,
- tunelovací mód s plnohodnotným klientem.

Webová SSL VPN, nebo také clientless (bez použití klienta) VPN, je založena na poskytování důvěrného obsahu pomocí webového prohlížeče. Moderní webový prohlížeč naváže zabezpečené spojení se serverem a autorizuje jej pomocí certifikátu, takže klient ví, že přistupuje ke správnému serveru. Server zpravidla identitu klienta nekontroluje a spoléhá na poskytnuté přihlašovací údaje.

Pomocí webové SSL VPN je možné zabezpečeně přistupovat k privátním zdrojům odkudkoli, kde je k dispozici webový prohlížeč. Privátními zdroji mohou být webové stránky intranetu, zjednodušený přístup k souborům či databázím a mnohé další.

SSL VPN **ve spojení s tenkým klientem** je podobná webovému typu VPN. Webový portál však funguje jen jako brána k dalším službám a je postaven například na Java appletu či Active X prvku, který se stará o přesměrování portů a další funkce. Proto se někdy tomuto typu přezdívá také SSL Portal VPN.

Díky přesměrování portů je možné využívat aplikace jako je mailový klient (POP3, SMTP, IMAP), telnet či SSH v nativních aplikacích a ne jen přes webový prohlížeč.

V tunelovém módu vyžaduje SSL VPN plnohodnotného klienta, který je potřeba nainstalovat na klientskou stanici. Klient je často dodáván již s konfigurací, například prostřednictvím webového portálu. V tunelovém módu je možné přistupovat vzdáleně a bezpečně v podstatě k libovolné aplikaci v privátní síti.

SSL VPN v tunelovém módu využívá ke vzdálenému přístupu studentů a zaměstnanců Univerzita Pardubice. Uživatelé si musí nainstalovat klientský software AnyConnect od společnosti Cisco.

1.6 Topologie VPN sítí

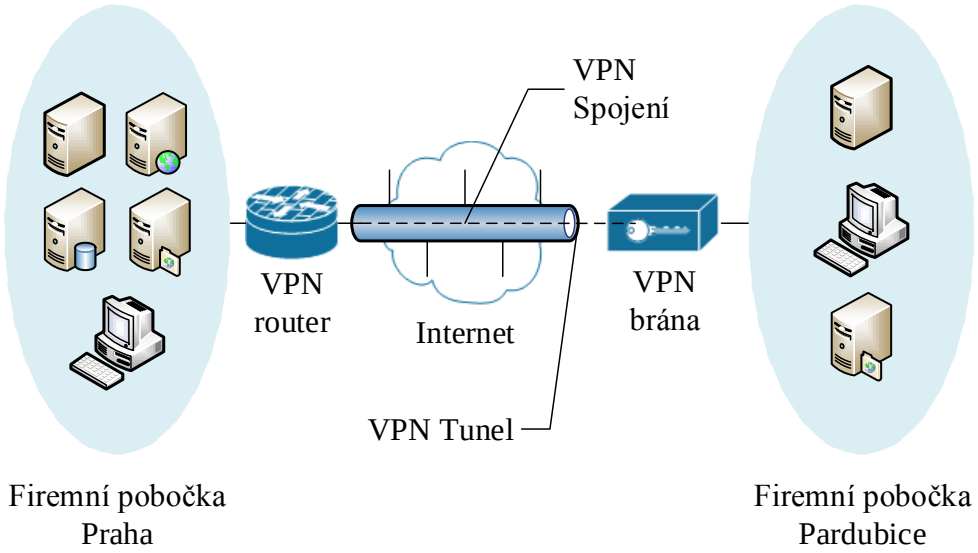
Existují dvě základní topologie VPN sítí – site-to-site a remote access VPN. Liší se zejména v tom, k jakému účelu se používají.

1.6.1 Site-to-site topologie

Site-to-site VPN je v podstatě rozšířením WAN sítě a zpravidla spojuje celé sítě. Příkladem může být spojení pobočkové firemní sítě s hlavní firemní sítí. Dříve se ke stejnému účelu využívaly pronajaté okruhy či spojení pomocí Frame Relay.

Koncová zařízení na dané síti komunikují v případě site-to-site VPN zcela běžně a fungování VPN je pro ně transparentní. O zajištění VPN tunelu, enkapsulaci a šifrování odchozích dat se stará VPN brána. Stejně je tomu u příjemce dat, kde se o dekapsulaci a dešifrování obsahu stará také VPN brána.

VPN bránu může představovat například směrovač, firewall či VPN Concentrator.



Obrázek 2 - Site-to-site VPN spojení

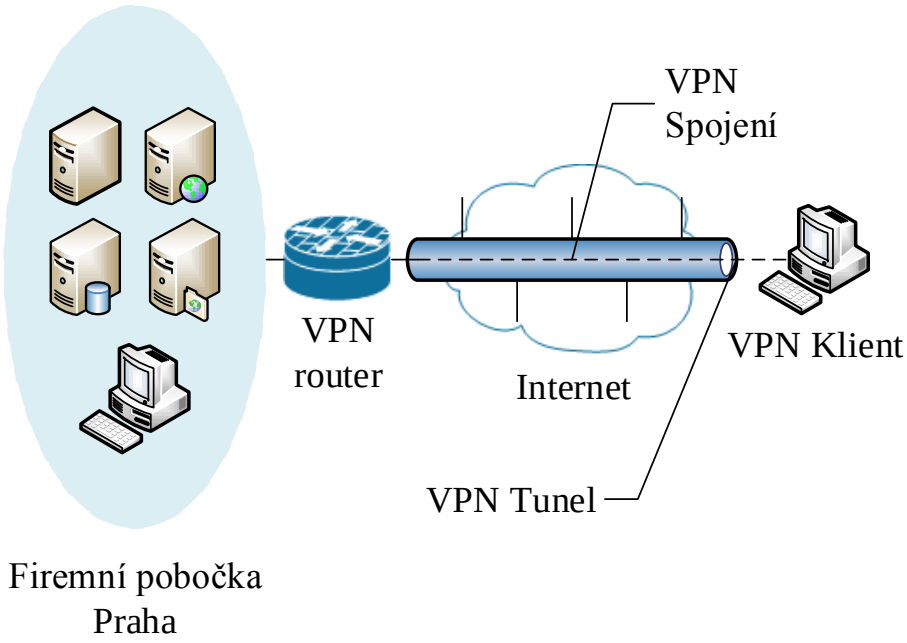
1.6.2 Remote access topologie

Remote access VPN neboli VPN pro vzdálený přístup je evolucí sítí s přepojovanými okruhy jako jsou POTS (Plain Old Telephone Service – starší telefonní sítě) či ISDN.

VPN pro vzdálený přístup je postavena na architektuře klient/server. Vzdálený host v takovém případě vyžaduje zabezpečený přístup do sítě pomocí VPN serveru, který se nachází na okraji sítě.

Host připojující se do vzdálené sítě musí využívat klientský software, který se stará o enkapsulaci a šifrování dat před odesláním do Internetu a také od dekapsulaci a dešifrování příchozích dat.

Typickým případem užití je mobilní pracovník připojující se zabezpečeným způsobem do firemní sítě.



Obrázek 3 - Remote access VPN

1.7 Přínosy VPN

K hlavním výhodám VPN patří především dramatické snížení nákladů, například při spojování jednotlivých firemních poboček. Díky VPN totiž není potřeba budovat oddělené privátní sítě, ale je možné využít veřejnou infrastrukturu.

VPN také rozšiřuje geografickou konektivitu, přičemž nezáleží na tom, kde se stanice nachází. VPN je z tohoto hlediska velmi flexibilní. Výhodou je také možnost využití stávajících technologií při expanzi nebo nasazování nových technologií. Například umožnění vzdáleného přístupu k dalším službám vnitřní sítě zpravidla neznamená výměnu infrastruktury, ale pouze změnu konfigurace.

Jednou z hlavních výhod je i zabezpečení dat, nicméně použití VPN ještě neznamená, že komunikace je bezpečná. Záleží totiž na implementaci a zvolené VPN technologii.

1.8 Rizika VPN

Přínosy VPN technologií nad negativy a riziky sice převládají, ale i tak je potřeba věnovat dostatečnou pozornost rizikům při použití VPN. Bezpečnost VPN sítí lze rozdělit do dvou kategorií – vnější a vnitřní.

Pokud je použito silné šifrování, jsou VPN chráněny proti vnějším hrozbám dobře a s proudem šifrovaných dat přenášeným po veřejné infrastruktuře toho útočník moc nezmůže. Větší bezpečnostní úskalí má vnitřní bezpečnost, zejména pokud se dokáže útočník dostat na některý z uzlů připojených ve VPN.

Při budování VPN je tedy potřeba dbát také na fyzickou a administrativní bezpečnost, zbudovat kvalitní dohled nad klienty, analyzovat jejich chování a vynucovat si bezpečnostní politiky. V případě vzdáleného přístupu se totiž často do VPN připojuje soukromé zařízení uživatele, které může být například zavirované.

Je proto nutné vymezit co mohou uživatelé v rámci VPN dělat a zamezit útokům z vnitřní strany VPN. VPN komunikace může být také často díky šifrování skrytá firewallu (záleží na architektuře sítě), nicméně provoz z VPN nelze nikdy považovat za bezpečný.

Z pohledu bezpečnosti se dají VPN označit jako „vrátka“ do vnitřní sítě a každá taková vrátka přináší určitá rizika.

2 Standard Internet Protocol Security

Internet Protocol Security, zkráceně používaný jako IPsec, je IETF standard, který definuje virtuální privátní síť na IP protokolu. IPsec není vázán žádnou konkrétní šifrovací a autentizační technologií, algoritmem pro výměnu klíčů či jinými bezpečnostními algoritmy.

IPsec je framework (struktura) otevřených standardů, která definuje pravidla pro bezpečnou komunikaci. IPsec je složen z existujících šifrovacích algoritmů, autorizačních algoritmů a algoritmů pro bezpečnou výměnu klíčů.

IPsec pracuje na síťové vrstvě a zajišťuje ochranu IP paketů a jejich autorizaci mezi jednotlivými účastníky komunikace (tzv. peery). Dokáže zabezpečit data libovolné aplikace, neboť zabezpečení může být implementováno od 4. do 7. síťové vrstvy.

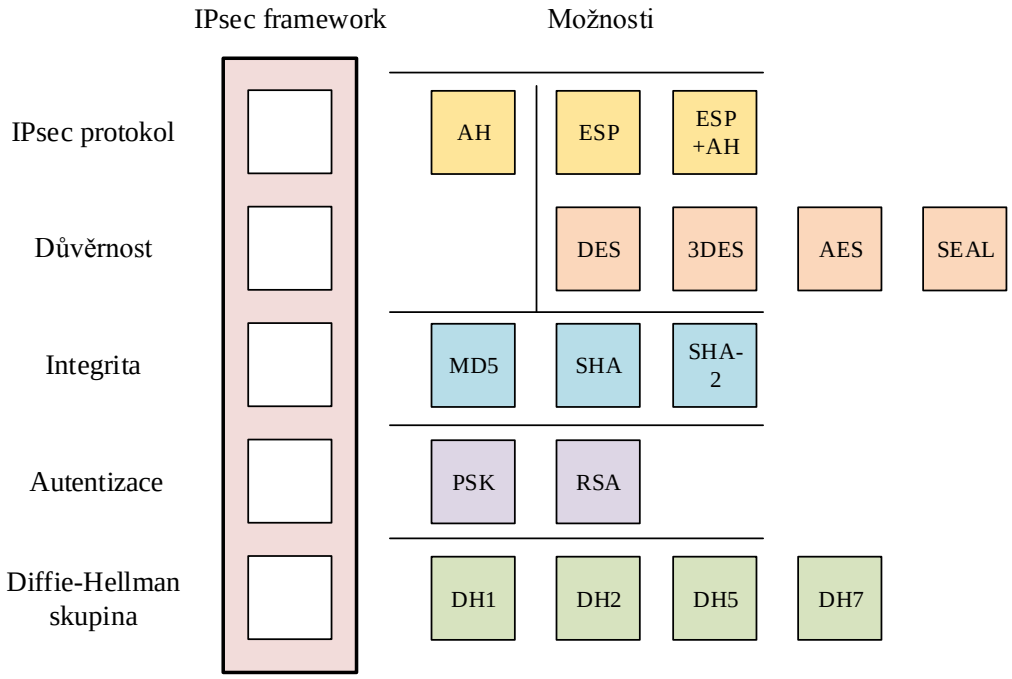
Protože pracuje na 3. síťové vrstvě, může být provozován na libovolné 2. síťové vrstvě, jako je Ethernet, ATM, Frame Relay a další.

2.1 Základní stavební kameny frameworku IPsec

Framework IPsec se skládá z pěti základních stavebních kamenů. Jsou jimi:

- IPsec protokol,
- důvěrnost,
- integrita,
- autentizace,
- skupina pro algoritmus Diffie-Hellman.

Jednotlivé prvky IPsec frameworku jsou vysvětleny v následujících kapitolách. Nutno podotknout, že algoritmy na obrázku 4 nejsou všechny povinnou součástí implementace IPsecu a záleží na výrobci, které do své implementace zahrne.



Obrázek 4 - Základní stavební kameny IPsec frameworku

IPsec dovoluje administrátorovi rozhodnout, jak budou data zabezpečena a to ze dvou pohledů:

- administrátor rozhoduje o bezpečnostním protokolu (AH či ESP), módu (tunelovací či transportní) a kryptografických algoritmech,
- administrátor zvolí granularitu, podle které se budou aplikovat bezpečnostní pravidla. Buď je vytvořen jeden tunel pro všechna spojení, či může být tvořen vždy nový tunel pro každé jednotlivé TCP spojení mezi danými hosty.

2.1.1 Kryptografické požadavky na IPsec

Algoritmy uvedené na obrázku 4 nejsou všechny povinnou součástí implementace IPsecu. Obrázek vychází z implementace společnosti Cisco. Standardizované kryptografické požadavky pro ESP a AH protokoly předepisuje RFC 4835 z roku 2007.

Dokument specifikuje, jaké algoritmy MUSÍ být implementovány, aby mohly spolupracovat dvě nezávislé implementace (tj. vždy budou implementovat min. jeden stejný algoritmus) a také specifikuje algoritmy, které BY MĚLY být implementovány.

Požadavky na autentizační algoritmy u AH

Pro implementaci AH protokolu jsou definovány tři algoritmy, přičemž pouze jeden z nich je povinný.

Tabulka 1 - Autentizační algoritmy pro AH dle RFC 4835

Požadavek	Algoritmus
Musí	HMAC-SHA1-96
Měl by	AES-XCBC-MAC-96
Může	HMAC-MD5-96

Požadavky na autentizační a šifrovací algoritmy u ESP

Požadavky na implementaci algoritmů při použití protokolu ESP shrnují následující dvě tabulky. V první tabulce jsou obsaženy šifrovací algoritmy, v druhé autentizační.

Tabulka 2 - Šifrovací algoritmy pro ESP dle RFC 4835

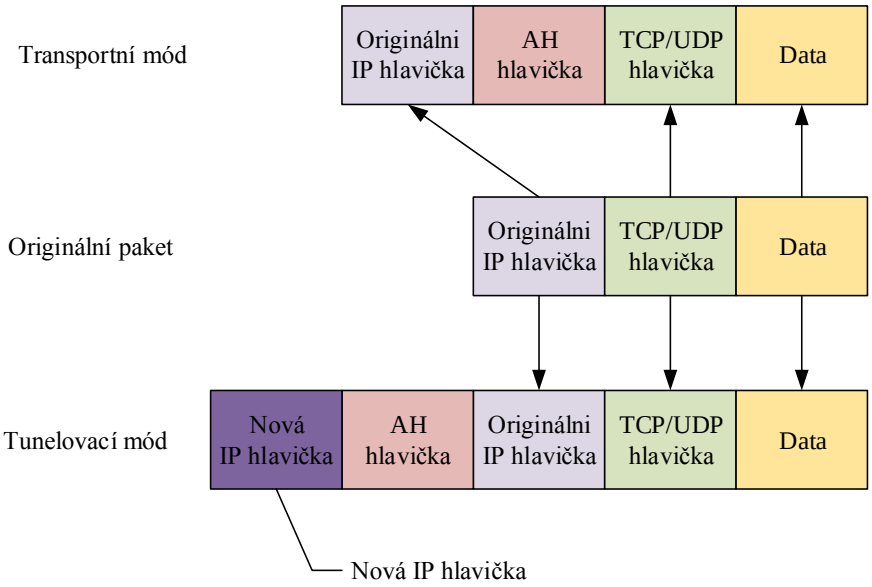
Požadavek	Šifrovací algoritmus
Musí	NULL (s daty se nic neděje)
Musí	AES-CBC se 128 bitovým klíčem
Musí	3DES CBC
Měl by	AES-CTR
Neměl by	DES-CBC

Tabulka 3 - Autentizační algoritmy pro ESP dle RFC 4835

Požadavek	Autentizační algoritmus
Musí	HMAC-SHA1-96
Měl by	AES-XCBC-MAC-96
Může	NULL (žádný)
Může	HMAC-MD5-96

2.2 Transportní s tunelovací režim protokolu IPsec

IPsec podporuje dva režimy – transportní a tunelovací. V transportním režimu chrání IPsec pouze protokoly vyšší vrstvy, v tunelovacím chrání celé IP pakety. Rozdíl je v tom, kam se umisťuje hlavička vybraného IPsec protokolu.



Obrázek 5 - Stavba paketu v transportním a tunelovacím módu (pro protokol AH)

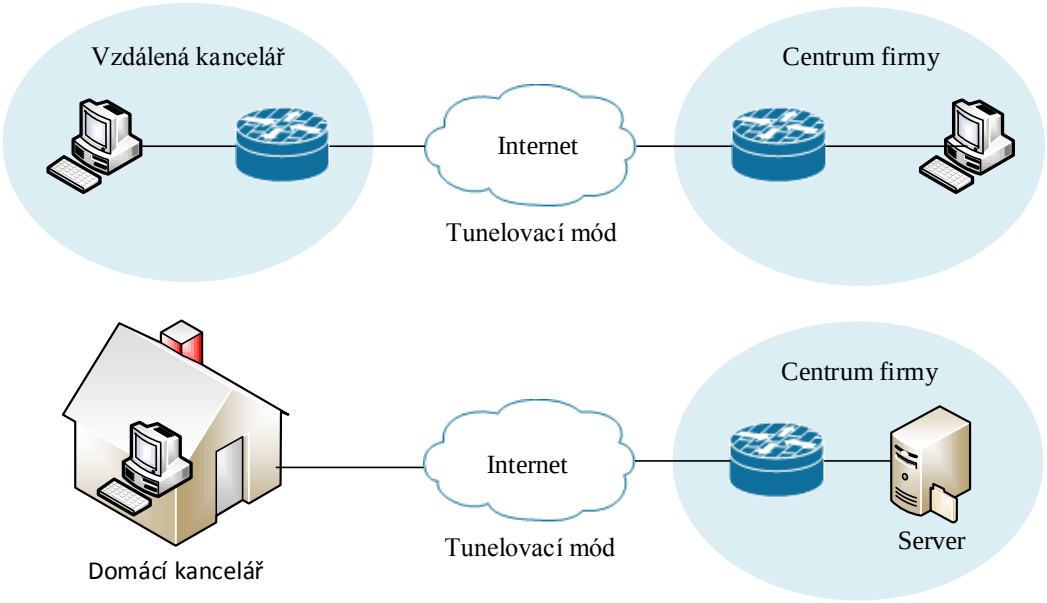
Jak je vidět z obrázku výše, tak v transportním módu je IPsec protokolem zpracována pouze užitečná část IP datagramu (IP payload) a IPsec hlavička se vkládá mezi IP hlavičku a hlavičku protokolu vyšší vrstvy. Původní IP hlavička datagramu je zachována v plaintextu.

V případě tunelovacího režimu je původní IP datagram spolu s původní IP hlavičkou zapouzdřen do zcela nového IP datagramu a je přidána nová IP hlavička.

Protokoly AH i ESP podporují oba módy. Detailní stavba paketu je vysvětlena v kapitolách věnovaných jednotlivým protokolům.

Použití tunelovacího a transportního módu

Tunelovací mód IPsecu se využívá, jak již název napovídá, k vytvoření zabezpečeného tunelu a to obvykle mezi dvěma VPN bránami (site-to-site VPN) nebo mezi klientským počítačem a VPN bránou (remote access VPN).



Obrázek 6 - Použití IPsecu v tunelovacím módu

V transportním módu není chráněna IP hlavička a obvykle se využívá pro zabezpečenou komunikaci dvou hostů. Často se využívá také ve spojení s GRE tunelem, který zamaskuje IP adresu z nechráněné hlavičky.

Obrázek 7 - Použití IPsecu v transportním módu

2.3 Bezpečnostní asociace – SA

Protože IPsec je poměrně komplexní a nabízí k ochraně dat řadu algoritmů, musí se uzly, které spolu chtějí komunikovat, nejprve dohodnout na bezpečnostní politice ve formě bezpečnostní asociace (anglicky Security Association – SA). Koncept SA je popsán v RFC 4301.

Bezpečností asociace je tzv. jednosměrná, v případě obousměrné komunikace mezi dvěma hosty musí být vytvořeny bezpečnostní asociace dvě, pro každý směr jedna. Bezpečnostní asociace je jednoznačně identifikována pomocí SPI (Security Parameter Index), cílové IP adresy a bezpečnostního protokolu (AH či ESP). Pokud je použit protokol ESP i AH, je nutno využít více bezpečnostních asociací najednou.

Parametry bezpečnostních asociací mohou být nastaveny buď manuálně, nebo automaticky pomocí protokolu IKE.

2.3.1 Databáze pro zpracování a uchování bezpečnostních asociací

Pro správu a uložení bezpečnostních asociací jsou definovány tři databáze – SPD (Security Policy Database), SAD (Security Association Database) a PAD (Peer Authorization Database).

Jak SPD, tak SAD může (ale nemusí) být více. Oddělené databáze mohou být využity pro příchozí a odchozí směry či pro každé rozhraní. PAD databáze existuje pouze jedna pro každé zařízení.

Databáze SPD

Databáze SPD specifikuje politiky, podle kterých se rozhoduje, jak se s daným datagramem naloží. Všechna příchozí i odchozí data musí být zpracována pomocí SPD a mohou nastat celkem tři situace:

- zahození datagramu,
- propuštění bez aplikace IPsec,
- aplikace IPsec.

V případě třetí volby, tedy aplikace IPsec na datagram, musí být v SPD databázi specifikováno, jaké mají být použity bezpečnostní služby, protokoly, algoritmy a další. Záznamy v SPD databázi musí být pevně seřazeny a pro zpracování datagramu se volí první padnoucí politika. V případě, že by stejný datagram přišel opětovně, musí být aplikována stejná politika.

SPD je logicky rozdělena na tři části:

- SPD-S,
- SPD-I,
- SPD-O.

SPD-S (secure traffic) obsahuje záznamy pro provoz, na který je aplikován IPsec. V části SPD-I jsou uloženy záznamy pro příchozí provoz, na který není aplikován IPsec. SPD-O (outbound) obsahuje záznamy pro provoz, který prochází běžným způsobem, tedy bez aplikace IPsecu.

Databáze SAD

Databáze SAD obsahuje záznam pro každou aktivní bezpečnostní asociaci a v každém záznamu jsou definovány parametry (SPI, zdrojové a cílové IP adresy a další) spojené s jednotlivými SA.

V SAD databázi se pro každou bezpečnostní asociaci uchovávají tato data:

- SPI neboli security parameter index, který identifikuje bezpečnostní asociaci,

- čítač sekvenčního čísla pro hlavičku AH či ESP paketů,
- indikátor pro přetečení čítače sekvenčního čísla (sequence counter overflow),
- čítač a bitová mapa pro detekci anti-replay útoků,
- autentizační algoritmus pro AH (pokud je IPsec protokolem AH) a daný klíč,
- šifrovací algoritmus pro ESP, klíč, inicializační vektor a další. Pole je obsaženo pouze pokud ESP poskytuje pouze šifrování a ne integritu dat,
- algoritmus pro zajištění integrity u ESP, klíč, mód, inicializační vektor a další. Pole je obsaženo pouze pokud ESP poskytuje pouze integritu dat a ne šifrování,
- algoritmy a klíče pro šifrování a integritu pro ESP. Toto pole je použito v případě kombinovaného módu u ESP (integrita + šifrování),
- životnost bezpečnostní asociace,
- stavové pole (flag) pro fragmentaci,
- DF (don't fragment) bit pro zákaz fragmentace,
- DSCP hodnoty (pro aplikaci QoS) povolené pro pakety v dané bezpečnostní asociaci. Pokud nejsou žádné hodnoty definovány, filtrace neprobíhá,
- bit pro zákaz použití DSCP,
- MTU (maximální přenosová jednotka) objevená po cestě k druhému uzlu,
- zdrojová a cílová adresa pro hlavičku tunelovacího paketu (pokud je IPsec v tunelovacím módu).

Databáze PAD

Poslední databází související se správou bezpečnostních asociací je PAD neboli Peer Authorization Database. PAD poskytuje spojení mezi SPD databází a protokolem pro navazování SA jako je například IKE.

PAD má několik základních funkcí:

- identifikuje jednotlivé uzly či skupiny uzlů, které jsou autorizovány pro komunikaci s IPsec uzlem,
- specifikuje protokol a metodu používanou pro autorizaci jednotlivých uzlů,
- obsahuje autentizační data pro jednotlivé uzly,
- omezuje vytváření bezpečnostních asociací pouze pro ověřené uzly,
- obsahuje informace o bráně uzlu (například IP adresu nebo DNS jméno), pokud je uzel za bezpečnostní bránou.

Záznamy v PAD databázi jsou v seřazeném pořadí. Nutnost seřazení je stejně jako v případě SPD databáze dána způsobem hledání záznamů a možností jejich překrývání.

Selektory

Selektory jsou parametry, podle kterých se volí jakou politiku a jakou SA pro daný datagram zvolit. Díky tomu může být spojení mezi dvěma hosty šifrováno různými metodami s využitím více bezpečnostních asociací.

Různé aplikace (pracující na různých protokolech či portech) tedy mohou využívat různé bezpečnostní asociace a tedy různý stupeň zabezpečení.

Každá IPsec implementace musí dle RFC 4301 využívat minimálně tyto selektory:

- vzdálená IP adresa či rozsah IP adres,
- lokální IP adresa či rozsah IP adres,
- protokol následující vrstvy,
- jméno (může být využito jako symbolický identifikátor pro lokální či vzdálenou adresu, například jméno uživatele či počítače, který inicioval spojení).

V případě, že protokol následující vrstvy využívá porty, jsou dalšími selektory lokální a vzdálený port. Pokud je protokolem následující vrstvy ICMP, jsou definovány další dva selektory – typ ICMP zprávy a její kód.

2.3.2 Zpracování příchozích a odchozích IP dat u IPsecu

Tato sekce se zabývá tím, jak zařízení pracuje s příchozími a odchozími daty a jak a kdy je na ně aplikován IPsec.

Zpracování odchozích dat

Při zpracování odchozích dat musí zařízení rozhodnout, jak s danými daty naloží. Existují tři možnosti, co se s daty stane:

- je aplikován IPsec (protect),
- data jsou propuštěna bez aplikace IPsecu (bypass),
- zahození dat (discard).

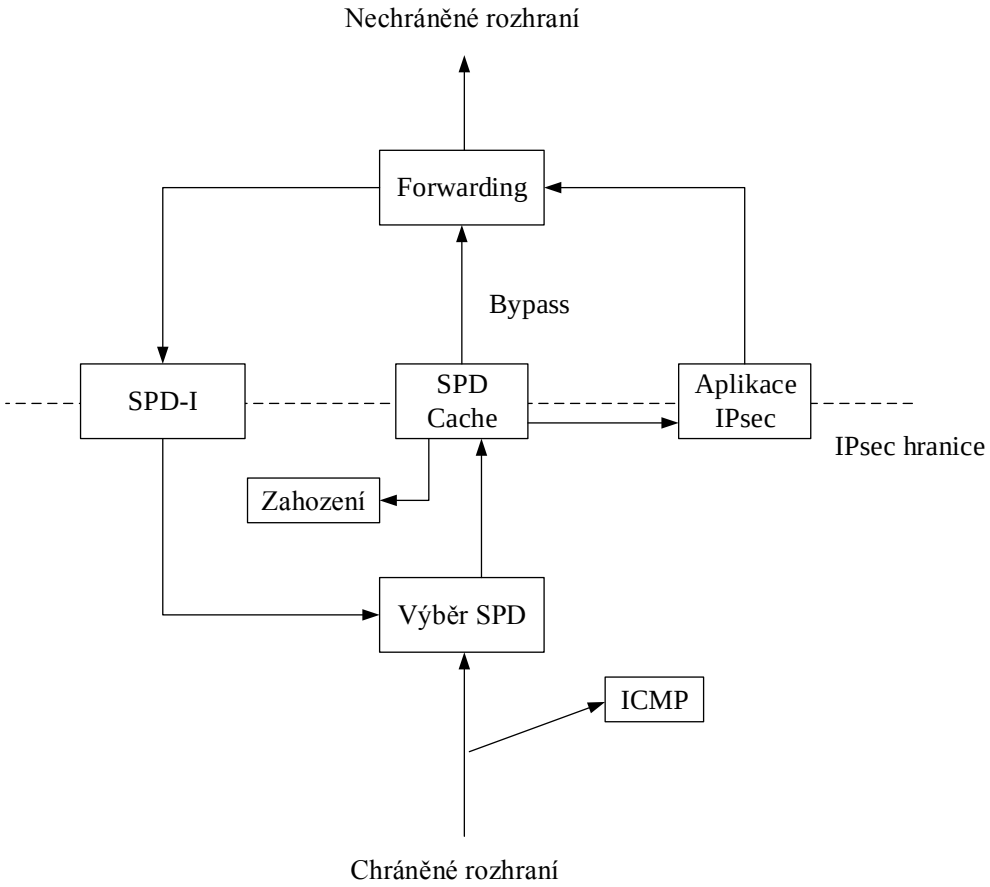
Při zpracování odchozích dat se předpokládá, že data do zařízení přichází z chráněného (protected) rozhraní a odchází do nechráněného rozhraní (unprotected). Typicky jde tedy o příchod dat na rozhraní vnitřní sítě (chráněné) a jeho opuštění skrze rozhraní do vnější sítě, obvykle Internetu (tj. nechráněné).

V takovém případě musí být na data aplikován tento postup:

1. Paket dorazí na chráněné rozhraní. Je zahájen výběr SPD databáze (databáze politik) pro daný provoz. Tento krok probíhá pouze v případě, že daná implementace podporuje více SPD databází, což podle nejnovějšího RFC 4301 nemusí. Pokud je definována pouze jedna databáze politik, tento krok neprobíhá.
2. Porovnání hlavičky paketu s aktuálními záznamy v cache SPD databázi vybrané v kroku 1. Cache obsahuje jen záznamy z databází SPD-O a SPD-S.
3. Krok tři může proběhnout dvěma způsoby:
    a. Záznam byl v cache nalezen a paket je zpracován, takže buď projde bez aplikace IPsecu, je zahozen, nebo je na něj aplikován IPsec.
    b. Pokud není záznam v cache, dojde k prohledání databází SPD-S a SPD-O. Pokud má být paket zahozen či má projít bez aplikace IPsecu, je vytvořen záznam v odchozí SPD cache. V případě průchodu bez aplikace IPsecu je

navíc vytvořen záznam pro příchozí SPD cache. Pokud má být chráněn pomocí IPsecu, je zahájen proces pro vytvoření bezpečnostní asociace (SA) například pomocí protokolu IKE. Pokud je bezpečnostní asociace úspěšně vytvořena, je přidán záznam do odchozí SPD-S cache a jsou také vytvořeny adekvátní záznamy v SAD databázi. Paket, který inicioval vytvoření SA je buď zahozen, nebo rovnou zabezpečen nově vytvořenou asociací (RFC toto nedefinuje a záleží tak na implementaci). Pokud se nepodařilo SA vytvořit, paket je zahozen.

4. Paket je předán dále pro směrování a je zahájen výběr správného odchozího rozhraní, což už nijak nesouvisí s IPsecem. Může se ale stát, že paket díky tomu bude procesem IPsecu procházet znovu a v takovém případě musí existovat příchozí záznam v SPD-I databázi. Jinak by byl paket zahozen.



Obrázek 8 - Zpracování odchozích dat IPsecem

Zpracování příchozích dat

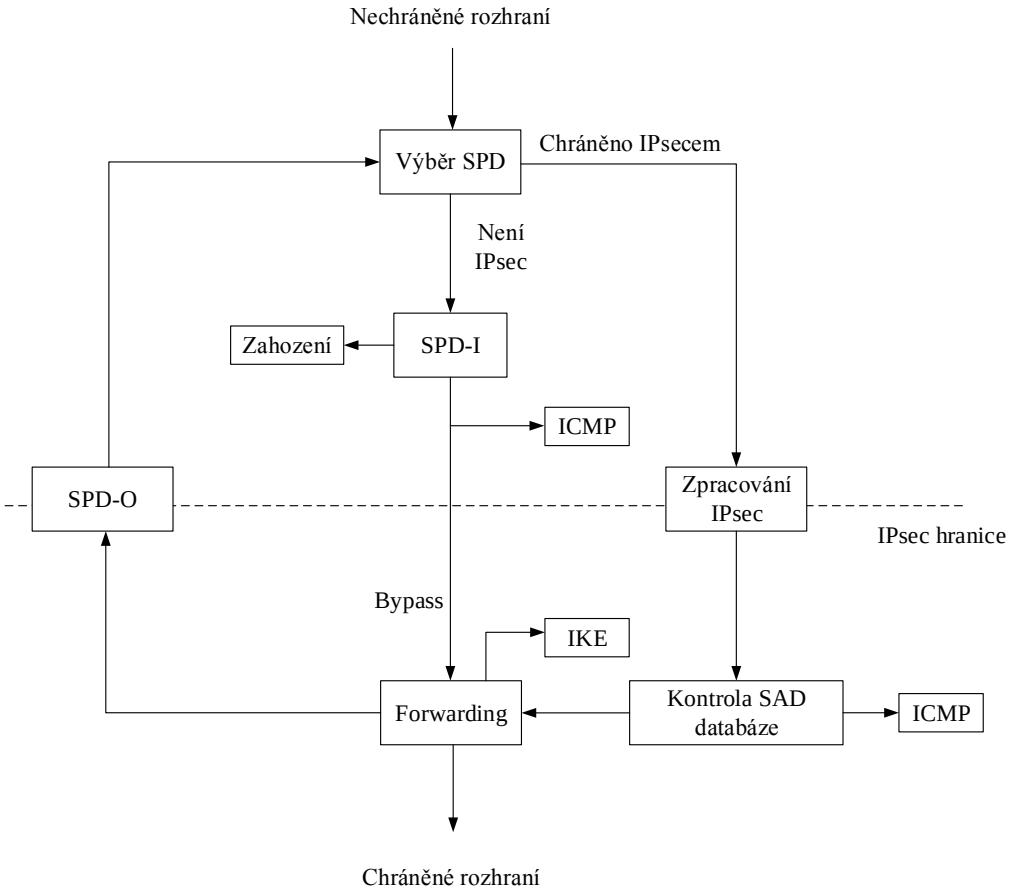
Zpracování příchozích dat je zpracování dat, která přicházejí přes nechráněné rozhraní (typicky Internet či WAN) a mají cíl ve vnitřní síti, tedy někde za chráněným rozhraním.

Příchozí data jsou tedy charakterizována jako data procházející od nechráněného rozhraní na chráněné rozhraní (anglicky unprotected-to-protected).

Zpracování příchozích dat se od zpracování odchozích dat liší v mnoha parametrech, například užitím cache SPD-I, která je aplikována pouze na data, která nejsou chráněna IPsecem či mají být zahozena.

Postup zpracování příchozích dat je popsán níže.

1. Po příchodu může být paket označen číslem (ID) síťového rozhraní, na které dorazil (fyzické nebo virtuální). Tento krok se provádí pouze v případě, že je na zařízení definováno více SPD databází. Rozhraní je v takovém případě mapováno na správnou databázi pomocí SPD-ID.
2. Druhý krok má několik variant.
    a. Pokud jsou data adresována danému zařízení a mají ESP nebo AH hlavičku, dojde k prohledání databáze SAD. Pokud není záznam nalezen, data jsou zahozena. Pokud je záznam nalezen, pokračuje se na krok 4.
    b. Pokud není paket adresován na dané zařízení a ani nemá AH či ESP hlavičku, je prohledána SPD-I cache. Pokud je nalezen záznam, je s paketem dle toho naloženo, tedy je buď zahozen, nebo propuštěn dále. Pokud není záznam v cache, prohledá se databáze SPD-I a v případě nalezení záznamu je v cache vytvořen záznam a paket zpracován. Pokud není záznam v SPD-I nalezen, je paket zahozen.
    c. Třetí variantou je příchod zprávy protokolu ICMP (Internet Control Message Protocol). Lokální politika definovaná v zařízení musí rozhodnout, jak s těmito zprávami naloží, tedy zda je zahodí či akceptuje.
3. Nalezený záznam z databáze SAD je přiřazen k bezpečnostní asociaci (SA) pomocí selektorů.
4. Pokud pole v hlavičce paketu nejsou konzistentní se selektory pro bezpečnostní asociaci, musí být paket zahozen. IKE také může v tuto chvíli vygenerovat informační zprávu INVALID_SELECTORS, kterou odešle druhému uzlu. Toho může být snadno využito pro DoS útok, proto by měl mít administrátor možnost odesílání této zprávy vypnout.
5. Poté, co jsou data propouštěna beze změny či zpracována pomocí IPsecu, jsou přendána na další směrování. To může způsobit opětovný průchod dat skrze IPsec zpracování. V takovém případě musí být tato odchozí data vždy propuštěna, jinými slovy, musí mít záznam v databázi SPD-O.
6. Paket je nakonec směrován k cílovému hostiteli.



Obrázek 9 - Zpracování příchozích dat IPsecem

3 IPsec protokol

Prvním základním kamenem IPsec frameworku je IPsec protokol. Administrátor může zvolit mezi protokolem Authentication Header (AH) a Encapsulating Security Payload (ESP). Protokoly se liší především v tom, jaké služby zajišťují.

Protokol AH zajišťuje:

- autentizaci,
- integritu.

Protokol ESP zajišťuje:

- důvěrnost dat,
- autentizaci,
- integritu.

Hlavním rozdílem je tedy to, že v případě protokolu AH jsou všechna data posílána jako plaintext, kdežto v případě protokolu ESP jsou šifrována vybraným algoritmem.

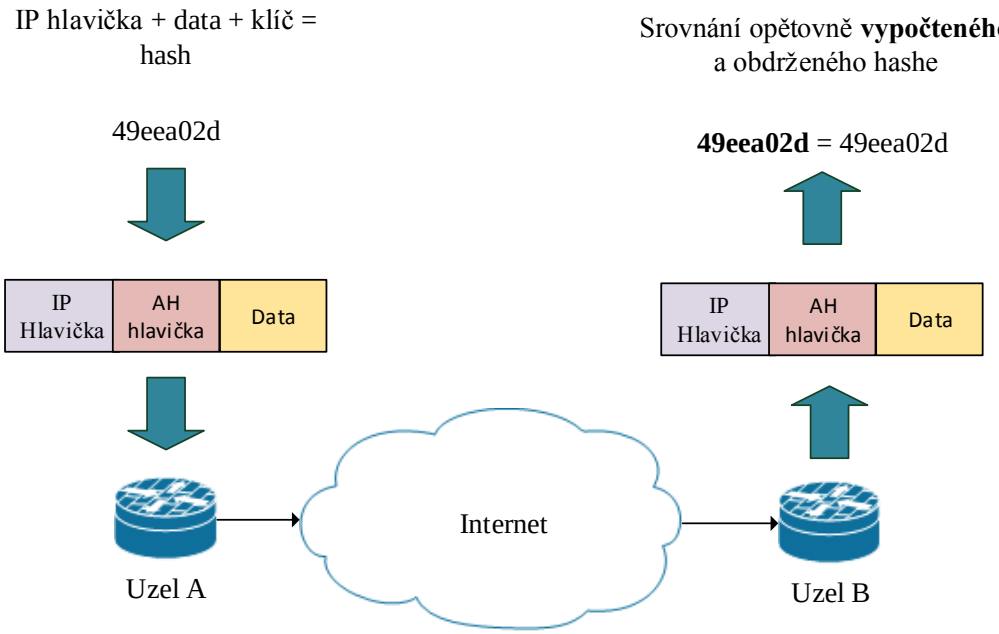
3.1 Protokol Authentication Header

IP protokol AH má číslo 51 a je podrobně popsán v RFC 4302. Z obrázku IPsec frameworku uvedeného na straně 24, je patrné, že při použití AH nelze nijak zajistit důvěrnost dat, neboť jsou přenášena nešifrovaně.

Protokol samozřejmě zajišťuje integritu IP datagramu, tedy to, že datagram nebyl útočníkem během přenosu zaměněn či upraven. Toho dosahuje vytvořením hashe (otisku) dat pomocí HMAC algoritmu.

AH pracuje ve 4 krocích:

1. Z IP hlavičky a dat je pomocí sdíleného tajného klíče vypočítán hash.
2. Je vytvořena nová AH hlavička, do které je vložen autentizační hash. Hlavička je poté vložena do originálního paketu.
3. Nový paket je přenesen na druhou stranu IPsec spojení.
4. Cílové zařízení nejprve spočítá hash z IP hlavičky a dat použitím stejného algoritmu a hashovací funkce se stejným klíčem. Poté rozbalí hash z příchozího paketu a srovná tyto dva hashe.



Obrázek 10 - Princip funkce protokolu AH a zajištění integrity

Hashe musí být shodné, v opačném případě byl paket modifikován a je zahozen. Hashovací funkce je tedy aplikována na celý paket s několika výjimkami. Jednou z nich je pole TTL, které se snižuje s každým průchodem směrovače. Pokud by byla hashovací funkce použita i na TTL pole, hashe by se nikdy nemohly shodovat. Protože je však hashovací funkce použita i na IP adresu, má AH problémy při průchodu skrze předklad adres (NAT).

Hlavička protokolu AH

Aby mohl protokol AH zajistit integritu IP datagramu, přidává k paketu AH hlavičku, která je zobrazena na obrázku níže.

Next header	Payload length	Reserved
Sequence parameter index (SPI)		
Sequence number		
Integrity check value		

Obrázek 11 - Hlavička protokolu AH

AH hlavička se skládá z celkem 6 polí – next header, payload lenght, reserved, security parameter index, sequence number a integrity check value (ICV). Délky jednotlivých polí jsou zobrazeny níže v tabulce. Všechna tato pole jsou v AH hlavičce povinná.

Tabulka 4 - Velikost jednotlivých polí v hlavičce protokolu AH[3]

Typ pole	Počet bajtů	Povinné
Next header	1	Ano
Payload length	1	Ano
Reserved	2	Ano
SPI	4	Ano
Sequence number	4	Ano
ICV	Proměnný	Ano

3.1.1 Popis jednotlivých polí AH hlavičky

V následující sekci jsou popsány jednotlivé pole AH hlavičky:

- **Next header** identifikuje typ následující hlavičky za AH hlavičkou. Hodnota udává číslo IP protokolu definované organizací IANA. Například číslo 4 označuje IPv6, 41 IPv6 či číslo 6 značí hlavičku protokolu TCP.
- **Payload lenght** označuje délku užitečné části paketu v bajtech, která se liší v závislosti na velikosti ICV.
- Pole **Reserved** je rezervováno pro budoucí účely. Odesílatel jej musí nastavit na nuly a příjemce by jej měl ignorovat.
- Pole **Security Parameters Index** identifikuje bezpečnostní asociaci (SA - Security Association) patřící k příchozímu paketu.
- **Sequence Number** obsahuje sekvenční číslo paketu pro každou SA. Odesílatel navyšuje toto číslo s každým odeslaným paketem. Sekvenční číslo chrání proti útokům anti-replay (opětném odeslání zachycených dat).
- Pole **Integrity Check Value** je pole proměnné délky, které obsahuje hash ze zvoleného integritního algoritmu, který je počítán odesílatelem a kontrolován příjemcem. Výpočet hashe je popsán níže.

3.1.2 Výpočet integrity pro pole Integrity Check Value

Aby mohl protokol AH zajistit integritu paketu, počítá podle zadaného algoritmu hodnotu ICV. Ta dle RFC 4302 k výpočtu využívá:

- pole IP hlavičky nacházející se před AH hlavičkou, která jsou považována za neměnná či předpověditelná,
- AH hlavičku (všechna pole včetně ICV, které je pro výpočet nastaveno na nulu),
- vše za AH hlavičkou (předpokládá se, že tento obsah je během přenosu neměnný),
- nejvýznamnější bity ESN (Extended Sequence Number), pokud je použito.

Hodnota ICV je poté počítána pomocí zadaného integritního algoritmu se zadaným tajným klíčem.

[3] KENT, S. RFC 4302: IP Authentication Header. In*: IETF Tools* [online]. 2005 [cit. 2013-05-01]. Dostupné z: http://tools.ietf.org/html/rfc4302

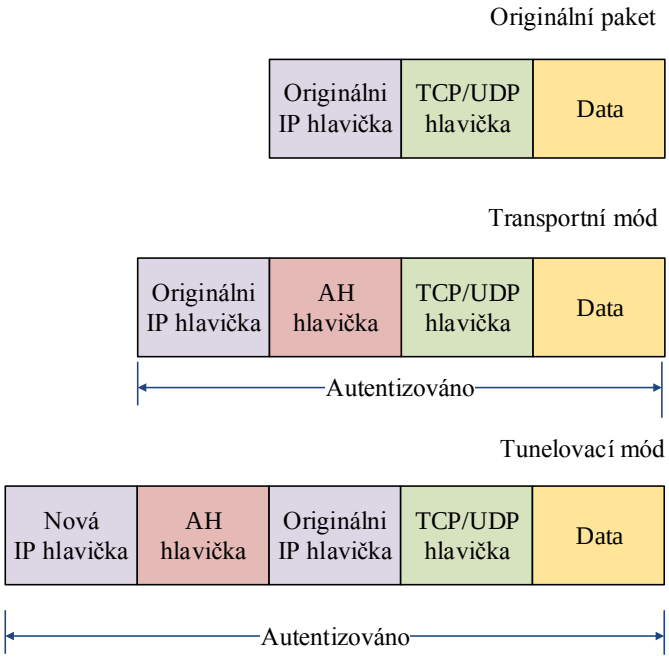
Protože mezi pole IP hlavičky, které jsou považovány za neměnné, patří IP adresy, nedovoluje AH použití překladu adres (NAT), který IP adresy v datagramu mění a ICV se tak stává neplatné. To je možné obejít pomocí rozšíření technologie NAT-Traversal.

Vkládání AH hlavičky do původního paketu

Umístění hlavičky do původního paketu se liší podle toho, zda je nastaven transportní či tunelovací mód IPsecu.

Při použití transportního módu se AH hlavička vkládá mezi IP hlavičku a hlavičku vyšších vrstev. Autentizován je tak celý paket, vyjma měnitelných polí v originální IP hlavičce (například TTL).

V tunelovacím módu je situace mírně odlišná, neboť AH hlavička se vkládá před originální IP hlavičku. Před vloženou AH hlavičku je poté vložena nová IP hlavička. Autentizován je opět celý paket, vyjma měnitelných polí v nové IP hlavičce. Původní hlavička je autentizována celá.



Obrázek 12 - Vkládání AH hlavičky v transportním a tunelovacím módu

3.2 Protokol Encapsulating Security Payload

Protokol ESP má přidělené číslo 50 a podrobně jej popisuje dokument RFC 4303. Použití ESP zajišťuje jednu z následujících možností:

- důvěrnost dat,
- integritu dat,
- důvěrnost a integritu dat najednou.

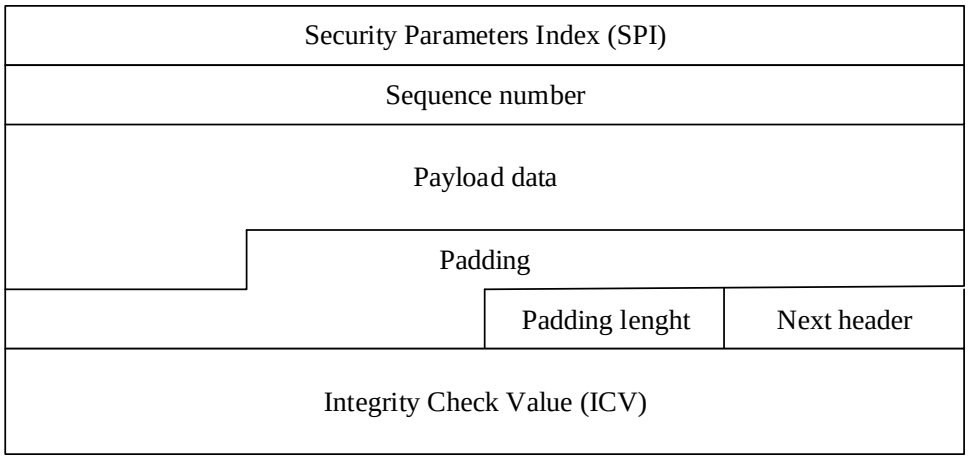
Typickým použitím je zajištění důvěrnosti i integrity najednou, což zajišťuje vyšší bezpečnost. Pokud je zvoleno zajištění integrity dat, chrání protokol také proti útokům anti-replay.

3.2.1 Popis ESP paketu

K původnímu paketu je přidávána ESP hlavička a také ESP trailer. O způsobu přidání hlavičky a traileru do paketu rozhoduje, zda je použit tunelovací či transportní mód IPsecu. Hlavička se přidává před původní paket a trailer za. Obecný koncept je zobrazen na obrázku níže.

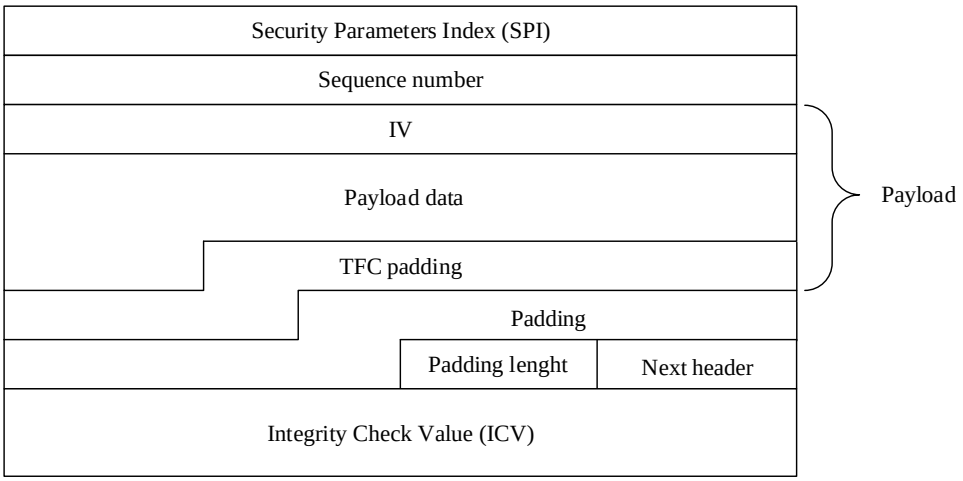
Obrázek 13 - Obecný obrázek stavby ESP paketu, přidání hlavičky a traileru

ESP paket se skládá ze 7 polí – SPI, sequence number, payload data, padding, pad length, next header a ICV. První dvě tvoří ESP hlavičku a pole za payload data tvoří trailer.



Obrázek 14 - Struktura ESP paketu

Pokud ESP zajišťuje důvěrnost dat (typicky ano) je pole payload data tvořeno substrukturou, která se skládá až z třech dalších polí – IV, payload data a TFC padding. Významy jednotlivých polí jsou vysvětleny níže.



Obrázek 15 - Substruktura payload data

Velikosti polí v ESP paketu

Hlavička ESP paketu je vždy dlouhá osm bajtů, neboť ji tvoří pouze dvě povinná pole. Délka ESP traileru poté záleží na délce vnitřních polí a není vždy stejná. Tabulka níže uvádí délky jednotlivých polí ESP paketu.

Tabulka 5 - Délky jednotlivých polí v ESP paketu[4]

Typ pole	Počet bajtů	Povinné
SPI	4	Ano
Sequence number	4	Ano
IV	Proměnný	Ne
IP datagram	Proměnný	Ano
TFC padding	Proměnný	Ne
Padding	0-255	Ano
Pad length	1	Ano
Next header	1	Ano
ICV	Proměnný	Ano

3.2.2 Popis jednotlivých polí ESP paketu

ESP paket může obsahovat až 9 polí, z čehož je 7 povinných a 2 volitelné. Níže jsou popsány významy jednotlivých polí:

- **Security Parameters Index (SPI)** definuje stejně jako u protokolu AH bezpečnostní asociaci v databázi SAD patřící k příchozímu paketu.
- **Sequence number** má stejný význam jako u protokolu AH. Zvyšuje se s každým odesílaným paketem sdruženým s danou SA a chrání před znovu odesláním

[4] KENT, S. RFC 4303: IP Encapsulating Security Payload (ESP). In: *IETF Tools* [online]. 2005 [cit. 2013-05-01]. Dostupné z: http://tools.ietf.org/html/rfc4303

zachycených dat (anti-replay útok). Každý přijatý paket s nižším sekvenčním číslem, než nejvyšší naposledy přijaté, je zahozen.

- **Payload data** je pole proměnné délky, které obsahuje data z originálního IP paketu. Pole může být rozděleno pomocí substruktury na tři:
    - Pokud algoritmus použitý pro šifrování dat vyžaduje synchronizační data, jsou tato data obsažena v přidaném poli IV (inicializační vektor). Pro samotný protokol ESP je přenos IV zcela transparentní.
    - Data.
    - Pole TFC neboli traffic flow confidentiality padding je přidáváno v případě, kdy je potřeba chránit datový tok. Volný překlad může znít například „služba pro důvěrnost toku dat“. Služba má zabránit tomu, aby útočník dokázal ze znalosti toku dat odvodit důležité informace (např. adresy a délky zpráv, intervaly mezi zprávami) o přenášených datech. Samotné pole padding je omezeno na 255 bajtů a to k ochraně toku dat nemusí stačit, proto je toto pole přidáváno mezi data.
- **Padding** neboli výplň slouží pro doplnění dat, například v případě použití blokové šifry, která vyžaduje bloky o stejné velikosti. Výplň se také může použít v případě (bez ohledu na použitou šifru), že šifrovaný text je třeba zarovnat na násobek velikosti 4 bajtového slova.
- **Pad length** udává velikost paddingu od 0 do 255 bajtů.
- **Next header** udává typ hlavičky pro nesená data v poli payload data, například číslo 4 pro IPv4 či 41 pro IPv6 paket.

3.2.3 Výpočet hodnoty pro ověření integrity

Integrita zajišťuje, že data nebyla během přenosu změněna. Odesílatel vypočte hash a přidá jej k ESP paketu do pole ICV (integrity check value). Příjemce spočítá hash z přijatých dat a srovná jej s přijatým hashem. Pokud se hashe shodují, je si příjemce jist, že data nebyla během přenosu pozměněna.

ESP protokol pro výpočet hashe využívá:

- ESP hlavičku,
- data (payload),
- ESP trailer.

Délka tohoto pole je proměnná a specifikuje ji použitý algoritmus, obsažený v bezpečnostní asociaci pro vybrané IPsec spojení.

3.2.4 Přádávání ESP hlavičky a ESP traileru k původnímu paketu

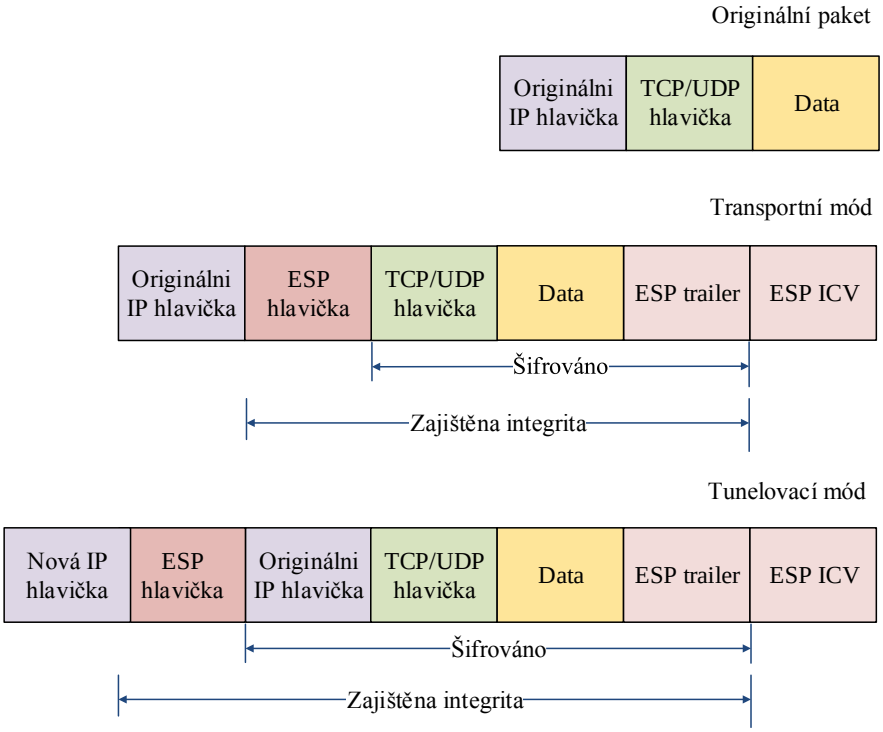
Stejně jako v případě protokolu AH se přidávání hlavičky protokolu ESP řídí tím, zda je použit tunelovací či transportní režim IPsecu.

V transportním módu se ESP hlavička vkládá mezi originální IP hlavičku a hlavičku protokolu vyšší vrstvy, tedy stejně jako v případě AH. Za originální IP paket je poté přidán

ještě ESP trailer a hodnota pro kontrolu integrity. V takovémto případě je šifrován celý původní paket bez IP hlavičky a ESP trailer. Autentizována je navíc ESP hlavička.

Tunelovací mód je taktéž podobný jako v případě AH protokolu. ESP hlavička se vloží před původní IP hlavičku. Před ESP hlavičku je poté přidána nová IP hlavička. Umístění ESP traileru je stejné jako v případě transportního módu.

Šifrován je tedy celý původní paket včetně IP hlavičky a ESP trailer. Integrita je zajištěna i pro hlavičku ESP.



Obrázek 16 - Přidávání ESP hlavičky a traileru k paketu, transportní a tunelovací mód

4 Zajištění důvěrnosti

Důvěrnost dat znamená, že nikdo krom příjemce je nebude moci přečíst. Důvěrnosti lze dosáhnout jen s použitím protokolu ESP, při samotném použití AH to možné není (v takovém případě je zajištěna pouze integrita dat, data mohou být přečtena, ale nemohou být změněna).

Důvěrnost dat je tedy zajištěna jejich kompletním šifrováním při průchodu VPN. Stupeň bezpečnosti pak závisí na délce šifrovacího klíče a volbě šifrovacího algoritmu.

Pokud by se útočník pokusil o útok hrubou silou, zabralo by rozšifrování dat při použití 56 bitového klíče a rychlosti 10^6 klíčů za mikrosekundu přibližně 10 hodin. Pokud by se snažil rozšifrovat data se 128 bitovým klíčem, trvalo by to při stejné rychlosti $5{,}9 * 10^{18}$ let[5].

Výběr algoritmů záleží na implementaci výrobce, neboť v RFC jsou předepsány pouze některé. V této kapitole jsou tedy rozebrány pouze vybrané algoritmy. Výrobce může implementovat v podstatě libovolný algoritmus. Kryptografické požadavky pro IPsec jsou rozebrány v kapitole 2.1.1.

Následující část se tedy zabývá těmito vybranými algoritmy:

- DES (mód CBC),
- 3DES (CBC),
- AES (CBC a CTR mód),
- SEAL (pouze někteří výrobci, např. Cisco).

4.1 Algoritmus DES

DES neboli Data Encryption Standard je symetrický šifrovací algoritmus, který byl v roce 1977 přijatý za standard organizací NIST. Data jsou v případě tohoto algoritmu šifrována po 64 bitových blocích s použitím 56 bitového klíče (respektive 64 bitového klíč, přičemž má pouze 56 významných bitů a osmý bit je vždy paritní).

Algoritmus transformuje 64 bitová vstupní data do 64 bitového výstupu v několika krocích. Protože se jedná o symetrický algoritmus, je stejný postup využit i k dešifrování dat.

Síla ochrany DES

DES byl předmětem kritiky ještě dříve, než se stal standardem. Vyvinul se z algoritmu LUCIFER, na kterém pracovala IBM a který používal 128 bitový klíč. V případě DES byl však klíč zredukován o 72 bitů na pouhých 56 bitů. Kritici namítali, že klíč není pro útoky hrubou silou dost dlouhý. Kritizována byla také vnitřní struktura algoritmu, postavená na

[5] STALLING, William. Cryptography and Network Security: Principles and Practice. 5. vyd. New York: PearsonEducation, 2010, s. 65.

tzv. S-boxech, jejichž struktura nebyla tehdy zveřejněna a mohla tak obsahovat skryté slabiny.

Protože DES používá 56 bitový klíč, existuje pouze 2^{56} klíčů. Jak je uvedeno v úvodu kapitoly, útok hrubou silou s rychlostí 10^6 klíčů za mikrosekundu by trval přibližně 10 hodin. Dosáhnout takové rychlosti je však velmi náročné.

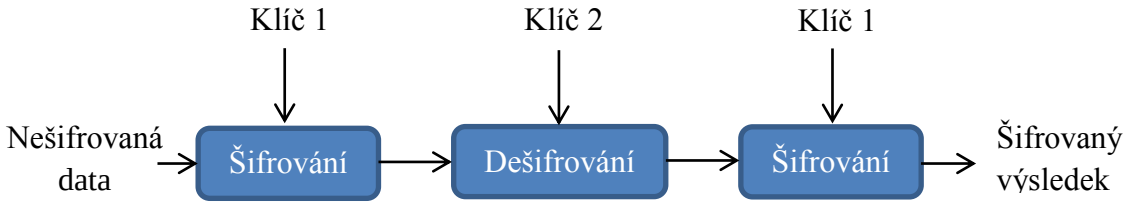
V roce 1998 oznámila nadace EFF, že se jí pomocí speciálního „DES crackeru" provádějícího útok hrubou silou podařilo rozšifrovat DES během pouhých tří dnů a DES se tak v podstatě okamžitě stal nepoužitelným algoritmem pro zabezpečený přenos dat.

V současné době je tedy DES pro šifrování nepoužitelný a snadno zranitelný.

4.2 Algoritmus 3DES

Triple DES, zkráceně 3DES, představuje alternativu k algoritmu DES, která vznikla kvůli možnosti objevení slabiny v DES (což se ukázalo jako oprávněná obava). 3DES nevyužívá pouze trojnásobného zašifrování, ale pracuje jinak. Délka bloku je stejně jako v případě DES 64 bitů.

Pracuje buď s pomocí dvou nebo tří různých klíčů. Nejrozšířenější je varianta se dvěma šifrovacími klíči, varianta se třemi klíči je však velmi podobná. Nejprve proběhne zašifrování plaintextu pomocí prvního klíče. Výsledek této operace je dešifrován druhým klíčem a na závěr opět zašifrován prvním klíčem. Postup je vidět na níže uvedeném obrázku.



Obrázek 17 - Postup šifrování algoritmem 3DES

Síla ochrany 3DES

V případě použití dvou klíčů se délka klíče prodlužuje na 112 bitů, v případě třech pak na 168 bitů. V porovnání s původním DES algoritmem se tedy jedná o značné zvýšení bezpečnosti a v současné době nejsou známy žádné útoky, které by bezpečnost algoritmu 3DES narušily.

Problémem je však trojnásobná aplikace šifry DES, což je poměrně neefektivní a pomalé. Novější algoritmy jako je AES jsou mnohem rychlejší.

4.3 Algoritmus AES

Advanced Encryption Standard (AES) je symetrický blokový šifrovací algoritmus, který byl organizací NIST standardizován v roce 2001 a stal se tak lepší alternativou k DES a 3DES. AES může pracovat v pěti módech (CBC, ECB, CFB, OFB a CTR), nicméně RFC 4835 doporučuje při použití v IPsecu pouze CBC (povinný) a CTR mód (měl by být implementován).

AES využívá 128 bitové bloky a klíč může mít délku 128, 192 nebo 256 bitů. V porovnání s dalšími algoritmy je princip AES poměrně složitý a komplexní, výpočet je však velmi efektivní a v porovnání s konkurencí je AES nejen bezpečnější, ale také rychlejší.

AES pracuje v iteracích, jejichž počet závisí na délce klíče. Pro 128 bitový klíč provádí 10 iterací, pro 192 bitový klíč 12 iterací a pro 256 bitový klíč 14 iterací.

Rozdíly mezi CBC a CTR módem

Módy CBC a CTR se liší z pohledu toho, jak je daná šifra počítána. CTR má například výhodu v možnosti paralelizace výpočtu, což CBC neumí. Jak probíhá výpočet šifrovaného textu u jednotlivých módů, není z pohledu této práce podstatné a proto zde nejsou tyto detaily uvedeny. Z pohledu ESP jsou však důležité rozdíly v tom, jak vypadá payload ESP paketu.

V případě CBC módu se ESP payload skládá s 16 oketů inicializačního vektoru a poté následují šifrovaná data. V případě CTR módu je ESP payload složen ze třech polí. Nejprve obsahuje 8 oketů inicializačního vektoru, následují šifrovaná data a autentizační data.

Inicializační vektor (16 oketů)
Šifrovaný payload (proměnná délka, násobek 16 oketů)

Obrázek 18 - ESP payload v případě CBC módu

Inicializační vektor (8 oketů)
Šifrovaný payload
Autentizační data

Obrázek 19 - ESP payload v případě CTR módu

Síla ochrany AES

V roce 2013 nejsou na tento šifrovací algoritmus známé žádné úspěšné útoky, které by mohly ohrozit bezpečnost dat. Kryptografický expert Bruce Schneier uvádí[6], že útoky zatím nejsou možné, ale jsou představitelné. Tedy může se stát, že se podaří najít útok, který šifru částečně či úplně prolomí.

Bruce Schneier doporučuje, aby organizace NIST navýšila počet iterací u všech variant klíčů, čímž by šifra získala větší „bezpečnostní rezervu".

Přesto Schneier v současné době i dohledné budoucnosti považuje AES s nejslabším, tedy 128 bitovým, klíčem za více než dostatečnou ochranu dat.

4.4 Algoritmus SEAL

SEAL (Software Encryption Algorithm) je další alternativou k výše zmíněným algoritmům. Vyskytuje se pouze v některých implementacích IPsecu, neboť RFC 4835 jej nepředepisuje. Ve své implementaci jej používá například výrobce Cisco.

SEAL je symetrickou proudovou šifrou využívající 160 bitový šifrovací klíč. V porovnání s konkurenčními algoritmy má podle společnosti Cisco[7] menší nároky na procesorový čas.

Na algoritmus se vztahují dva patenty, oba patřící firmě IBM. Zřejmě díky tomu není SEAL tolik rozšířen, jako konkurenční algoritmy.

[6] SCHNEIER, Bruce. New Attack on AES. In: *Schneier on Security: A blog coveringsecurity and security technology* [online]. 2011 [cit. 2013-05-01]. Dostupné z: http://www.schneier.com/blog/archives/2011/08/new_attack_on_a_1.html

[7] NextGenerationEncryption. In: *Cisco System* [online]. 2012 [cit. 2013-05-01]. Dostupné z: http://www.cisco.com/web/about/security/intelligence/nextgen_crypto.html

5 Zjištění integrity

Další velmi důležitou součástí IPsecu je zajištění integrity dat. Jinými slovy jde o ochranu dat před možnou modifikací či podstrčením dat útočníkem. IPsec k tomuto účelu využívá mechanismus HMAC (Keyed-Hashing for Message Authentication) generující autentizační kód zprávy za pomocí tajného klíče.

HMAC popisuje RFC 2104 a mechanismus je možné využít s libovolným kryptografickým hashovacím algoritmem. V případě IPsecu je pro výpočet autentizačního kódu využívána celá řada algoritmů – HMAC-MD5, HMAC-SHA1, HMAC-SHA-256, HMAC-SHA-384 a HMAC-SHA-512. Pouze některé jsou povinnou součástí implementace a opět jej definuje RFC 4835.

Následující část se zabývá obecným konceptem MAC a HMAC algoritmů a poté algoritmy HMAC-SHA1-96, AES-XCBC-MAC-96 a HMAC-MD5-96.

5.1 MAC a HMAC

IPsec chrání integritu dat pomocí autentizačního kódu zprávy, anglicky Message Authentication Code neboli MAC. Pro generování MAC se využívá tajného klíče, který musí komunikující strany sdílet. MAC se také někdy přezdívá kryptografický kontrolní součet.

MAC lze popsat takto:

$$MAC = C(K, M)$$

kde:

M = vstupní zpráva
C = funkce pro výpočet MAC
K = sdílený tajný klíč
MAC = výsledný autentizační kód zprávy

Komunikace dvou uzlů A a B poté probíhá tak, že zdrojový uzel A odešle svou zprávu spolu s autentizačním kódem. Cílový uzel po přijetí zprávy vypočítá opětovně MAC a porovná jej s autentizačním kódem přirazeným ke zprávě.

Pokud uvažujeme, že pouze příjemce a odesílatel znají sdílený tajný klíč, je situace následující:

- Pokud vypočtený a přijatý autentizační kód souhlasí, může si být příjemce jistý, že zpráva nebyla nijak pozměněna. Pokud by ji totiž útočník změnil, nesouhlasil by vypočtený autentizační kód s přijatým. Pokud se útočníkovi podaří změnit i MAC, nemohl by ho bez znalosti tajného klíče spočítat správně a kód by nesouhlasil.
- Příjemce si tedy může být také jist, že zpráva je od správného odesílatele, neboť nikdo jiný nezná tajný klíč.

Funkce pro výpočet autentizačního kódu musí být jednosměrná. To znamená, že musí být snadné vypočítat autentizační kód, ale musí být velmi obtížné či nemožné získat z kódu vstupní data.

5.1.1 HMAC

HMAC je autentizační kód založený na některé hashovací funkci. V porovnání s MAC založenými na symetrických kryptografických funkcích má HMAC dvě hlavní výhody:

1. Hashovací funkce jako je MD5 či SHA jsou obecně rychlejší, než symetrické šifry jako je DES.
2. Jsou bez problémů dostupné a implementované v celé řadě knihoven.

Návrh HMAC je uzpůsoben tak, že HMAC přistupuje k hashovací funkci jako k černé skříňce. Tento přístup má dvě výhody:

1. Existující implementace hashovací funkce může být snadno použitá jako modul v implementaci HMAC.
2. Hashovací funkci v implementaci HMAC lze velmi snadno nahradit jinou, a to odebráním starého modulu a přidáním nového modulu, beze změny samotné implementace HMAC.

Hashovací algoritmy jako je MD5 a SHA nepracují s tajným klíčem, práce se samotným klíčem zůstává na mechanismu HMAC.

5.2 Algoritmy MD5 a HMAC-MD5-96

Algoritmus MD5 (Message-Digest ve verzi 5) je definován v RFC 1321 a jeho verze pro použití s HMAC v RFC 2403. Je jedním z nejpoužívanějších algoritmů pro výpočet hashe. Vstupní data mohou být libovolné délky a algoritmus z nich vypočítá 128 bitů dlouhý hash (otisk).

MD5 byl stejně jako každý hashovací algoritmus navržen jako bezkolizní. To znamená, že kolizi je velmi obtížné (respektive výpočetně neproveditelné) najít, nikoli že neexistuje. Vezmeme-li v potaz nekonečně mnoho vstupních dat, existuje teoreticky také nekonečně mnoho kolizí.

Jak se ale ukázalo, návrh nebyl bezchybný. V roce 2004 se podařilo několika kryptoanalytikům objevit kolidující zprávy pro čtyři hashovací funkce a funkce MD5 mezi nimi nechyběla[8]. Nalezení kolize je tedy možné a algoritmus byl oslaben.

Dle RFC 4835 neznamená nalezení kolizí problém při použití spolu s HMAC. Společnost Cisco však doporučuje tento algoritmus pro integritu dat již nepoužívat[9].

[8] KLÍMA, Vlastimil. Hašovací funkce MD5 a další prolomeny!. In: *Root.cz: Informace nejen ze světa Linuxu* [online]. 2004 [cit. 2013-05-01]. Dostupné z: http://www.root.cz/clanky/hasovaci-funkce-md5-a-dalsi-prolomeny/

[9] Next Generation Encryption. In: *Cisco System* [online]. 2012 [cit. 2013-05-01]. Dostupné z: http://www.cisco.com/web/about/security/intelligence/nextgen_crypto.html

HMAC-MD5-96

Tento algoritmus je založen na konceptu HMAC a algoritmu MD5. Bloky jsou v tomto případě velké 64 bitů a výsledný hash je dlouhý 128 bitů. Tento hash muže být ořezán až na 96 bitů (proto se v názvu nachází číslice 96). V takovém případě porovnává příjemce pouze prvních 96 bitů. Pro použití s ESP a AH je pro algoritmus definováno použití 128 bitového klíče.

5.3 Algoritmy SHA a HMAC-SHA1-96

SHA (Secure Hash Standard) byl vyvinut organizací NIST a publikován již v roce 1993. Tato verze je nyní známa jako SHA-0 a obsahuje několik bezpečnostních slabin. V roce 1995 prošel algoritmus revizí známou jako SHA-1. SHA je založen na hashovací funkci MD4.

SHA-1

Hashovací algoritmus SHA-1 je specifikován v RFC 3174 a vytváří hash dlouhý 160 bitů a maximální délka zprávy je omezena na 2^{64} bitů.

SHA-2

V roce 2002 provedla organizace NIST další revizi SHA a představila tři nové verze – SHA-256, SHA-384 a SHA-512 s délkou hashe 256, 384 a 512 bitů. V roce 2008 byla přidána také verze s délkou hashe 224 bitů. Tyto algoritmy jsou známy pod souhrnným názvem SHA-2.

SHA-2 má stejnou základní strukturu a používá stejnou aritmetiku a logiku operací jako SHA-1 a je specifikován v RFC 4634. Jednotlivé vlastnosti verzí algoritmu SHA jsou srovnány v níže uvedené tabulce. Kompletním popisem algoritmu se tato práce nezabývá.

Srovnání algoritmů

Všechny algoritmy z rodiny SHA jsou v současné době považovány za bezpečné a žádný z nich nebyl doposud prolomen. Pouze v případě algoritmu SHA-1 je popsán útok pro nalezení kolize se složitostí 2^{51}. Útok nebyl v době psaní této práce prozatím nikdy prakticky proveden.[10]

[10] MANUEL, Stéphane. Classification and Generation of Disturbance Vectors for Collision Attacks against SHA-1. In: *Cryptology ePrint Archive* [online]. 2008 [cit. 2013-05-01]. Dostupné z: http://eprint.iacr.org/2008/469.pdf

Tabulka 6 - Srovnání vlastností různých verzí SHA algoritmu[11]

Verze	SHA-1	SHA-224	SHA-256	SHA-384	SHA-512
Délka hashe	160	224	256	284	512
Délka zprávy	$<2^{64}$	$<2^{64}$	$<2^{64}$	$<2^{128}$	$<2^{128}$
Velikost bloku	512	512	512	1024	1024
Počet kroků algoritmu	80	64	64	80	80

HMAC-SHA-1-96

Algoritmus HMAC-SHA-1-96 využívá bloky dat dlouhé 64 bitů a produkuje hash o délce 160 bitů. Stejně jako v případě HMAC-MD5-96 může být tento hash ořezán na pouhých 96 bitů a použit pro kontrolu integrity. SHA-1 v HMAC módu musí mít pevnou délku klíče 160 bitů.

Na podobných principech jsou založeny také algoritmy MAC-SHA-256, HMAC-SHA-384 a HMAC-SHA-512.

5.4 Algoritmus AES a AES-XCBC-MAC-96

Obecný koncept algoritmu AES je popsán v kapitole 4.3 a tato část práce se zabývá pouze jeho použitím v HMAC módu.

Algoritmus AES-XCBC-MAC-96

Použití tohoto algoritmu ve spolupráci s IPsecem je definováno v RFC 3566 z roku 2003. Algoritmus pracuje ve speciálním XCBC módu, který vychází z CBC módu pro MAC. Jedním z hlavních rozdílů je, že AES-XCBC-MAC-96 dokáže zašifrovat zprávu o libovolné délce.

Tento algoritmus pracuje s bloky o délce 128 bitů a délka klíče je stanovena také na 128 bitů. Stejně jako v případě předchozích dvou algoritmů může být hash zkrácen až na 96 bitů. Příjemce poté počítá celý hash, ale srovnává jen prvních 96 bitů.

[11] STALLING, William. *Cryptography and Network Security: Principles and Practice*. 5. vyd. New York: Pearson Education, 2010, s. 343.

6 Autentizace

Autentizace je proces ověření udávané identity komunikujícího subjektu. Ve středověku se k takovému účelu používaly pečeti, alternativou v moderní době jsou digitálně podepsané dokumenty.

Pokud probíhá zabezpečené připojení, je kromě šifrování a integrity dat také nutné ověřit, zda komunikujeme s tím, s kým komunikovat chceme a nenavázali jsme náhodou šifrované spojení s útočníkem.

V případě VPN existují pro ověření identity tři hlavní volby:

- použití předsdíleného tajného klíče (PSK),
- digitální certifikáty,
- AAA autentizace s využitím protokolu EAP.

Autentizace uzlů v případě IPsecu probíhá v rámci IKE relace.

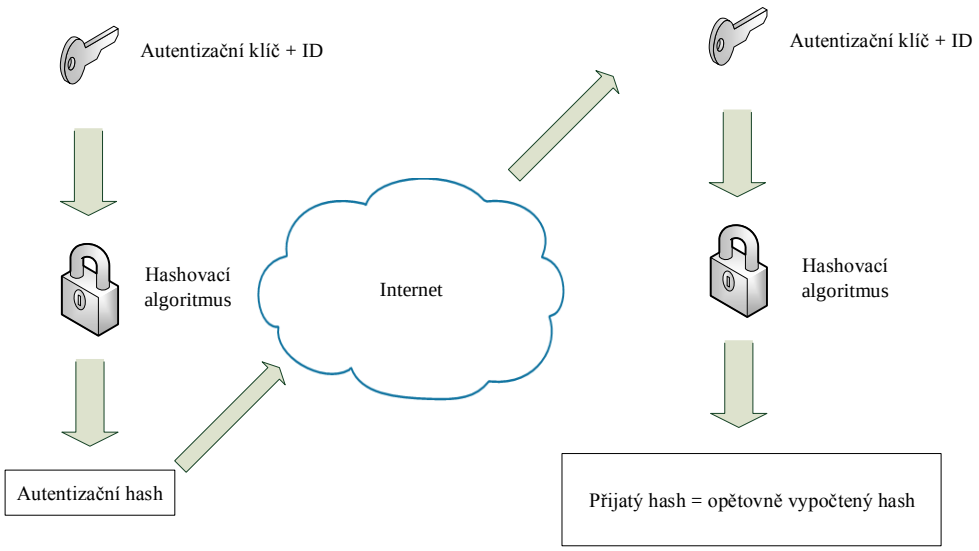
6.1 Předsdílený tajný klíč (PSK)

Předsdílený tajný klíč (pre-shared key) je klíč, který znají pouze komunikující uzly. Slouží pro autentizaci jednotlivých uzlů a na obou koncích spojení je zadáván manuálním způsobem při konfiguraci.

Z toho vyplývá, že je snadné jej nakonfigurovat, ale takové řešení není příliš dobře škálovatelné, neboť pro každé dvě komunikující strany je potřeba zadávat klíč. Pří více navázaných VPN spojení není tento přístup příliš efektivní.

Pro autentizaci není využíván pouze předsdílený klíč, ale také například ID daného uzlu. Pomocí těchto informací je vypočítán autentizační hash. Příjemce poté vypočítá autentizační hash a srovná jen s přijatým hashem. Pokud se shodují, identita druhého uzlu je považována za ověřenou.

Celý koncept tedy spoléhá na to, že pouze ověřený uzel zná sdílené tajné heslo a útočník jej nemůže zfalšovat.



Obrázek 20 - Autentizace uzlů pomocí předsdíleného klíče v rámci IKE relace

Tato volba je vhodná při komunikaci dvou uzlů. Při komunikaci více uzlů ale nastávají problémy. Kromě toho, že je všude potřeba konfigurovat stejné klíče, není řešení příliš škálovatelné. Druhou, podstatnější vadou je, že ve skupině uzlů se stejným klíčem nelze přesně autentizovat komunikující uzel (víme, že je ze skupiny, ale nevíme přesně který to je), neboť všechny uzly mají stejné předsdílené klíče a tedy se autentizují stejně. Tento problém řeší digitální certifikáty.

6.2 Digitální certifikáty

Digitální certifikát je digitálně podepsaný veřejný šifrovací klíč, který využívá asymetrické kryptografie. Certifikáty jsou uchovávány ve standardizovaném formátu X.509, který je popsán v RFC 6818.

Tato část práce se nejprve zabývá konceptem asymetrické kryptografie, poté digitálními podpisy a certifikáty a jejich použití spolu s IPsecem.

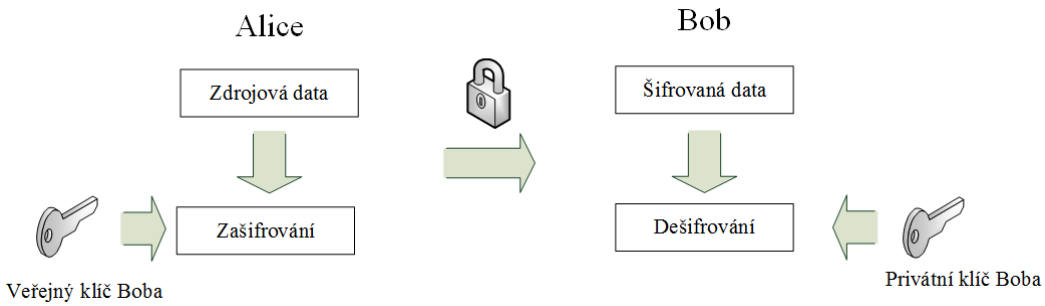
6.2.1 Asymetrická kryptografie

Digitální podpisy využívají asymetrické kryptografie (neboli šifrování veřejným klíčem), která využívá dva typy klíčů:

- veřejné,
- soukromé (privátní).

Data jsou šifrována pomocí veřejného klíče a mohou být rozšifrována pouze pomocí soukromého klíče. Veřejný klíč uzlu zná kdokoli, privátní klíč zná pouze uzel, kterému je přiřazen. Pokud útočník získá privátní klíč, může rozšifrovat data určená pro jiný uzel. Výhodou tohoto přístupu je, že uzly se nemusí vzájemně dohadovat na tajném klíči, který

slouží pro šifrování i dešifrování dat, ale jednoduše si vymění své veřejné klíče. Útočníkovi je totiž odposlechnutý veřejný klíč k ničemu.

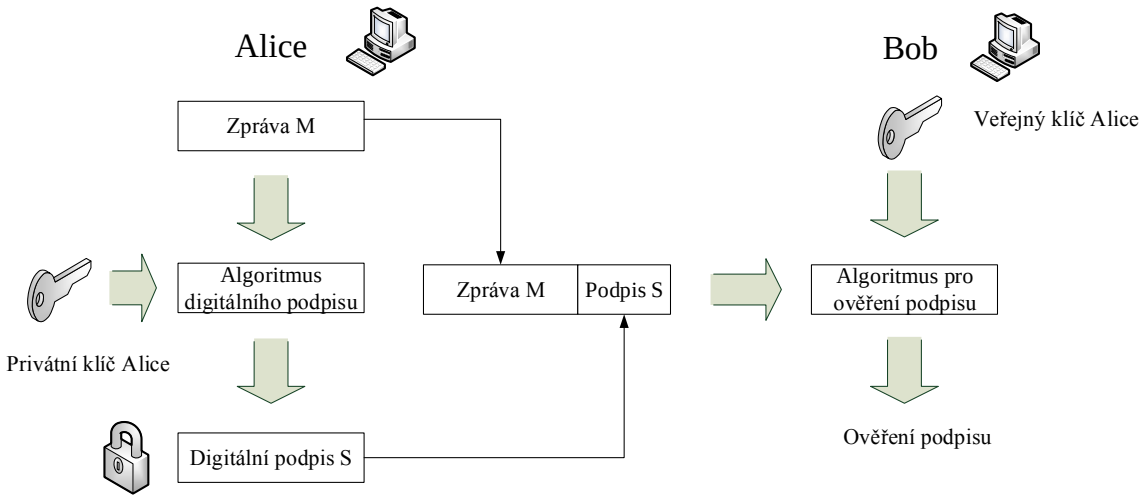


Obrázek 21 - Princip asymetrické kryptografie

6.2.2 Digitální podpisy

Digitální podpisy jsou založeny na asymetrické kryptografii a chrání komunikaci uzlů před zásahem třetí strany. Digitální podpis tedy zajišťuje integritu dat (nemohou být během přenosu nikým pozměněna), nepopíratelnost (uzel nemůže popřít, že data nepodepsal a nepochází od něho) a dokáží zajistit také autenticitu zprávy. Autenticita zprávy je závislá na tom, zda si uzel může být jist, že veřejný šifrovací klíč, který využívá, skutečně patří druhému uzlu a ne útočníkovi.

Obecný model digitálního podpisu ilustruje následující obrázek.

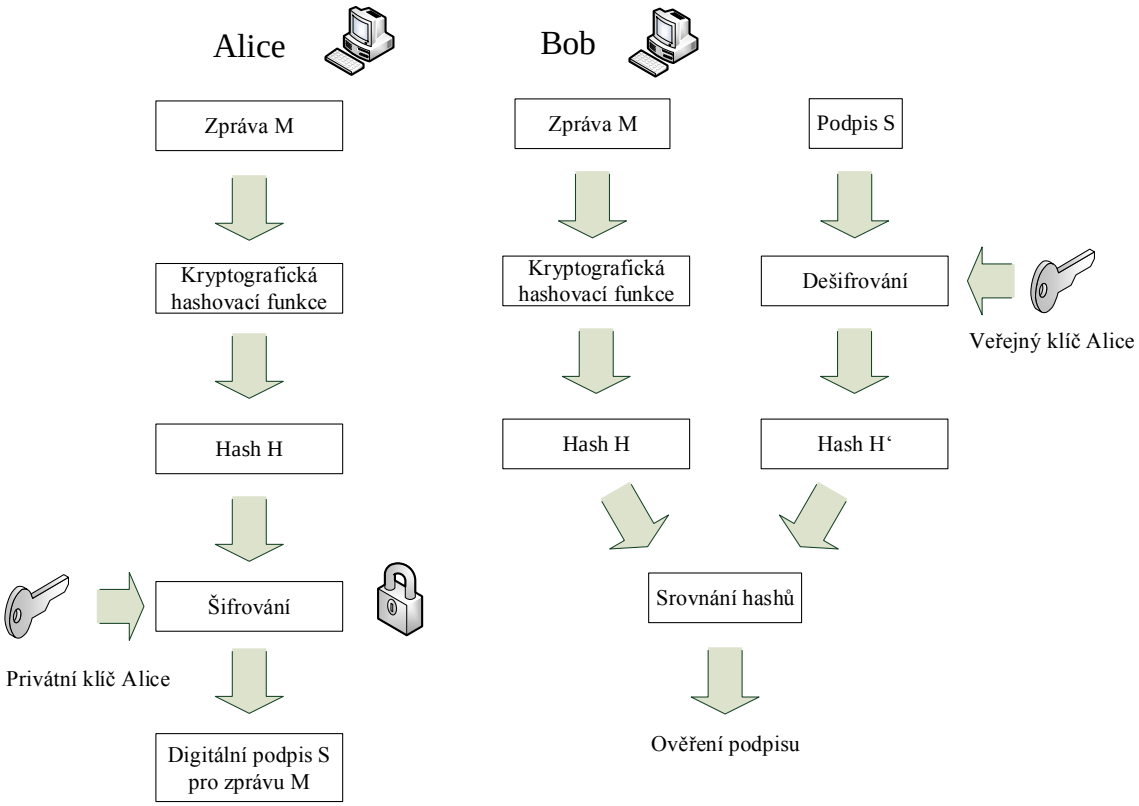


Obrázek 22 - Obecný koncept digitálního podpisu

Koncept digitálního podpisu zprávy lze popsat v několika krocích. V této situaci spolu chtějí komunikovat dva uzly – Alice a Bob.

1. Bob chce odeslat podepsanou zprávu Alici.
2. Z vytvořené zprávy vygeneruje pomocí hashovacího algoritmu a privátního klíče digitální podpis.

3. Podpis přiřadí ke zprávě a odešle Alici.
4. Alice ověří za pomocí Bobova veřejného klíče autenticitu zprávy a buď ji akceptuje, nebo odmítne.



Obrázek 23 - Zjednodušené zobrazení základních prvků digitálního podpisu

Podpis zprávy je tvořen pomocí hashe (otisku) zprávy. Bob tedy nejprve vypočítá hash *h* ze zprávy, kterou chce odeslat. Tento hash *h* poté zašifruje svým privátním klíčem a přiřadí jej ke zprávě jako digitální podpis S.

Alice potřebuje ověřit autenticitu zprávy. Za pomoci hashovacího algoritmu tedy vypočítá vlastní hash zprávy *h‘*. Poté pomocí Bobova veřejného klíče rozšifruje digitální podpis S, který byl přiřazen ke zprávě a získá hash *h.* Pokud se hodnoty *h* a *h‘* rovnají, je si Alice jistá, že zprávu odeslal Bob a nebyla cestou změněna.

Digitální podpisy mají kromě ověření integrity zprávy následující vlastnosti:

- ověřují autora a datum podpisu,
- ověřují obsah zprávy v době podpisu,
- musí být ověřitelné třetí stranou pro vyřešení sporů.

Autenticita podpisu závisí na tom, zda vyměněné veřejné klíče jsou skutečně veřejnými klíči jednotlivých uzlů a nejsou podvrženy útočníkem. K tomu je potřeba bezpečná výměna veřejných klíčů mezi uzly.

6.2.3 Výměna veřejných klíčů

Pro vzájemnou komunikaci je u asymetrické kryptografie vždy potřeba, aby si uzly vyměnily své veřejné klíče. Existuje několik možnosti, jak mohou být klíče vyměněny:

- veřejné oznamování,
- s pomocí veřejné autority,
- pomocí certifikátů s veřejným klíčem.

Při veřejném oznamování klíčů si nemůže být uzel nikdy jist, zda se za druhý uzel nevydává útočník a nepodvrhl mu tak svůj veřejný klíč.

Do procesu lze zapojit veřejnou autoritu, která spravuje klíče, a uzly jí důvěřují. Uzel A tedy vždy může požádat autoritu, aby mu poskytla veřejný klíč uzlu B. Protože autorita zašifruje tento klíč svým privátním klíčem, je si uzel jist, že zpráva pochází od dané autority. I tento přístup má svá negativa. Pro navázání komunikace mezi dvěma uzly je potřeba minimálně 7 zpráv, z čehož 4 slouží na získání klíčů od autority a zbylé tři pro iniciaci spojení. Dalším negativem je nutnost zapojení autority vždy, když chtějí uzly vzájemně komunikovat (pokud již své klíče nemají uloženy).

Poslední možností jsou digitální certifikáty s veřejným klíčem. Díky nim mohou být klíče vyměněny i bez aktivního zapojení certifikační autority. Uzel jednoduše pošle svůj certifikát druhému uzlu a protože je certifikát potvrzen autoritou, je považován za platný.

6.2.4 Digitální certifikáty

Digitální certifikáty jsou digitálně podepsané veřejné šifrovací klíče, které podepisuje certifikační autorita a jsou šířeny ve speciálním formátu.

Schéma digitálních certifikátů má několik základních požadavků:

- kdokoli může přečíst z certifikátu jméno a veřejný klíč vlastníka,
- kdokoli může ověřit, že certifikát byl vytvořen certifikační autoritou a nebyl porušen,
- pouze certifikační autorita může tvořit a měnit certifikáty.

Certifikační autorita je tedy v procesu zapojena (stejně jako v případě výměny klíčů přímo pomocí certifikační autority), ale pouze jako vydavatel certifikátů a vyměnit veřejné klíče lze bez jejího aktivního zapojení.

Ověření autenticity je dosaženo za pomoci tzv. přenosu důvěry. To znamená, že třetí strana, které uzel důvěřuje, ověří autenticitu druhého uzlu. Pokud uzel důvěřuje třetí straně, může věřit i ověřené identitě druhého uzlu.

Certifikační autorita prokazuje původ vydání certifikátu pomocí digitálního podpisu. Další část se věnuje již konkrétní implementaci certifikátů pomocí formátu X.509.

6.2.5 Standard X.509 a certifikáty

Z pohledu kryptografie je X.509 standard pro infrastrukturu, správu a distribuci veřejných klíčů (PKI - public key infrastructure) vytvořený organizací ITU-T v roce 1988. Tento standard specifikuje formát a parametry certifikátů, seznamy neplatných certifikátů (CRL - certificate revocation list), autentizační protokoly, algoritmus pro validaci certifikátů a další.

X.509 je z pohledu internetové komunikace velice důležitý standard, neboť struktura certifikátů a autentizační protokoly jsou využity pro různé aplikace, například pro formát S/MIME (standard pro zabezpečení elektronické pošty), protokol SSL/TLS a také pro IPsec.

Standard je založen na využití asymetrické kryptografie a digitálních podpisů. Nepředepisuje použití algoritmů pro podepisování ani pro hashování. Pro podpisy nicméně doporučuje RSA. Aktuálně je standard popsán v RFC 5280, které bylo vydáno v květnu roku 2008. X.509 je součástí standardu X.500, který mimo jiné definuje adresářové služby. Adresářem je v tomto případně myšlena databáze údajů o uživatelích a zdrojích systému.

6.2.6 Certifikačních autority a jejich struktura

Certifikační autorita je důvěryhodný subjekt (organizace), který se stará o vydávání certifikátů. Každá CA má přidělenou množinu uzlů, pro které spravuje certifikáty. Při velkém počtu uživatelů je totiž nepraktické, aby se o všechny klíče starala pouze jedna CA.

Pokud chce uživatel ověřit certifikát, který vydala CA pod kterou uzel spadá, je situace snadná. Uzel vezme veřejný klíč CA a ověří podpis certifikátu. Pokud ale veřejný klíč CA nezná, je situace složitější.

Uzel A vlastní certifikát vydaný CA označenou jako X_1. Uzel B vlastní certifikát vydaný autoritou X_2. Uzel A v této situaci nezná veřejný klíč X_2 a nemůže tak ověřit podpis certifikátu. Pokud si však certifikační autority vzájemně (a bezpečně) vymění své veřejné klíče, může dojít k tzv. hierarchickému ověřování, což umožní uzlu A ověřit certifikát uzlu B.

Aby mohl uzel A ověřit certifikát B, musí učinit následující kroky:

1. A obdrží z adresářové služby certifikát X_2 podepsaný autoritou X_1. Protože A zná veřejný klíč X_1, může ze získaného certifikátu bezpečně (tj. může ověřit podpis) získat klíč X_2.
2. Nyní se A vrátí do adresáře a získá certifikát uzlu B podepsaný X_2. Protože A má nyní důvěryhodnou kopii veřejného klíče X_2, může bez problémů ověřit podpis certifikátu náležícího B.

S použitím notace X.509 lze tento řetězec zapsat následujícím způsobem:

$$\mathbf{X_1\langle\langle X_2\rangle\rangle\ X_2\langle\langle B\rangle\rangle}$$

Stejným způsobem může uzel B bezpečně získat veřejný klíč A:

$$\mathbf{X_2 <<X_1>> X_1<<A>>}$$

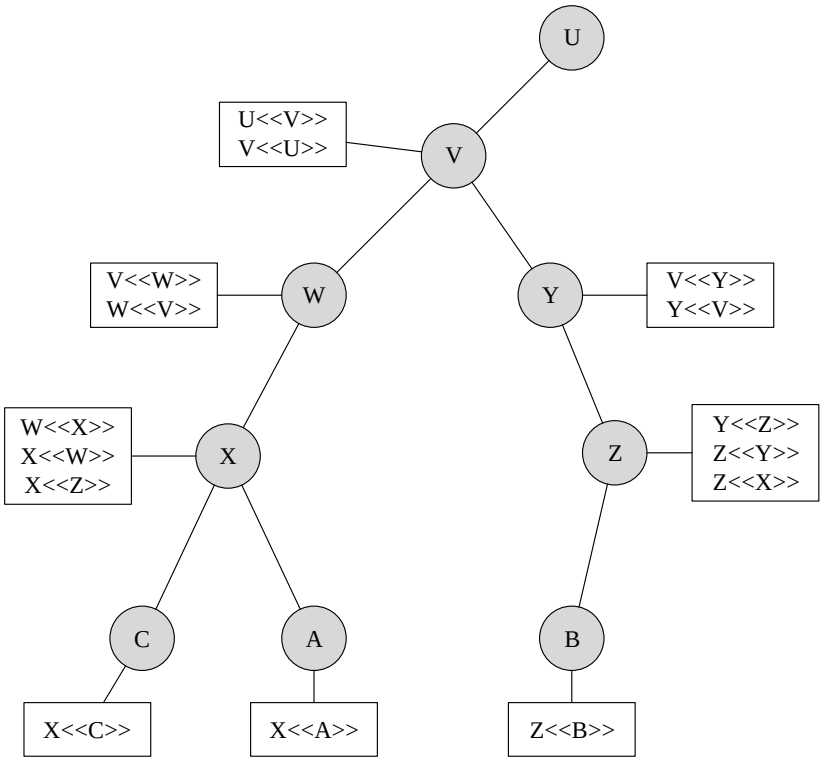
Schéma řetězce důvěry není limitováno dvěma certifikáty a může pokračovat i při N certifikátech.

$$\mathbf{X_1<<X_2>> X_2<<X_3>> ... X_N<<B>>}$$

V každém řetězci důvěry pak musí platit, že CA X_i a X_{i+1} musí mít vzájemně ověřené certifikáty. Tedy musí existovat certifikát pro X_i podepsaný X_{i+1} a zároveň musí existovat certifikát X_{i+1} podepsaný X_i.

Všechny tyto certifikáty se musí nacházet v adresářové službě a uživatel (uzel) musí znát způsob jejich spojení. V případě X.509 je struktura CA hierarchická, nicméně mohou existovat i jiné struktury.

Následující obrázek znázorňuje ukázku hierarchické struktury certifikačních autorit.



Obrázek 24 - Hypotetický příklad hierarchické struktury certifikačních autorit

Jednotlivá kolečka v obrázku znázorňují certifikační autority a jejich spojení definuje hierarchickou strukturu. Přidružené značky značí certifikáty pro každou CA, které se musí nacházet v adresáři.

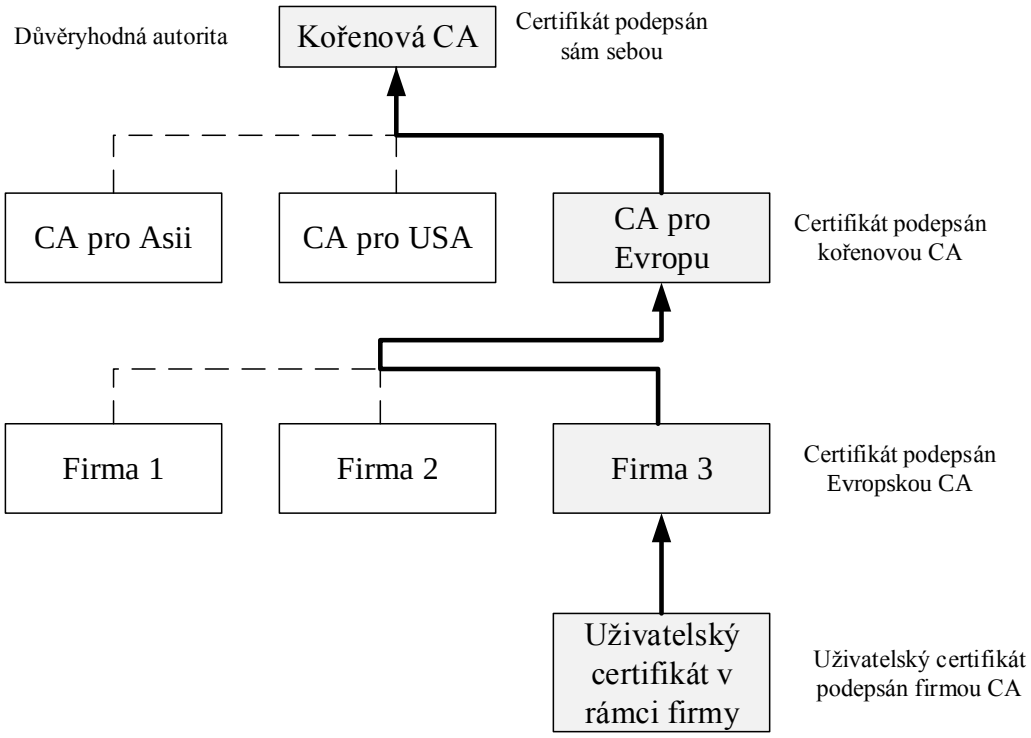
V adresáři se nachází dva typy certifikátů:

- **Dopředné (forward) certifikáty:** Certifikáty autority X vytvořeny ostatními autoritami.
- **Reverzní certifikáty:** Certifikáty vytvořeny autoritou X určené pro ostatní CA.

Tvorba hierarchické cesty je poté stejná jako ve výše popsaném případě dvou certifikátů. Cesta pro autoritu A k ověření B (a získání jejího veřejného klíče) tedy může vypadat takto:

X<<W>> W<<V>> V<<Y>> Y<<Z>> Z<<B>>

Jak může vypadat struktura skládající se z reálných certifikačních autorit, ilustruje následující obrázek.



Obrázek 25 - Možná struktura certifikačních autorit

Na vrcholu samotné hierarchie se nachází kořenový (root) certifikát, který podepisuje autorita, kterou certifikát identifikuje. Jde o tzv. self-signed certifikát (podepsán sám sebou). Kořenový certifikát lze běžně vygenerovat např. pro účely firmy.

Častěji se však využívají kořenové certifikáty vystavené nějakou globální certifikační autoritou, jejíž kořenové certifikáty jsou zpravidla dodávány v operačním systému či programu (např. webovém prohlížeči).

Mezi globální CA patří například:

- CAcert.org,
- DigiCert,
- GlobalSign,
- Verisign,
- a další.

6.2.7 Odvolávání (revokace) certifikátů

Každý certifikát obsahuje kromě jiných informací také dobu platnosti, podobně třeba jako kreditní karta. Nový certifikát je pak zpravidla vystaven ještě před skončením platnosti toho starého. Může tak být žádoucí platný certifikát odvolat, jinými slovy zneplatnit.

Může existovat řada dalších důvodů, proč certifikát odvolat:

- privátní klíč může být kompromitován,
- entita již není nadále certifikována danou CA, například pro porušení bezpečnostních politik autority,
- certifikát dané autority mohl být kompromitován.

Odvolané certifikáty jsou uchovávány v tzv. CRL (Certificate Revocation List) seznamech a spravuje je certifikační autorita. CRL seznamy by měly být dostupné pomocí adresářové služby.

Každý CRL seznam je podepsán svým vydavatelem a obsahuje jméno vydavatele, datum vytvoření seznamu, datum plánovaného vydání nového seznamu a záznam pro každý revokovaný certifikát. Každý záznam se poté skládá ze sériového čísla certifikátu a data jeho revokace. Aby seznamy CRL nepřesáhly únosné meze, jsou z nich postupně revokované certifikáty také odebírány.

Před tím než uzel použije certifikát, musí si nejprve ověřit, zda nebyl odvolán a to kontrolou CRL seznamu dané CA. Odkaz na seznam se nachází přímo v certifikátu. Problémem může být aktuálnost CRL seznamu, která není vždy zaručena.

Druhou možností pro ověřování certifikátů je využití OCSP (On-line Certificate Status Protocol) pomocí kterého lze interaktivně ověřit platnost certifikátu. Problémem je, jak uvádí Peterka[12], že ne všechny certifikační autority tento protokol podporují. Druhým negativem je, že pro výměnu certifikátů u dvou uzlů je potřeba zapojovat třetí stranu.

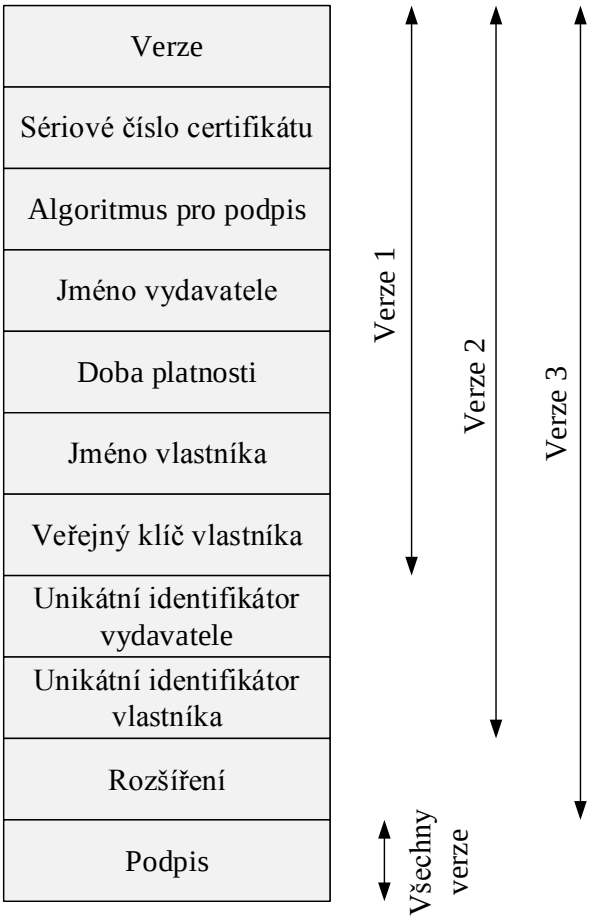
6.2.8 Formát certifikátů X.509

Srdcem samotného standardu X.509 je formát certifikát s veřejným klíčem vytvořený a podepsaný certifikační autoritou. Kdokoli, kdo vlastní veřejný klíč certifikační autority se může ujistit, že certifikát náleží danému subjektu.

[12] PETERKA, Jiří. Co je revokace certifikátu?. In: *EArchiv.cz: archiv článků a přednášek Jiřího Peterky* [online]. 2012 [cit. 2013-05-01]. Dostupné z: http://www.earchiv.cz/b12/b0504001.php3

Formát X.509 certifikátu existuje ve třech verzích (v1,v2,v3) a pokud je v této práci zmíněn certifikát X.509, je myšlena jeho nejnovější verze v3 z roku 1999.

Strukturu certifikátu znázorňuje následující obrázek.



Obrázek 26 - Struktura certifikátu X.509

Jednotlivá pole přitom nesou tyto informace:

- **Verze (version):** určuje verzi použitého X.509 formátu. Defaultně je nastavena na 1. Verze 2 nebo 3 je nastavena v případě přidání dalších polí (viz obrázek).
- **Sériové číslo certifikátu (serial number):** Unikátní celočíselná hodnota spojená s certifikátem a přiřazená certifikační autoritou.
- **Identifikátor algoritmu pro podpis (signature algorithm identifier):** Určuje podepisovací algoritmus a jeho parametry. Tato informace je zopakována také v poli u samotného podpisu a toto pole tak má minimální využití.
- **Jméno vydavatele (issuer name):** Jméno autority (X.500), která vydala a podepsala certifikát.
- **Doba platnosti (period of validity):** Skládá se ze dvou časových údajů – první udává od kdy je certifikát platný, druhý do kdy.

- **Jméno vlastníka (subject name):** Jméno vlastníka certifikátu, pro kterého byl vystaven. Potvrzuje, že subjekt s tímto jménem je vlastníkem daného páru privátního a soukromého klíče.
- **Veřejný klíč vlastníka (subject's public-key information):** Pole obsahuje veřejný klíč daného subjektu a informace o algoritmu, kterým byl vytvořen, včetně parametrů.
- **Unikátní identifikátor vydavatele (issuer unique identifier):** Volitelné pole sloužící k jednoznačné identifikaci vydavatele.
- **Unikátní identifikátor vlastníka (subject unique identifier):** Volitelné pole pro jednoznačnou identifikaci vlastníka.
- **Rozšíření (Extensions):** Jedno nebo více rozšiřujících polí. Toto pole bylo přidáno ve verzi 3 a podrobně je popsáno v další části.
- **Podpis (signature):** Podpis všech ostatních polí certifikátu. Pole obsahuje hash kód vytvořený ze všech ostatních polí podepsaný privátním klíčem certifikační autority. Obsahuje také identifikátor algoritmu, kterým byl hash vytvořen.

Třetí verze X.509 přidává k formátu certifikátu důležité pole – rozšíření. Druhá verze totiž nebyla schopna splnit další požadavky, jako je identifikace různých klíčů stejného vlastníka, využívaných v různých časech.

Místo přidávání dalších speciálních polí tak bylo přidáno jedno univerzální pole, které nabízí vyšší flexibilitu. Verze 3 tak nabízí mnoho volitelných rozšíření, přičemž každé rozšíření se skládá z identifikátoru, critical identifikátoru a hodnoty rozšíření. Critical identifikátor určuje, zda je možné rozšíření bezpečně ignorovat. Pokud je hodnota nastavena na TRUE a nelze rozšíření rozpoznat, musí být certifikát označen jako neplatný.

Rozšíření spadají do třech hlavních kategorií:

- informace o klíči a politikách,
- atributy certifikovaného subjektu a podepisovatele,
- omezení certifikační cesty.

První kategorie rozšíření, **informace o klíči a politikách**, uchovává dodatečné informace o certifikovaném subjektu a vystaviteli spolu s indikátorem politiky certifikátů. Certifikační politika je pojmenovaná sada pravidel, která určuje použitelnost certifikátu v dané komunitě uživatelů či ve třídě aplikací se společnými bezpečnostními požadavky.

Tato kategorie rozšíření zahrnuje:

- **Identifikátor klíče autority** (Authority key identifier): Identifikuje veřejný klíč autority, která podepsala certifikát. Používá se v případě, že CA využívá k podepisování více veřejných klíčů.
- **Identifikátor klíče subjektu** (subject key identifier): Identifikuje veřejný klíč, který je certifikován. Subjekt může mít stejně jako CA více klíčů.

- **Použití klíče** (key usage): Určuje a omezuje k čemu může být daný klíč využit (například šifrování či digitální podepisování).
- **Časový úsek pro použití privátního klíče** (private-key usage period): Označuje časový úsek pro použití odpovídajícího privátního klíče k veřejnému klíči. Privátní klíč je například u digitálních podpisů zpravidla používán pro podepisování kratší dobu, než veřejný klíč pro ověření podpisu.
- **Certifikační politiky** (certificate policies): Certifikáty mohou být používány v prostředích, kde je aplikováno více politik. Toto rozšíření obsahuje seznam politik, které daný certifikát podporuje.
- **Mapování politik** (policy mappings): Toto pole se využívá pouze tehdy, když certifikát pro jednu CA vydala druhá CA. Pole se využívá pro mapování politik (neboli k definici, které politiky mohou být rovnocenné) mezi jednotlivými doménami daných CA.

Do druhé kategorie rozšíření, nazvané **atributy certifikovaného subjektu a podepisovatele**, spadají alternativní jména v alternativních formátech pro certifikovaný subjekt či pro podepisovatele. Tyto pole také mohou nést dodatečné informace o certifikovaném subjektu, které mohou zvýšit důvěru uživatele v certifikát. Dodatečnými informacemi mohou být například poštovní adresa, pozice v korporaci (pro osobu) či obrázek.

Tato kategorie zahrnuje následující rozšíření:

- **Alternativní jméno subjektu** (subject alternative name): Obsahuje jedno čí více alternativních jmen subjektu v různých formách. Toto pole je důležité v různých aplikací jako je email, EDI či IPsec, které mohou vyžadovat jména ve specifickém formátu.
- **Alternativní jméno podepisovatele** (issuer alternative name): Obsahuje jedno čí více alternativních jmen podepisovatele certifikátu.
- **Atributy subjektu pro adresář** (subject directory attributes): Zahrnuje požadované X.500 atributy složky a jejich hodnoty pro subjekt certifikátu.

Poslední kategorie rozšíření, **omezení certifikační cesty**, umožňuje nastavit omezení validační cesty certifikátu, dokáže omezit typy certifikátů, které mohou být vydány CA nebo které se mohou objevit ve validačním řetězci.

Toto rozšíření zahrnuje následující pole:

- **Základní omezení** (basic constraints): Určuje, zda subjekt může fungovat také jako CA. Může omezit délku validační cesty pro certifikát.
- **Jmenné omezení** (name constraints): Využívá pouze certifikát dané CA, určuje jmenný prostor, který omezuje jména subjektů u vydaných certifikátů.
- **Politické omezení** (policy constraints): Možno využít pouze u certifikátu určeném pro CA. Omezuje validační cestu pro certifikát dvěma způsoby – může zakázat

mapování vybraných politik či explicitně vyžadovat konkrétní politiku v rámci certifikační cesty.

Hardwarová řešení

Hardwarovým řešením je myšleno řešení pro bezpečné uložení soukromého klíče a certifikátu. Zpravidla se jedná o speciální USB tokeny, čipové (smart) karty či HSM (hardware security modul – hardwarový bezpečnostní modul) moduly, na kterých je certifikát bezpečně úložen. Obecně se hardwarové zařízení pro uložení certifikátu nazývá bezpečnostní token (nebo také autentizační token, hardwarový token či kryptografický token).

Bezpečné uložení znamená, že k soukromému klíči se může dostat pouze jeho uživatel, který má příslušný software pro komunikaci s tokenem a zároveň zná přístupový pin. Soukromý klíč přitom nemůže být z tokenu zkopírován a není možné jej získat ani z počítače, na kterém je používán.

Čipové karty mají zpravidla kapacitu pro uložení více privátních klíčů pro více certifikátů. Zpravidla je na nich také uložen kořenový certifikát certifikační autority a kartu je tak možné používat i v neznámém (a potažmo nebezpečném) prostředí. Na kartě také bývá volný prostor pro umístění libovolných uživatelských dat. Citlivé operace se vždy provádí po zadání pinu. Pro čtení dat z karty je zpravidla potřeba speciální čtečka karet.

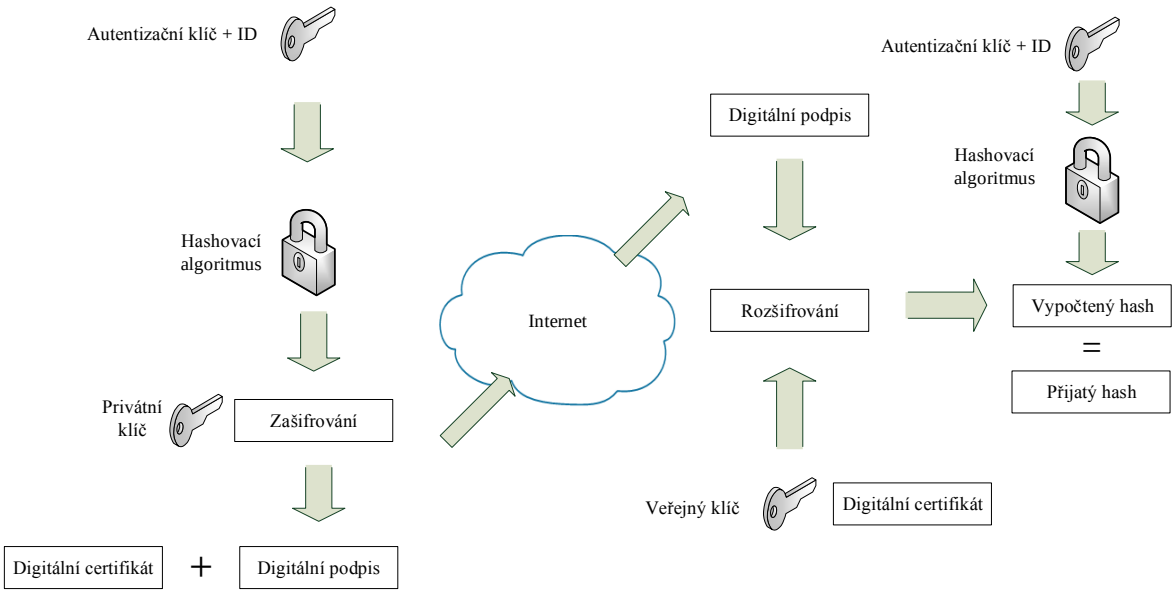
USB tokeny (dongly či klíčenky) jsou pohodlnější variantou čipových karet, kterou lze připojit ke každému počítači s USB rozhraním. Mezi jednotlivými tokeny jsou velké funkční rozdíly a také se liší možnostmi zabezpečení.

Jednoduché tokeny mohou obsahovat pouze paměťový prostor, ty chytřejší mohou obsahovat také procesor chránící přístup k datům, vlastní OS a nesložitější tokeny mohou obsahovat i speciální kryptografický procesor.

Protože klíč v případě hardwarových tokenů nelze dostat z tokenu ven a nijak s ním pracovat, je přístup šifrování či podepisování dat odlišný. Aplikace tak musí nejprve data odeslat do tokenu a token poté provede samotnou kryptografickou operaci. Teprve poté token předá zašifrovaná či podepsaná data zpět aplikaci.

6.2.9 Digitální certifikáty a IPsec

IPsec využívá certifikáty pro autentizaci jednotlivých uzlů v rámci IKE relace. Jejich obecné použití pro ověření identity v případě IPsec ilustruje následující obrázek. Certifikáty, respektive veřejné a soukromé klíče se nevyužívají pro šifrování přenášených dat, k tomu jsou určeny jiné, symetrické kryptografické algoritmy.



Obrázek 27 - Ověřování identity pomocí certifikátů v případě IPsecu

Ověření identit uzlů poté probíhá pomocí protokolu IKE takto:

1. Uzel vezme autentizační klíč (získaný při výměně klíčů pomocí algoritmu DH v předchozích fázích IKE) a své specifické informace (například ID) a vypočítá z nich hash.
2. Tento hash je nyní zašifrován soukromým klíčem uzlu, čímž vznikne digitální podpis.
3. Podpis je spolu s certifikátem přenesen v rámci autentizační relace IKE k vzdálenému uzlu.
4. Vzdálený uzel ověří digitální podpis tak, že jej rozšifruje pomocí privátního klíče prvního uzlu.
5. Nyní vytvoří vzdálený uzel nový hash za pomoci autentizačního klíče a ID prvního uzlu.
6. Pokud se přijatý a opětovně vytvořený hash shodují, uzly jsou autentizovány.

Pro elektronické podpisy se používají například algoritmy RSA či DSA. Informace o algoritmu je dodaná přímo v X.509 certifikátu. V případě použití RSA a protokolů AH či ESP je dle RFC 4359 nutné použít pro vytvoření autentizačního hashe algoritmus SHA-1.

6.3 Autentizace s využitím protokolu EAP

Díky protokolu EAP je možné využít v případě autentizace pro IPsec infrastrukturu AAA, neboli Autentizace, Autorizace a Accounting (účtování). Autentizace znamená ověření identity uživatele, pravidla přes uživatelské jméno a heslo. Autorizace je myšleno přidělení síťových prostředků a služeb a účtováním je myšleno sledování využití služeb jednotlivými uživateli.

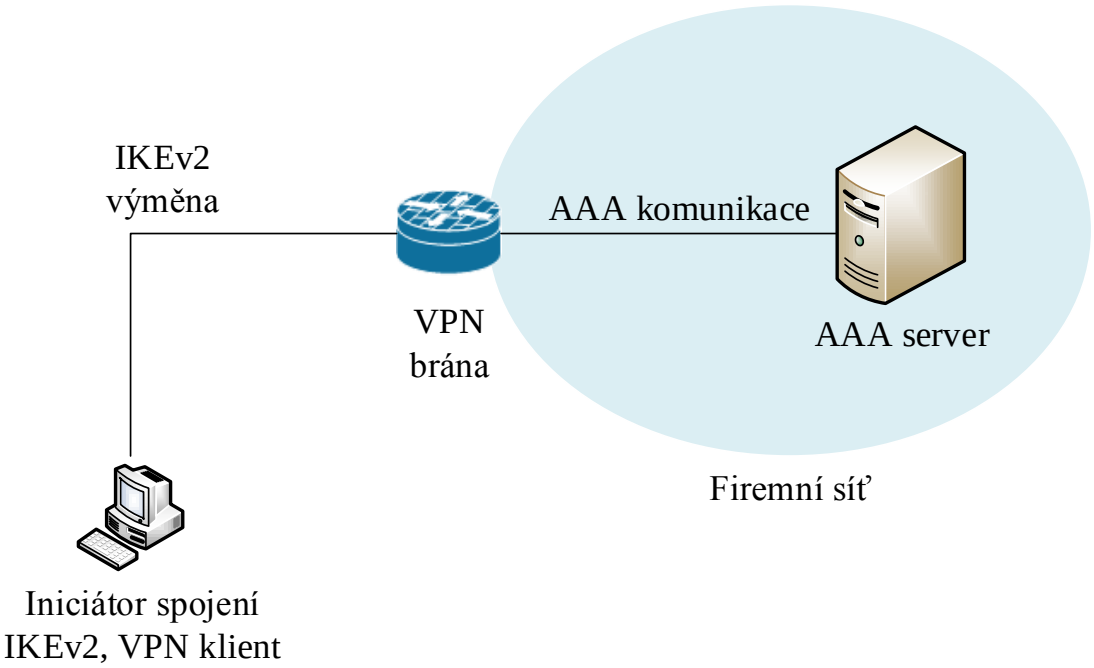
EAP je autentizační framework, nikoli konkrétní autentizační mechanismus. Zprostředkovává přenos a použití klíčů a parametrů generovaných EAP autentizačními algoritmy (označovanými také jako metody). EAP je definován v RFC 5247 a jeho použití s protokolem IKEv2 v RFC 5998.

EAP podporuje celou řadu autentizačních metod, přičemž konkrétní metody jsou definovány v samostatných RFC či jsou proprietární implementací výrobců. K nejznámějším metodám patři:

- EAP-MD5 (RFC 3748),
- EAP-TLS (RFC 5216),
- EAP-PSK (4764),
- EAP-TTLS (RFC 5281),
- EAP-MS-CHAPv2 (proprietární protokol Microsoftu),
- EAP-FAST (RFC 451),
- a další.

Organizace IANA, která se stará o přidělování čísel pro EAP metody, jich k 15.4.2013 registruje více jak čtyřicet[13].

Schéma pro použití tohoto typu autentizace v korporátní síti v rámci IKEv2 relace ilustruje následující obrázek.



Obrázek 28 - Schéma použití AAA infrastruktury a EAP protokolu v korporátní síti

[13] Internet Key Exchange Version 2 (IKEv2) Parameters. In: *IANA.org* [online]. 2005, 18.4.2013 [cit. 2013-05-01]. Dostupné z: http://www.iana.org/assignments/ikev2-parameters/ikev2-parameters.xml

Pro použití v rámci IKEv2 relace lze použít několik metod. V rámci RFC 5998 jsou stanoveny tři požadavky, které musí tyto metody splňovat:

- musí poskytovat vzájemnou autentizaci uzlů,
- musí umět generovat klíče,
- musí být odolná proti slovníkovým útokům.

Metody splňující tyto požadavky jsou považovány za bezpečné. RFC 5998 jmenuje následující metody, nicméně uvádí, že seznam není kompletní a existují i další bezpečné metody pro protokol EAP.

Mezi bezpečné EAP metody pro IKEv2 tedy patří:

- EAP-SIM,
- EAP-AKA,
- EAP-AKA' (vylepšená verze),
- EAP-GPSK,
- EAP-pwd,
- EAP-EKE,
- EPA-PAX,
- EAP-SAKE,
- EAP-SRP,
- EAP-POTP,
- EAP-TLS,
- EAP-FAST,
- EAP-TTLS.

Záleží tedy na výrobci, kterou metodu pro EAP se rozhodne implementovat do svých zařízení.

7 Algoritmus Diffie-Hellman pro generování a výměnu klíčů

Šifrovací algoritmy jako je DES, 3DES a AES, stejně jako hashovací algoritmy HMAC-MD5 či HMAC-SHA-1, pracují se symetrickým tajným klíčem, který pro šifrování a následné dešifrování musí znát oba komunikující uzly.

Manuální konfigurace je v takovém případě velmi neefektivní. Nejrychlejším způsobem je výměna tajných klíčů mezi jednotlivými uzly po veřejné infrastruktuře.

K tomuto účelu se v případě VPN využívá algoritmus Diffie-Hellman a to ve dvou verzích - More Modular Exponential Difiie-Hellman (MODP DH) a Elliptic curve Diffie–Hellman. (ECDH). Z toho vyplívá, že také existují dva typy skupin algoritmu DH – MODP a EC.

Cisco ve svých zařízeních podle dostupných informací[14] implementuje skupiny 1, 2, 5, 14, 19 a 20. Délky parametru P pro jednotlivé skupiny a jejich typ definuje následující tabulka.

Tabulka 7 - Délka parametru P pro jednotlivé DH skupiny

Skupina	Typ skupiny	Délka parametru P
1	MODP	768 bitů
2	MODP	1024 bitů
5	MODP	1536 bitů
14	MODP	2048 bitů
19	EC	256 bitů
20	EC	384 bitů

7.1 MODP Diffie-Hellman

Nejčastěji je algoritmus DH používán ve variantě MODP (More Modular Exponential), která je založena na obtížnosti výpočtu diskrétních logaritmů.

7.1.1 Princip MODP Diffie-Hellman algoritmu

V systému existují dva veřejné parametry – **Prime (modulo) a Base (základ)** a komunikují spolu dva uzly – A a B, které můžeme označit jako Alice a Bob. Algoritmus lze rozdělit do pěti kroků.

První krok

V prvním kroku se Alice s Bobem shodnou na veřejných parametrech p a g. V tomto příkladě se shodli na číslech 17 a 5.

p = 17
g = 5

[14] NextGenerationEncryption. In: *Cisco System* [online]. 2012 [cit. 2013-05-01]. Dostupné z: http://www.cisco.com/web/about/security/intelligence/nextgen_crypto.html

Druhý krok

Alice si vygeneruje náhodné privátní číslo **a**, provede níže uvedený výpočet a výsledek odešle Bobovi. V našem případě si Alice vybrala číslo 4 a vypočtený výsledek je číslo 13. Operace **mod** neboli modulo představuje zbytek po celočíselném dělení.

A = g^a mod p
A = 5^4 mod 17
A = 13

Třetí krok

Bob vygeneruje svoje privátní číslo **b**, provede stejnou operaci jako Alice a výsledek pošle Alici. V našem případě volil Bob číslo 10 a jeho výsledek činil 9.

B = g^b mod p
B = 5^10 mod 17
B = 9

Čtvrtý krok

Alice provede výpočet tajného klíče pomocí čísla, které získala od Boba (9) a svého privátního čísla (4). Získá tak tajný klíč, kterým je číslo 16.

Tajný klíč = B^a mod p
Tajný klíč = 9^4
Tajný klíč = 6 561 mod 17
Tajný klíč = 16

Pátý krok

Stejný výpočet provede také Bob s číslem, které získal od Alice (13) a se svým tajným číslem (10).

Tajný klíč = A^b mod p
Tajný klíč = 13^10
Tajný klíč = 137 858 491 849 mod 17
Tajný klíč = 16

Jak je vidět, tak po pátém kroku zná Alice i Bob tajný klíč. Útočník jej ale nemůže získat. Zná sice veřejná čísla p, g, A a B, ale nezná ani jedno privátní číslo a tak nemůže klíč vypočítat.

Problémem je v případě algoritmu DH útok typu „man-in-the-midle“, který může Alici a Bobovi podvrhnout svá čísla.

7.2 EC Diffie Hellman algoritmus

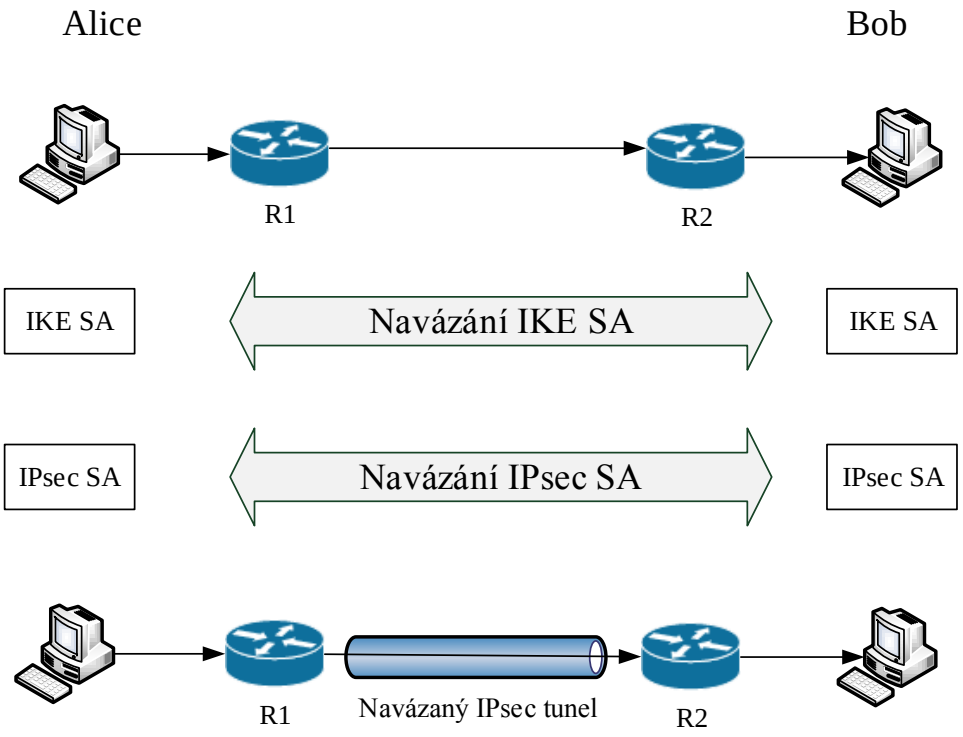
ECDH ve varianta algoritmu Diffie-Hellman, která využívá pro výměnu tajných informací kryptografii nad eliptickými křivkami. Algoritmus funguje podobně jako verze MODP DH, je však výpočetně složitější a proto se tato práce jeho detailním popisem nezabývá.

8 Navazování IPsec tunelu

Aby mohly dva uzly navázat vzájemné IPsec spojení, musí se vytvořit bezpečnostní asociace, které definují, jak je daný provoz chráněn. K tomu se využívá protokol IKE (Internet Key Excahnge).

V současné době existuje protokol IKE ve verzi v1 a také ve vylepšené verzi v2. Detaily navazování IPsec tunelu se liší podle použité verze protokolu, obecný koncept lze popsat v několika krocích.

1. Alice chce poslat data Bobovi po zabezpečeném kanále.
2. Směrovače R1 a R2 zahájí komunikaci na protokolu IKE.
3. R1 a R2 se dohodnou na bezpečnostních asociacích pro IKE.
4. Nyní, po již zabezpečeném kanále, si vymění informace potřebné pro navázání bezpečnostních asociací pro IPsec.
5. Po vyjednání bezpečnostních asociací je navázán IPsec tunel.
6. Alice nyní může poslat Bobovi data skrze IPsec tunel.



Obrázek 29 - Navazování IPsec tunelu

Nejdůležitější rozdíly mezi IKEv1 a IKEv2

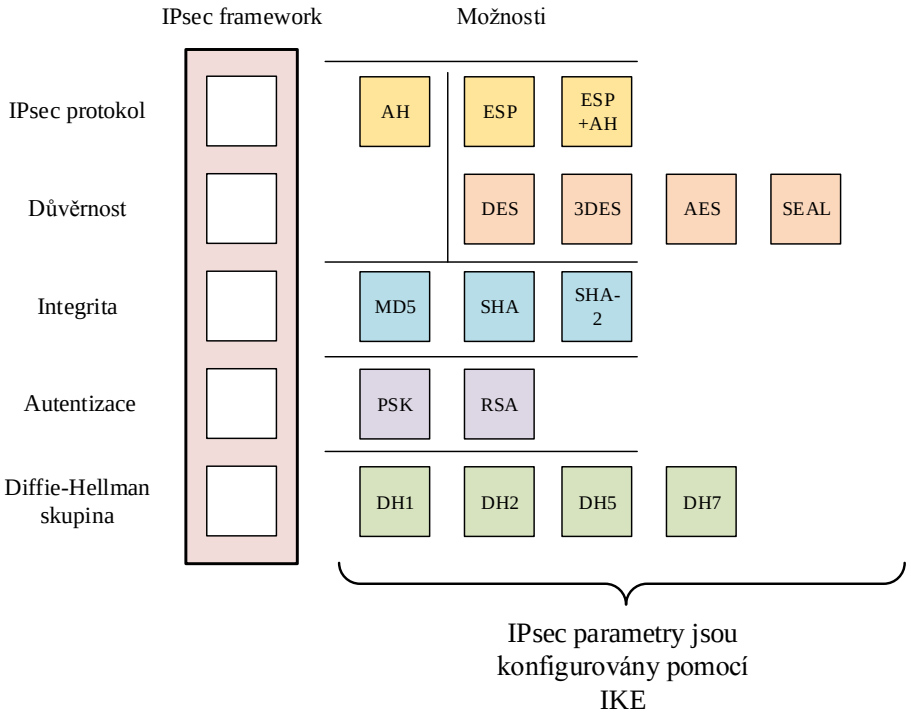
Protokoly IKEv1 a IKEv2 jsou vzájemně nekompatibilní a v mnoha ohledech velmi odlišené. Níže uvedená tabulka uvádí seznam nejdůležitějších rozdílů.

Tabulka 8 - Nejdůležitější rozdíly mezi protokoly IKEv1 a IKEv2

Parametr	IKEv1	IKEv2
Výměnné módy	Dva – main a aggressive	Pouze jedna procedura výměny, módy byly zrušeny
Počet zpráv pro vytvoření IPsec tunelu	Main mode: 9 Aggressive mód: 6	4 zprávy
Autentizační metody	Předsdílený klíč Digitální certifikáty (RSA) Public key encryption Public key encryption (revidovaná verze)	Předsdílený klíč Digitální certifikáty (RSA)
Možnosti autentizace	Oba uzly musí využívat stejný typ autentizace	Asymetrická autentizace – každý uzel může být autentizován jinak
Metody pro obnovu klíče	Nedefinovány	Definovány
NAT Traversal	Definován jako rozšíření	Podporován v základu
Spolehlivost	Méně spolehlivý než IKEv2	Všechny zprávy v páru požadavek – odpověď Definována procedura pro mazání SA Procedura pro opětovný přenos zpráv
Ochrana proti útokům DoS	Nepodporováno	Anti-replay ochrana Definovány tzv. cookies jako ochrana proti zahlcení Opraveny další zranitelnosti z IKEv1

8.1 Navázání bezpečnostních asociací pomocí IKEv1 a IKEv2

Před tím než jsou vytvořeny bezpečnostní asociace (tento koncept je popsán v kapitole 2.3) mezi dvěma uzly, musí se zařízení nejprve domluvit na sdílených tajných klíčích, použitém typu šifrování a také musí vzájemně ověřit svou identitu. K vytvoření bezpečnostních asociací v případě IPsecu se využívá protokol IKE neboli Internet Key Exchange.



Obrázek 30 - Vztah IPsec a IKE

Vztah mezi IPsecem a IKE lze zjednodušeně popsat v 5. bodech:

1. IPsec vyžaduje bezpečnostní asociace, aby mohl chránit přenášená data.
2. Pokud nejsou žádné SA definovány, požádá protokol IKE o jejich vytvoření.
3. IKE naváže spojení s daným uzlem a domluví bezpečnostní asociace pro IPsec.
4. IPsec nyní může šifrovat a chránit přenášená data.
5. Po vypršení doby platnosti naváže IKE nové bezpečnostní asociace pro IPsec.

IKE pracuje na UDP portu 500 a jeho první verzi (IKEv1) popisuje RFC 2409 z roku 1998. Druhá verze, známá jako IKEv2 byla představena na konci roku 2005 a popisuje ji RFC 2005. Poslední update IKEv2 je z roku 2010 zveřejněn pod RFC 5996.

IKE je hybridním protokolem, který přebírá vlastnosti od třech jiných:

- ISAKMP,
- Oakley,
- SKEME.

ISAKMP poskytuje framework pro autentizaci a výměnu klíčů, ale samotný mechanismus výměny klíčů nedefinuje. Definuje především formát zprávy a může obsahovat libovolný algoritmus pro výměnu klíčů.

Oakley protokol, přesněji Oakley Key Determination Protocol, je protokol pro výměnu klíčů, který využívá Diffie-Hellman algoritmu. SKEME je protokol pro univerzální výměnu klíčů, která poskytuje anonymitu, autentizaci a rychlou změnu klíče.

Protože IKE z výše popsaných protokolů využívá pouze některé části, zabývá se nadále tato práce IKE jako jedním celkem, který je popsaným výše zmíněnými RFC.

Často také dochází k záměně ISAKMP a IKE, například i v konfiguraci Cisco zařízení. Pro konfiguraci IKE je totiž nutné využít příkazu ISAKMP.

Perfect forward secrecy

Při použití IKE protokolu se předpokládá PFS neboli Perfect Forward Secrecy. Jde o předpoklad, že použitý klíč nebyl použit pro odvození jiných klíčů, takže prolomení jednoho klíče neovlivní bezpečnost ostatních klíčů.

8.1.1 Protokol IKEv1

Pro navázání zabezpečené komunikace mezi dvěma uzly používá IKEv1 dvě fáze (phase 1 a phase 2). V první fázi navíc může protokol pracovat ve dvou módech – main a aggressive.

Protokol pracuje se dvěma typy bezpečnostních asociací – IKE (ISAKAMP) SA a IPsec SA. První typ slouží pro ochranu dat při výměně pomocí IKE (ochrana management spojení), druhý typ je následně odvozen a využíván protokolem IPsec pro ochranu přenášených dat v rámci VPN spojení.

Fáze 1 v případě main módu

První fáze slouží k navázání IKE SA pro ochranu management spojení a skládá se ze třech výměn zpráv.

V první výměně dochází k dohodě na základních bezpečnostních pravidlech pro zabezpečení. Místo dohadování po jednotlivých algoritmech se dohadují kompletní skupiny algoritmů, takzvané IKE politiky (IKE policy set).

Iniciátor odešle všechny své IKE politiky. Respondent je přijme a srovná se svými politikami. Mohou nastat dvě situace:

- žádné politiky se neshodují, zabezpečené připojení nemůže být navázáno,
- jedna z politik iniciátora je totožná s politikou příjemce a fáze jedna může pokračovat.

Po první výměně ve fázi jedna se uzly shodly na bezpečnostních politikách a mohou si vyměnit bezpečnostní klíče.

Politika v případě Cisco zařízení může vypadat následovně:

- **3DES** důvěrnost (šifrování) dat,
- **SHA** pro integritu,
- **předsdílený klíč** pro autentizaci,
- **DH5** je skupina 5 definovaná pro algoritmus Diffie-Hellman,
- **životnost** je délka platnosti politiky.

Druhá výměna slouží ke generování a výměně tajných klíčů pro politiky dohodnuté v první výměně a to pomocí algoritmu Diffie-Hellman. Ten je popsán v kapitole 7. V politice, na které se uzly shodly v první fázi, je definována také skupina pro algoritmus DH, která se použije právě k výměně klíčů. Následně je všechna komunikace šifrována pomocí algoritmů dohodnutých v první výměně za použití klíčů z druhé výměny.

V třetí výměně první fáze se uzly vzájemně autentizují po již zabezpečeném kanále (rozdíl vůči agresivnímu módu).

Pro autentizaci v rámci IKEv1 mají uzly na výběr čtyři možnosti:

- předsdílený klíč,
- digitální podpisy,
- public key encryption (šifrování veřejným klíčem),
- public key encryption v revidované verzi.

Fáze 1 v případě aggressive módu

V případě agresivního módu je první fáze IKE rychlejší a kratší. Main mode využívá 3 výměny, které se skládají ze 6 paketů. Aggressive mode si vystačí s komprimovaným procesem v rámci jedné výměny, která se skládá ze 3 paketů a je ho možné popsat ve 4 krocích.

1) V prvním paketu odešle iniciátor návrh IKE politiky a také rovnou veřejný klíč pro algoritmus DH.
2) V druhém paketu odpoví příjemce akceptováním vybrané politiky, autentizačními údaji a veřejným klíčem pro algoritmus DH. Rovnou také vypočítá sdílený klíč pro DH.
3) Ve třetím paketu potvrdí iniciátor přijaté informace, následně vypočítá sdílený tajný klíč pro DH a ověří identitu uzlu a odešle své autentizační údaje.
4) Po přijetí třetího paketu ověří respondent identitu iniciátora a první fáze končí.

Nevýhodou agresivního módu je možnost odposlechnutí identit jednotlivých uzlů, které se na rozdíl od main módu posílají nešifrovaně. Výhodou je pak vyšší rychlost při navazování spojení. Možnosti pro autentizaci jsou stejné jako v případě main módu.

Fáze 2

V druhé fázi využívá IKE tzv. quick mód pro vytvoření IPsec SA (bezpečnostních asociací) a to pouze v případě, že první fáze byla dokončena a uzly se shodly na politikách a vzájemně se autentizovaly.

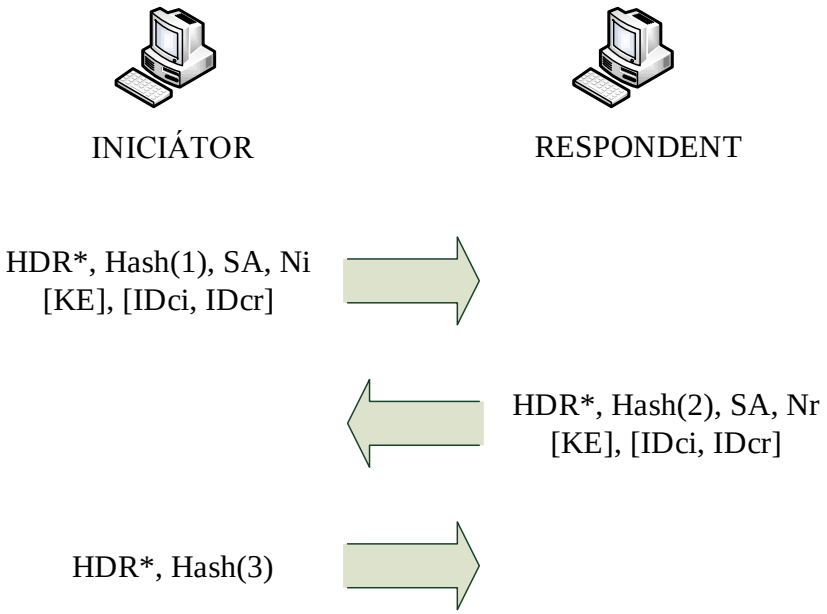
Fáze 2 zajišťuje 4 důležité funkce:

- vyjedná bezpečnostní parametry IPsecu (IPsec transform set),
- vytvoří bezpečnostní asociace pro IPsec,

- periodicky bezpečnostní asociace obnovuje (po vypršení doby životnosti),
- může také provádět další výměny klíčů pomocí DH.

Quick mód ve fázi 2 si vystačí se třemi pakety, celý postup zachycuje níže uvedený obrázek.

1. Iniciátor odešle návrh bezpečnostní asociace pro IPsec respondentovi (může být jedna i více).
2. Respondent schválí (či neschválí) navrhovanou politiku.
3. Iniciátor potvrdí příjem a fáze 2 může být ukončena.



Obrázek 31 – Průběh druhé fáze protokolu IKEv1

Kde:

HDR* - šifrovaná hlavička protokolu ISAKMP
SA – navrhovaná bezpečnostní asociace
Hash(1) – hash počítaný z ISAKMP hlavičky a dalších hlaviček
Hash(2) – stejný jako Hash(1), pouze neobsahuje iniciátorovu nonce
Hash(3) – skládá se z id zprávy (v ISAKMP hlavičce) a dvou nonce – iniciátorovy a respondentovi
Ni – inicátorova nonce
Nr – respondentova nonce

Zprávy také mohou obsahovat volitelné parametry:

- **KE** – Key Exchange payload, který obsahuje veřejné informace pro výměnu klíčů algoritmem Diffie-Hellman,
- **IDic** – ID inicáotora (může být např. IP adresa, rozsah IP adres, IP subnet),
- **IDir** – ID respondenta.

Poté co je fáze 2 ukončena tak, že iniciátor pošle poslední třetí paket, vznikly na každém uzlu dvě bezpečnostní asociace (SA) pro IPsec – jedna pro příchozí a druhá pro odchozí data.

8.1.2 Protokol IKEv2

IKE ve verzi 2 není zpětně kompatibilní s první verzí, protokoly jsou si však podobné. Činnost protokolu ale není na rozdíl od první verze rozdělena do fází, ale skládá se pouze z jednotlivých výměn.

Popis činnosti IKEv2

Jednotlivé výměny se nazývají následovně:

- IKE_SA_INIT,
- IKE_AUTH,
- CREATE_CHILD_SA,
- INFORMATIONAL.

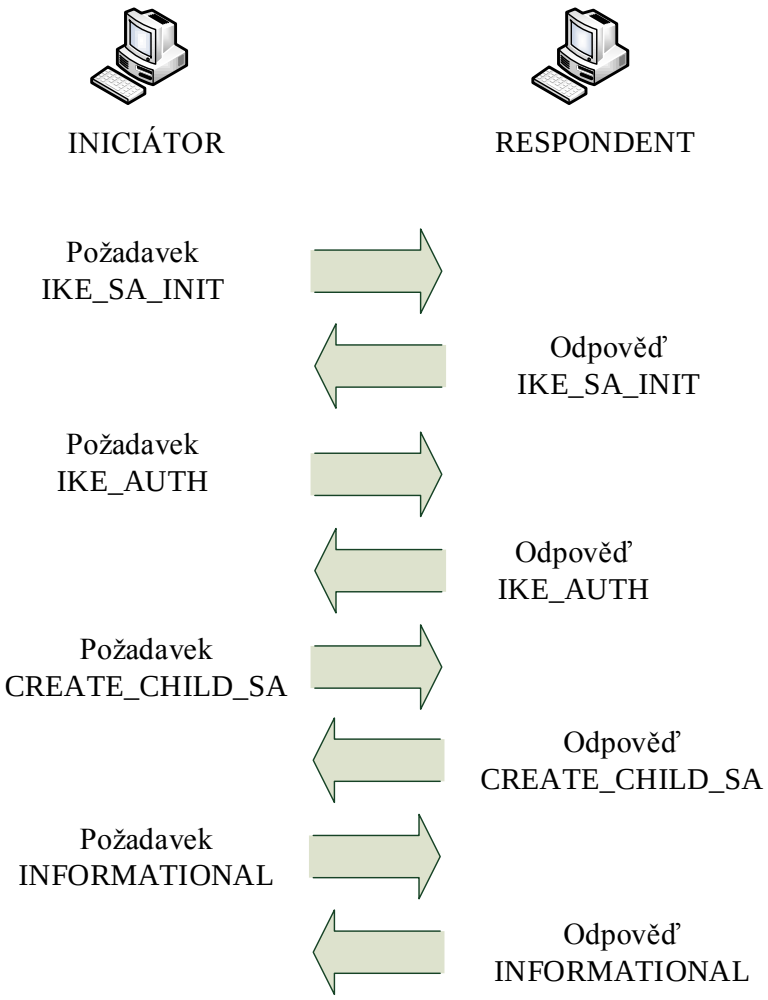
První výměnou je IKE_SA_INIT a slouží k vytvoření bezpečnostních asociací pro IKE, posílá nonce a také hodnoty pro algoritmus Diffie-Hellman.

IKE_AUTH se stará o vzájemné potvrzení identit uzlů, prokázání znalosti tajného klíče a vytvoření první bezpečnostní asociace pro přenos dat pomocí IPsec (nazývané také CHILD_SA).

Výměna definovaná jako CREATE_CHILD_SA vytváří další bezpečnostní asociace pro IPsec (CHILD_SA), případně obnovuje starší asociace pro vypršení doby platnosti. V některých ohledech se tato výměna podobá fázi dvě u protokolu IKEv1. Tuto výměnu mohou jako jedinou zahájit oba uzly a ne jen iniciátor.

Poslední výměna se nazývá INFORMATIONAL a je určena pro mazání bezpečnostních asociací, hlášení chyb, kontroly doby platnosti a další služby informačního charakteru.

Pokud je z nějakého důvodu (například vypršení doby životnosti) potřeba změnit šifrovací klíče (tzv. rekeying) nebo vytvořit nové bezpečnostní asociace, využije se fáze CREATE_CHILD_SA.



Obrázek 32 - Příklad IKEv2 relace

Možnosti autentizace

Protokol IKEv2 umožňuje v rámci výměny IKE_AUTH tři možnosti autentizace. Na rozdíl od IKEv1 již nepodporuje šifrování veřejným klíčem (pubic key encryption). Tento přístup nahradil moderní způsob autentizace pomocí protokolu EAP (Extensible Authentication Protocol).

V případě IKEv2 tedy mohou uzly využít následující metody autentizace:

- předsdílený klíč,
- digitální podpisy,
- EAP protokol.

Poslední zmíněná možnost má význam především u VPN pro vzdálený přístup, neboť ve spojení s Radius serverem dokáže autentizovat jednotlivé uživatele. Všechny výše zmíněné metody autentizace jsou rozebrány v kapitole 6 věnující se autentizaci uživatelů u IPsecu.

9 Návrh a konfigurace testovacího scénáře

Tato kapitola je zaměřena na návrh a konfiguraci virtuální privátní sítě postavené na zařízeních společnosti Cisco a frameworku IPsec. Zmiňuje všechny hlavní příkazy pro konfiguraci IPsec tunelu pro vzdálený přístup.

9.1 Bezpečnostní doporučení společnosti Cisco

Cisco na svém webu cisco.com pravidelně zveřejňuje doporučení týkající se kryptografických algoritmů. Poslední aktualizace tohoto dokumentu proběhla v dubnu 2012. Dokument je dostupný online na stránkách cisco.com[9].

Cisco dělí algoritmy do čtyř skupin:

- nevhodné (avoid),
- zastaralé (legacy),
- akceptovatelné (acceptable),
- algoritmy následující generace (next generation encryption – NEG).

Nevhodné jsou takové algoritmy, které neposkytují adekvátní stupeň ochrany oproti moderním hrozbám a útokům a neměly by tak být používány pro ochranu citlivých dat.

Zastaralé algoritmy nutně neznamenají špatnou úroveň ochrany, podle Cisca poskytují akceptovatelnou úroveň ochrany, nicméně by měly být využity pouze v případě, že neexistují lepší alternativy (například díky použití starších zařízení). Označení zastaralé (legacy) získaly tyto algoritmy z toho důvodu, že již byly nahrazeny lepšími a modernějšími alternativami.

Akceptovatelné algoritmy poskytují adekvátní úroveň ochrany a mohou být použity pro ochranu citlivých dat.

Algoritmy z kategorie **NGE** (následující generace) jsou nejlepší algoritmy, které by měly z pohledu bezpečnostních parametrů dostačovat na následující dvě dekády.

Tabulka 9 - Bezpečnostní doporučení společnosti Cisco[15]

Algoritmus	Operace	Doporučení	Alternativa
DES	Šifrování	Nevhodný	AES
3DES	Šifrování	Zastaralý	AES
RC4	Šifrování	Zastaralý	AES
AES-CBC mód	Šifrování	Akceptovatelný	AES-GCM
AES-GMC mód	Šifrování	NGE	--
DH 768-1024	Výměna klíčů	Nevhodný	DH-2048
RSA 768-1024	Šifrování		RSA-2048
DSA 768-1024	Autentizace		DSA-2048
DH 2048	Výměna klíčů	Akceptovatelný	ECDH-256
RSA 2048	Šifrování		--
DSA 2048	Autentizace		ECDSA-256
MD5	Zajištění integrity	Nevhodný	SHA-256
SHA-1	Zajištění integrity	Zastaralý	SHA-256
SHA-256	Zajištění integrity	NGE	SHA-384
SHA-384	Zajištění integrity		--
SHA-512	Zajištění integrity		--
HMAC-MD5	Zajištění integrity	Zastaralý	HMAC-SHA-1
HMAC-SHA-1	Zajištění integrity	Akceptovatelný	--
ECDH-256	Výměna klíčů	Akceptovatelný	ECDH-384
ECDSA-256	Autentizace	Akceptovatelný	ECDSA-384
ECDH-384	Výměna klíčů	NGE	--
ECDSA-384	Autentizace	NEG	--

Společnost Cisco také ve stejném dokumentu stanovila doporučení na minimální kryptografické algoritmy (naposledy aktualizováno v polovině roku 2012).

Tabulka 10 - Minimální kryptografické algoritmy podle doporučení společnosti Cisco

Operace	Minimální doporučení
Šifrování	AES-128 v CBC módu
Šifrování s veřejným klíčem (PKE)	RSA-2048
Autentizace	RSA-2048, DSA-2048
Integrita	SHA-256
Výměna klíčů	DH 2048 (Group 14)

[15] Next Generation Encryption. In: *Cisco System* [online]. 2012 [cit. 2013-05-01]. Dostupné z: http://www.cisco.com/web/about/security/intelligence/nextgen_crypto.html

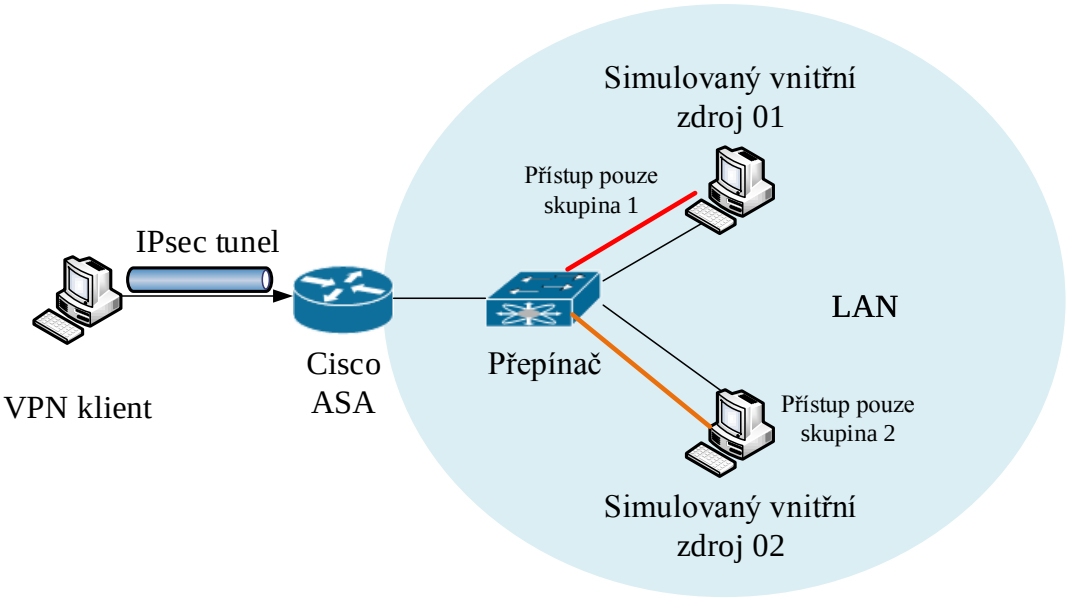
9.2 Návrh testovací konfigurace

Tato část se zabývá praktickým návrhem vzdáleného přístupu na koncové prvky s využitím frameworku IPsec. Jedním z požadavků na konfiguraci je využití dvoufaktorové autentizace. Jde o metodu autentizace, při které se ověřují dva faktory. Zpravidla se ověřuje něco, co uživatel vlastní (např. certifikát, generátor jednorázových hesel OTP), něco co uživatel zná (PIN nebo heslo), nebo nějaký biometrický znak (otisk prstů či sítnice).

Generátor jednorázových hesel bohužel nebyl při psaní práce k dispozici. Proto byla pro autentizaci zvolena kombinace certifikátu a uživatelského jména a hesla. Tento scénář se bohužel nepodařilo nakonfigurovat, důvody jsou uvedeny v kapitole 9.4.3. Autentizace tedy byla konfigurována pomocí předsdíleného klíče, uživatelského jména a hesla.

9.2.1 Schéma testovací konfigurace

Navrhovaná konfigurace počítá s dvěma počítači a jedním bezpečnostním zařízením Adaptive Security Appliance (ASA). Schéma navrhované konfigurace znázorňuje následující obrázek.



Obrázek 33 - Schéma testovací konfigurace

9.2.2 Popis konfigurace a použitá zařízení

Konfigurace odpovídá řešení Easy VPN pro vzdálený přístup pomocí IPsec tunelu od společnosti Cisco. Celá konfigurace je postavena na dvou hlavních prvcích. Jako VPN brána pracuje zařízení Cisco ASA 5505. Na klientské straně je nainstalován Cisco VPN klient ve verzi 5.0.0.7 pro Windows.

Z pohledu základních stavebních kamenů pro IPsec je konfigurováno:

- ESP protokol v tunelovacím módu,

- SHA algoritmus pro výpočet hashe,
- AES algoritmus pro zajištění důvěrnosti,
- předsdílený klíč pro autentizaci,
- skupina 5 pro algoritmus DH.

Scénář s použitím klienta Cisco VPN klienta spolu s Cisco ASA 5505 bohužel nepodporuje vyšší DH skupinu než 5 a tak nebylo možné split minimální kryptografické požadavky doporučované společností Cisco.

Tunel je tvořen mezi VPN klientem a zařízením ASA, kde je prováděna hlavní konfigurace. Simulované zdroje představují vnitřní servery na síti, schované za překladem adres. Pomocí ACL je omezen přístup na jednotlivé vnitřní zdroje pro dané skupiny uživatelů. První skupina může přistupovat pouze na vnitřní zdroj 1, druhá skupina může přistupovat pouze na vnitřní zdroj 2.

Konfiguraci lze rozdělit do několika kroků:

- konfigurace IP adres na ASA,
- konfigurace routingu,
- konfigurace IKE politik,
- konfigurace IPsec politik,
- nastavení pro klienty,
- nastavení přístupových politik,
- instalace softwaru a nastavení klientského zařízení.

9.2.3 Použitá adresace

Následující dvě tabulky shrnují IP adresy použité v návrhu testovací konfigurace. Pro simulovanou LAN je využit adresní rozsah 192.168.1.0/24, pro simulovaný internet je vyčleněn rozsah 172.16.1.0/24.

Tabulka 11 - Adresní rozsahy pro testovací konfiguraci

Adresní rozsah	Použití
192.168.1.0/24	Simulovaná LAN
172.16.1.0/24	Simulovaný internet
192.168.10.0/24	VPN tunel

Tabulka 12 - IP adresy pro testovací konfiguraci

Zařízení	Rozhraní	IP adresa	Popis
Cisco ASA	vlan 1 (fa 0/0)	192.168.1.1	Vnitřní rozhraní
Cisco ASA	vlan 2 (fa 0/7)	172.16.1.1	Vnější rozhraní
VPN klient	fa 0/0	172.16.1.2	-
Simulovaný zdroj 01	fa 0/0	192.168.1.10	-
Simulovaný zdroj 02	fa 0/0	192.168.1.20	-

9.3 Realizace testovací konfigurace

Realizace konfigurace probíhá postupně podle kroků zmíněných v kapitole 9.2.2 a obsahuje všechny hlavní příkazy zadávané do příkazového řádku zařízení Cisco ASA 5505. Příkazy, které se přímo netýkají tvorby IPsec tunelu tento popis neobsahuje a jsou uvedeny pouze v kompletní konfiguraci, která se nachází na přiloženém CD, spolu s konfigurací klientského softwaru.

9.3.1 Konfigurace IP adres na zařízení ASA

Nejprve je potřeba nakonfigurovat IP adresy na obou rozhraních zařízení ASA. U modelu ASA 5505 nelze konfigurovat IP přímo na jednotlivá rozhraní a musí se konfigurovat pomocí VLAN.

ASA na rozhraních pracuje s bezpečnostními úrovněmi (security levels) s hodnotou od 0 (nejnižší) do 100 (nejvyšší). Pouze zařízení na rozhraní s vyšší bezpečnostní úrovní může navázat spojení se zařízením na nižší bezpečnostní úrovni, ne naopak. Pokud je rozhraním přiděleno jméno, ASA přidělí bezpečnostní úrovně automaticky, 0 pro rozhraní outside a 100 pro rozhraní inside. Z outside rozhraní tak nelze navázat spojení s nikým za inside rozhraním.

Konfigurační příkazy jsou následující:

```
ciscoasa(config)# interface Vlan1
ciscoasa(config-if)# nameif inside
ciscoasa(config-if)# ip address 192.168.1.1 255.255.255.0

ciscoasa(config)# interface Vlan2
ciscoasa(config-if)# nameif outside
ciscoasa(config-if)# ip address 172.16.1.1 255.255.255.0
```

Fyzická rozhraní je potřeba přiřadit k do jednotlivých VLAN.

```
ciscoasa(config)# int ethernet 0/0
ciscoasa(config-if)# switchport access vlan 1

ciscoasa(config)# int ethernet 0/7
ciscoasa(config-if)# switchport access vlan 2
```

9.3.2 Konfigurace routingu

Vzhledem k jednoduchosti topologie stačí pro účely testovací konfigurace jediný příkaz a to vytvoření defaultní „routy“ pro vnější rozhraní.

```
ciscoasa(config)# route outside 0.0.0.0 0.0.0.0 172.168.1.1 1
```

9.3.3 Konfigurace IKE politik

Dalším krokem je konfigurace politik pro protokol IKE, který slouží po vyjednávání bezpečnostních asociací. Politik může existovat více, přičemž číslo politiky značí její prioritu (čím nižší, tím vyšší priorita). Politiky specifikují typ autentizace, šifrování dat, hashovací algoritmus a další. Před vytvořením politiky je potřeba nejprve zapnut protokol IKE na vnějším rozhraní.

```
ciscoasa(config)# crypto isakmp enable outside

ciscoasa(config)# crypto isakmp policy 5
ciscoasa(config-isakmp-policy)# authentication pre-share
ciscoasa(config-isakmp-policy)# encryption aes
ciscoasa(config-isakmp-policy)# hash sha
ciscoasa(config-isakmp-policy)# group 5
ciscoasa(config-isakmp-policy)# lifetime 86400
```

9.3.4 Konfigurace IPsec politik

V této fázi je potřeba nakonfigurovat tzv. krypto mapy, které definují politiky vytvářených tunelů. Nejprve se konfigurují dynamické krypto mapy, které jsou poté aplikovány na statické. Statické krypto mapy se používají u site-to-site spojení, kde jsou předem známy všechny parametry tunelu. Dynamické se používají u vzdáleného přístupu a dovolují domluvení parametrů tunelu přímo s klientem.

Krypto mapy využívají transform sety, které definují parametry tunelu. Transform sety definují IPsec protokol, šifrovací algoritmus a algoritmus pro hmac. Transform setů může být v zařízení více.

Postup konfigurace je následující:

- vytvoření transform setu,
- vytvoření dynamické krypto mapy pro příchozí spojení s definovanými transform sety,
- vytvoření statické krypto mapy s referencí na dynamickou mapu,
- přiřazení krypto mapy na rozhraní.

Konfigurace v případě testovacího návrhu je následující.

```
ciscoasa(config)# crypto ipsec transform-set ESP-AES-128-SHA esp-aes esp-sha-
hmac
ciscoasa(config)# crypto dynamic-map EASY_DYN_CRYPTO_MAP 65535 set
transform-set ESP-AES-128-SHA
ciscoasa(config)#crypto map EASYMAP 65535 ipsec-isakmp dynamic
EASY_DYN_CRYPTO_MAP
ciscoasa(config)# crypto map EASYMAP interface outside
```

9.3.5 Konfigurace klientského nastavení

V této části se konfigurují IP adresy, DNS server a další údaje přidělované připojovaným klientům. Klient musí obdržet od ASA korektní IP adresu, jinak nemůže být tunel navázán. Nejprve musí být vytvořen rozsah adres (ip pool), který je poté přidělen ke skupině pro připojení. Buď lze využít defaultní tunelovací skupinu, nebo vlastní. V případě testovací konfigurace byla vytvořena skupina VPN-CONN.

```
ciscoasa(config)# ip local pool 192vpn 192.168.10.0-192.168.10.254 mask
255.255.255.0

ciscoasa(config)# tunnel-group VPN-CONN type remote-access
ciscoasa(config)# tunnel-group VPN-CONN general-attributes
ciscoasa(config-tunnel-general)# address-pool 192vpn
```

9.3.6 Nastavení základních přístupových politik a autentizace

Dále je potřeba vytvořit VPN politiky, které jsou poté aplikovány na skupiny tunelů (tunnel group). Následující příkazy vytváří VPN politiku s názvem VPN-POLICY. Politika uživatele nijak neomezuje a umožňuje mu přistoupit ke všem zdrojům v síti. Filtrování je provedeno až v další části.

```
ciscoasa(config)# group-policy VPN-POLICY internal
ciscoasa(config)# group-policy VPN-POLICY attributes
ciscoasa(config-group-policy)# vpn-tunnel-protocol ipsec
ciscoasa(config-group-policy)# dns-server value 192.168.1.1
ciscoasa(config-group-policy)# default-domain value vpn-ipsec
```

Vytvořená politika se musí aplikovat na tunnel-group, vytvořenou v předchozím kroku. Zde se také definuje předsdílený klíč pro autentizaci skupiny.

```
ciscoasa(config)# tunnel-group VPN-CONN
ciscoasa(config-tunnel-ipsec)# pre-shared-key cisco
ciscoasa(config-tunnel-general)# default-group-policy VPN-POLICY
```

Autentizace jednotlivých uživatelů jménem a heslem pomocí XAUTH (Extended Authentication – rozšířená autentizace) je na zařízení ASA 5505 defaultně zapnutá a není potřeba ji aktivovat. Defaultní konfigurace jiných zařízení může být odlišná a rozšířenou autentizaci lze aktivovat následujícími příkazy.

```
ciscoasa(config-tunnel-ipsec)# isakmp ikev1-user-authentication xauth
ciscoasa(config-tunnel-ipsec)# radius-sdi-xauth
```

Nyní stačí vytvořit uživatelský účet v lokální databázi a VPN spojení je připraveno k použití. Použití uživatelského účtu je možné omezit pouze pro vzdálený přístup.

```
ciscoasa(config)# username user01 password pass01
ciscoasa(config)# username user01 attributes
ciscoasa(config-username)# service-type remote-access
```

9.4 Nastavení rozšířených přístupových politik

Nakonfigurovaný tunel je v této fázi již funkční, umožňuje však vzdáleným klientům plný přístup ke zdrojům ve vnitřní síti. Takové chování není zpravidla žádoucí a je potřeba aplikovat rozšířené přístupové politiky. Omezení přístupu se prování pomocí přístupových (ACL) listů.

Možností pro omezení přístupu existuje několik:

- ACL na síťových rozhraních,
- ACL pro uživatele či skupinu uživatelů,
- split tunneling (tunelování pouze některého typu dat).

Nejvhodnějším přístupem je v tomto případě využít přístupových listů pro skupinu uživatelů.

V testovací konfiguraci byly vytvořeny dvě skupiny. První skupina může přistupovat pouze k prvnímu vnitřnímu zdroji, druhá skupina může přistupovat pouze k druhému vnitřnímu zdroji. Koncept je znázorněn na obrázku 33 uvedeném na začátku kapitoly.

Protože první tunelovací skupina pro VPN je již vytvořena, je potřeba vytvořit druhou skupinu pomocí následujících příkazů.

```
ciscoasa(config)#ip local pool 192vpn2 192.168.20.0-192.168.20.254 mask
255.255.255.0

ciscoasa(config)# tunnel-group VPN-CONN2 type remote-access
ciscoasa(config)# tunnel-group VPN-CONN2 general-attributes
ciscoasa(config-tunnel-general)# address-pool 192vpn2
ciscoasa(config-tunnel-ipsec)# pre-shared-key cisco
ciscoasa(config-tunnel-ipsec)# radius-sdi-xauth
```

Protože druhá skupina bude využívat odlišnou VPN politiku, je potřeba jí také vytvořit a připojit k tunelové skupině.

```
ciscoasa(config-group-policy)# group-policy VPN-POLICY2 internal
ciscoasa(config)# group-policy VPN-POLICY2 attributes
ciscoasa(config-group-policy)# dns-server value 192.168.1.1
ciscoasa(config-group-policy)# vpn-tunnel-protocol IPsec
ciscoasa(config-group-policy)# exit
ciscoasa(config)# tunnel-group VPN-CONN2 general-attributes
ciscoasa(config-tunnel-general)# default-group-policy VPN-POLICY2
```

Nyní lze definovat přístupové listy. ACL 103 dovoluje přístup pouze na zdroj s IP adresou 192.168.1.10. ACL 104 dovoluje přístup pouze ke zdroji s adresou 192.168.1.20. Vzhledem k defaultnímu pravidlu na konci každého přístupového listu, které zakazuje jakoukoli komunikaci (deny any any), je první pravidlo zbytečné a je uvedené pouze pro přehlednost.

```
ciscoasa(config)# access-list 103 extended deny ip any host 192.168.1.20
ciscoasa(config)# access-list 103 extended permit ip any host 192.168.1.10

ciscoasa(config)# access-list 104 extended deny ip any host 192.168.1.10
ciscoasa(config)# access-list 104 extended permit ip any host 192.168.1.20
```

Vytvořené přístupové listy pro filtrování je nyní potřeba přiřadit k VPN politikám. Protože jsou VPN politiky přiřazeny k tunelovacím skupinám, je po následujících příkazech omezení přístupu již aktivní.

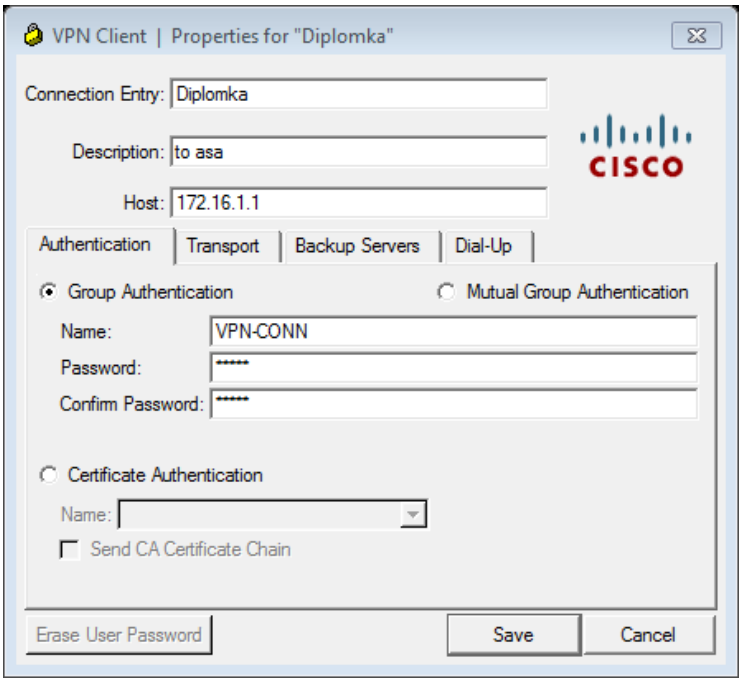
```
ciscoasa(config)# group-policy VPN-POLICY attributes
ciscoasa(config-group-policy)# vpn-filter value 103

ciscoasa(config)# group-policy VPN-POLICY2 attributes
ciscoasa(config-group-policy)# vpn-filter value 104
```

9.4.1 Nastavení softwarového klienta

Tato sekce shrnuje nastavení klientského softwaru Cisco Systems VPN Client ve verzi 5.0.07.0440.

Nové připojení lze přidat kliknutím na ikonu New. Nastavení v případě testovací konfigurace je vidět na následujícím obrázku. Ostatní parametry zůstaly na defaultním nastavení.



Obrázek 34 - Nastavení klientského softwaru

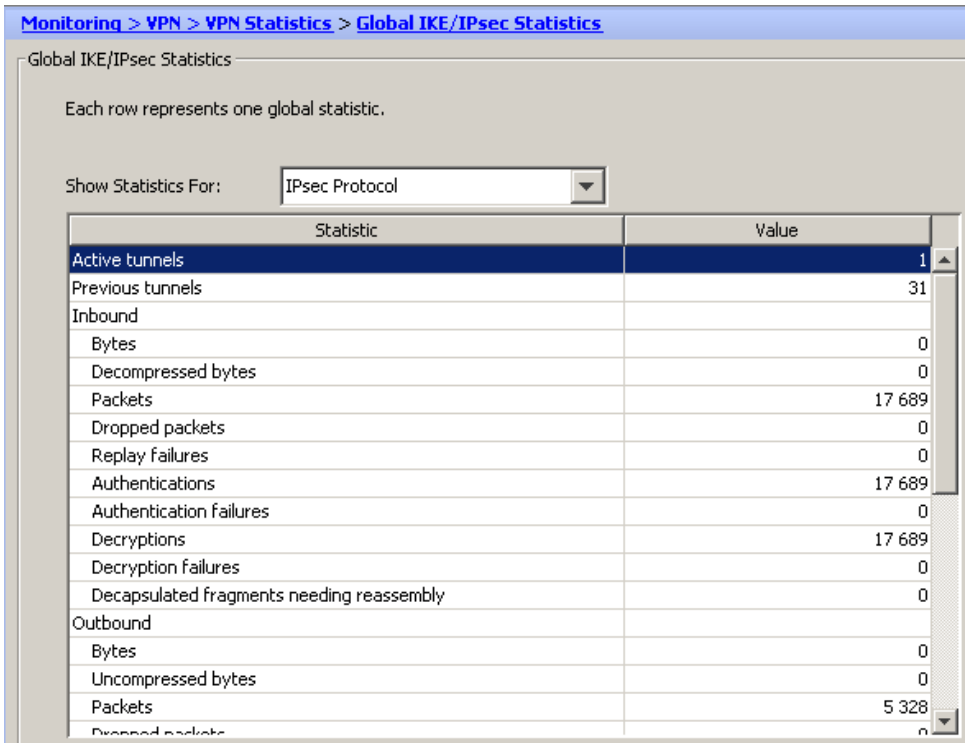
Byla zvolena skupinová autentizace pomocí předsdíleného klíče a názvu skupiny. V případě vyplnění názvu VPN-CONN má IPsec klient přístup k vnitřnímu zdroji 1, název skupiny VPN-CONN2 umožňuje přístup k vnitřnímu zdroji 2. Hesla u skupiny jsou v případě testovací konfigurace shodná, v praxi by se samozřejmě měla lišit.

9.4.2 Navázání tunelu a ověření funkčnosti

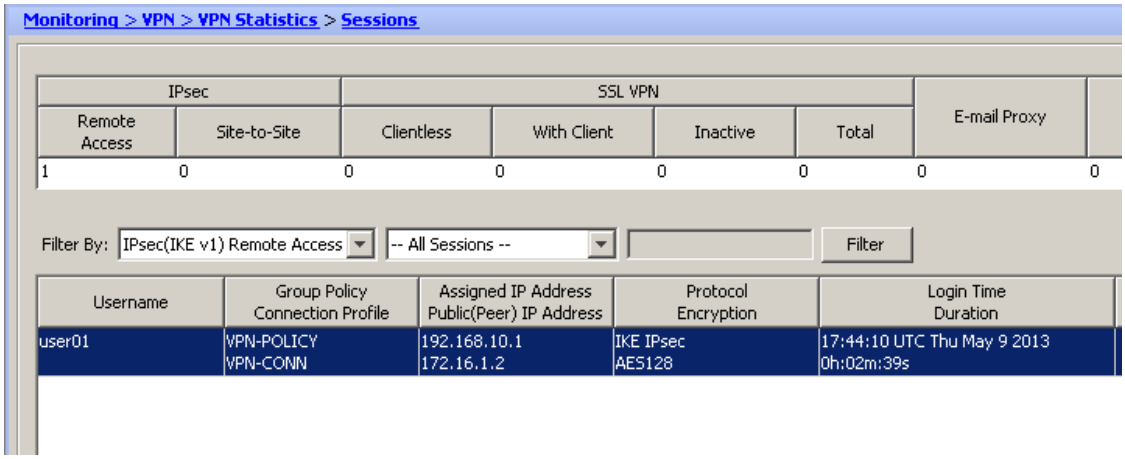
Tunel lze vytvořit prostým stisknutím tlačítka Connect ve VPN klientovi. Tunel je po vyjednání bezpečnostních politik a vytvoření bezpečnostních asociací navázán.

Funkčnost tunelu lze ověřit několika způsoby. Prvním znamením je možnost přístupu na vnitřní zdroje pomocí nástroje ping, stav tunelu zobrazuje také Cisco VPN klient. Je ale nutné ověřit, zda komunikace, např. nástrojem ping, skutečně probíhá skrze vytvořený tunel.

Nabízí se možnost využít populární nástroj WireShark pro analýzu paketů. Ten bohužel neumožňuje monitorování VPN rozhraní v systému Windows 7. Statistiky tunelu však lze snadno zobrazit v zařízení ASA pomocí grafického konfiguračního nástroje ASDM. Statistiky zobrazují například celkový počet přenesených bajtů skrze tunel. Při komunikaci lze sledovat, že statistiky přenesených bajtů se aktivně mění.



Obrázek 35 - Zobrazení globálních IKE/IPsec statistik



Obrázek 36 - Navázaná IPsec spojení, použité politiky a šifrování

9.4.3 Problémy při konfiguraci

V rámci testovacího návrhu byla konfigurována také autentizace uzlů pomocí certifikátů, bohužel neúspěšně. Klientský software totiž z neznámého důvodu odmítal ověřit podepsaná data od zařízení ASA.

Pro generování testovací hierarchie certifikátů byla vytvořena vlastní certifikační autorita pomocí nástroje OpenSSL. V použitém klientském softwaru lze všechny certifikáty ručně ověřit, což vždy proběhlo bez problémů. Navazování tunelu ale pokaždé skončilo ve fázi 1 u protokolu IKEv1, konkrétně u vzájemné autentizace uzlů (výměna autentizačního hashe).

Cisco VPN klient v logu hlásí, že nemůže ověřit autentizační hash a tudíž nemůže ověřit vzdálený uzel.

```
1295 13:57:26.897 05/09/13 Sev=Warning/3 IKE/0xE300007C
Failed to verify signature

1296 13:57:26.897 05/09/13 Sev=Warning/2 IKE/0xE300009B
Failed to authenticate peer (Navigator:915)
```

Zařízení ASA následně obdrží od klienta zprávu o nemožnosti ověření hashe a navazování VPN tunelu je následně ukončeno. Cisco ASA zobrazuje v debugovacím módu následující zprávu.

```
[IKEv1 DEBUG]: Group = VPN-CONN, IP = 172.16.1.2, processing hash payload
[IKEv1 DEBUG]: Group = VPN-CONN, IP = 172.16.1.2, processing notify payload
[IKEv1]: Group = VPN-CONN, IP = 172.16.1.2, Received authentication failure
message
[IKEv1]: Group = VPN-CONN, IP = 172.16.1.2, Received non-routine Notify
message: Authentication failed (24)
```

Tento problém přisuzuji buď chybně vygenerovaným certifikátům, nebo špatné kompatibilitě klientského softwaru se zařízením ASA 5505. Protože se problém přes veškerou snahu nepodařilo vyřešit, byla zvolena autentizace pomocí předsdíleného klíče.

Závěr

Cílem práce bylo shrnout teoretické poznatky ohledně vzdáleného přístupu na koncové prvky s využitím frameworku IPsec a také navrhnout testovací konfiguraci a popsat možnosti nastavení na zařízeních Cisco. Oba tyto cíle se podařilo splnit.

Teoretická část práce shrnuje ucelené informace o frameworku IPsec, rozdělené do kapitol podle základních stavebních kamenů této technologie. Obsahuje také popis průběhu navazování IPsec tunelu pomocí protokolu IKE.

Praktická část práce ukazuje jednu z možností, jak lze tuto technologii konfigurovat na zařízeních společnosti Cisco. Navrhovaná a otestovaná konfigurace zahrnuje krom vytvoření IPsec tunelu také možnosti pro omezení přístupu na koncové prvky pro jednotlivé skupiny uživatelů.

V této práci se mi podařilo vytvořit ucelenou teoretickou i praktickou příručku pro framework IPsec. Problematika IPsec je poměrně komplexní a zahrnuje celou řadu protokolů. Věřím tedy, že výsledky práce poslouží jak administrátorům, tak budoucím studentům, kteří se chtějí o této technologii dozvědět co možná nejvíce informací.

Literatura

[1] KENT, S. a K. SEO. RFC 4301: Security Architecture for the Internet Protocol. In: *IETF Tools* [online]. 2005 [cit. 2013-05-01]. Dostupné z: http://tools.ietf.org/html/rfc4301

[2] KENT, S. RFC 4302: IP Authentication Header. In: *IETF Tools* [online]. 2005 [cit. 2013-05-01]. Dostupné z: http://tools.ietf.org/html/rfc4302

[3] KENT, S. RFC 4303: IP Encapsulating Security Payload (ESP). In: *IETF Tools* [online]. 2005 [cit. 2013-05-01]. Dostupné z: http://tools.ietf.org/html/rfc4303

[4] PUŽMANOVÁ, Rita. *Moderní komunikační sítě od A do Z.* 2. akt. vydání. Brno: Computer Press, 2006. ISBN 80-251-1278-0.

[5] STALLING, William. *Cryptography and Network Security: Principles and Practice*. 5. vyd. New York: Pearson Education, 2010. ISBN 0-13-609704-9.

[6] DOSTÁLEK, Libor a Marta VOHNOUTOVÁ. V*elký průvodce infrastrukturou PKI a technologií elektronického podpis*u. Brno: Computer Press, 2006. ISBN 80-251-0828-7.

[7] HOOPER, Howard. *CCNP Security VPN 642-648: Official Cert Guide*. Indianapolis: Cisco Press, 2012. ISBN 1-58720-447-9.

[8] BRISCOE, B. RFC 6040: Tunnelling of Explicit Congestion Notification. In: *IETF Tools* [online]. 2010 [cit. 2013-05-01]. Dostupné z: http://tools.ietf.org/html/rfc6040

[9] *Podpora bezpečného tunelování v sítích IPv6 v operačních systémech*. Pardubice, 2009. Dostupné z: http://hdl.handle.net/10195/34780. Bakalářská práce. Univerzita Pardubice.

[10] *Hardwarové kryptografické moduly pro zabezpečení LAN*. Brno, 2008. Dostupné z: http://hdl.handle.net/11012/16634. Diplomová práce. Vysoké učení technické v Brně.

[11] PŘIBYL, Tomáš. VPN – pomocník (nejen) v bezpečnosti. In: S*ystemOnLine: S přehledem ve světě informačních technologií* [online]. 2007 [cit. 2013-05-01]. Dostupné z: http://www.systemonline.cz/it-security/vpn-pomocnik-nejen-v-bezpecnosti.htm

[12] SCHNEIER, Bruce. New Attack on AES. In: *Schneier on Security: A blog covering security and security technology* [online]. 2011 [cit. 2013-05-01]. Dostupné z: http://www.schneier.com/blog/archives/2011/08/new_attack_on_a_1.html

[13] KLÍMA, Vlastimil. Hašovací funkce MD5 a další prolomeny!. In: *Root.cz: Informace nejen ze světa Linuxu* [online]. 2004 [cit. 2013-05-01]. Dostupné z: http://www.root.cz/clanky/hasovaci-funkce-md5-a-dalsi-prolomeny/

[14] MANUEL, Stéphane. Classification and Generation of Disturbance Vectors for Collision Attacks against SHA-1. In: *Cryptology ePrint Archive* [online]. 2008 [cit. 2013-05-01]. Dostupné z: http://eprint.iacr.org/2008/469.pdf

[15] SUDJIMAN, David. Diffie-Hellman usage in IPSec. In: *David Sudjiman: Being different is hard, but not being different is harder* [online]. 2012 [cit. 2013-05-01]. Dostupné z: http://www.davidsudjiman.info/2012/08/22/diffie-hellman-usage-in-ipsec/

[16] MIČKA, Pavel. Diffie-Hellman. In: *Algoritmy.net: příručka vývojáře* [online]. 2012 [cit. 2013-05-01]. Dostupné z: http://www.algoritmy.net/article/84/Diffie-Hellman

[17] KUNDEROVÁ, Ludmila. Bezpečnost IS/IT: Kryptografické systémy. In: *Ústav informatiky, PEF MZLU v Brně* [online]. 2011 [cit. 2013-05-01]. Dostupné z: https://akela.mendelu.cz/~lidak/bis/8kryp.htm

[18] ORMAN, H. RFC 2412: The OAKLEY Key Determination Protocol. In: *IETF Tools* [online]. 1998 [cit. 2013-05-01]. Dostupné z: http://tools.ietf.org/html/rfc2412

[19] IKE, Internet Key Exchange: Network Communications. In: *Network Sorcery* [online]. 2006 [cit. 2013-05-01]. Dostupné z: http://www.networksorcery.com/enp/protocol/ike.htm

[20] MASON, Andrew. IPSec Overview Part Four: Internet Key Exchange (IKE). In: *Ciscopress.com* [online]. 2002 [cit. 2013-05-01]. Dostupné z: http://www.ciscopress.com/articles/article.asp?p=25474&seqNum=7

[21] HANÁČEK, Petr. Bezpečnostní funkce v počítačových sítích. In: *Zpravodaj ÚVT MU* [online]. 1999 [cit. 2013-05-01]. Dostupné z: http://www.ics.muni.cz/bulletin/articles/171.html

[22] LUHOVÝ, Karel. Seriál o VPN. In: Svět sítí [online]. 2003 [cit. 2013-05-01]. Dostupné z: http://www.svetsiti.cz/rubrika.asp?rid=17&tid=219

[23] VPN CONSORTIUM. Virtual Private Network Consortium [online]. 2008 [cit. 2013-05-01]. Dostupné z: http://www.vpnc.org/

[24] JAIN, Raj. IP Security. In: *Washington University in St. Louis* [online]. 2011 [cit. 2013-05-01]. Dostupné z: http://www.cse.wustl.edu/~jain/cse571-11/ftp/l_19ips/sld001.htm

[25] MANRAL, V. RFC 4835: Cryptographic Algorithm Implementation Requirements for Encapsulating Security Payload (ESP) and Authentication Header (AH). In: *IETF Tools* [online]. 2007 [cit. 2013-05-01]. Dostupné z: http://tools.ietf.org/html/rfc4835

[26] KRAWCZYK, H., M. BELLARE a R. CANETTI. RFC 2104: HMAC: Keyed-Hashing for Message Authentication. In: *IETF Tools* [online]. 1997 [cit. 2013-05-01]. Dostupné z: http://tools.ietf.org/html/rfc2104

[27] PATEL, B., B. ABOBA, W. DIXON, G. ZORN a S. BOOTH. RFC 3193: Securing L2TP using IPsec. In: *IETF Tools* [online]. 2001 [cit. 2013-05-01]. Dostupné z: http://tools.ietf.org/html/rfc3193

[28] MADSON, C. a R. GLENN. RFC 2404: The Use of HMAC-SHA-1-96 within ESP and AH. In: *IETF Tools* [online]. 1998 [cit. 2013-05-01]. Dostupné z: http://tools.ietf.org/html/rfc2404

[29] MADSON, C. a R. GLENN. RFC 2403: The Use of HMAC-MD5-96 within ESP and AH. In: *IETF Tools* [online]. 1998 [cit. 2013-05-01]. Dostupné z: http://tools.ietf.org/html/rfc2403

[30] FRANKEL, S., R. GLENN a S. KELLY. RFC 3602: The AES-CBC Cipher Algorithm and Its Use with IPsec. In: *IETF Tools* [online]. 2003 [cit. 2013-05-01]. Dostupné z: http://tools.ietf.org/html/rfc3602

[31] TURNER, S. a L. CHEN. RFC 6151: Updated Security Considerations for the MD5 Message-Digest and the HMAC-MD5 Algorithms. In: *IETF Tools* [online]. 2011 [cit. 2013-05-01]. Dostupné z: http://tools.ietf.org/html/rfc6151

[32] BEDELL, Crystal. The benefits and different types of SSL VPNs. In: *SearchEntepriseWAN* [online]. 2010 [cit. 2013-05-01]. Dostupné z: http://searchenterprisewan.techtarget.com/tutorial/The-benefits-and-different-types-of-SSL-VPNs

[33] SSL VPN. In: *Cisco Systems* [online]. 2012 [cit. 2013-05-01]. Dostupné z: http://www.cisco.com/en/US/docs/ios-xml/ios/sec_conn_sslvpn/configuration/15-2mt/sec-conn-sslvpn-ssl-vpn.html

[34] Next Generation Encryption. In: *Cisco System* [online]. 2012 [cit. 2013-05-01]. Dostupné z: http://www.cisco.com/web/about/security/intelligence/nextgen_crypto.html

[35] Internet Key Exchange Version 2 (IKEv2) Parameters. In: *IANA.org* [online]. 2005, 18.4.2013 [cit. 2013-05-01]. Dostupné z: http://www.iana.org/assignments/ikev2-parameters/ikev2-parameters.xml

[36] DOLEŽAL, Dušan. Co to je digitální certifikát. In: *Interval.cz* [online]. 2003 [cit. 2013-05-01]. Dostupné z: http://interval.cz/clanky/co-to-je-digitalni-certifikat/

[37] PETERKA, Jiří. Co je revokace certifikátu?. In: *EArchiv.cz: archiv článků a přednášek Jiřího Peterky* [online]. 2012 [cit. 2013-05-01]. Dostupné z: http://www.earchiv.cz/b12/b0504001.php3

[38] KOUŘIL, Daniel. Certifikáty veřejných klíčů. In: *Zpravodaj ÚVT MU* [online]. 2000 [cit. 2013-05-01]. Dostupné z: http://www.ics.muni.cz/bulletin/articles/181.html

[39] CARTS, David A. A Review of the Diffie-Hellman Algorithm and its Use in Secure Internet Protocols. In: *SANS Institute* [online]. 2001 [cit. 2013-05-01]. Dostupné z: http://www.sans.org/reading_room/whitepapers/vpns/review-diffie-hellman-algorithm-secure-internet-protocols_751