Slovenská technická univerzita v Bratislave
Fakulta informatiky a informačných technológií

Ing. Peter Kaňuch

Autoreferát dizertačnej práce

Optimalizácia protokolu HIP pre Internet vecí

na získanie akademického titulu philosophiae doctor - PhD

v doktorandskom študijnom programe aplikovaná informatika D-AI
v študijnom odbore aplikovaná informatika D-AI
Forma štúdia Denná
Miesto a dátum Bratislava 04.07.2022

Dizertačná práca bola vypracovaná na Ústave počítačového inžinierstva a aplikovanej informatiky Fakulty informatiky a informačných technológií v Bratislave v študijnom odbore aplikovaná informatika

Predkladateľ: **Ing. Peter Kaňuch**
Ústav počítačového inžinierstva a aplikovanej informatiky
Fakulta informatiky a informačných technológií
Ilkovičová 2, 842 16 Bratislava

Školiteľ: **doc. Ing. Ladislav Hudec, CSc.**
Ústav počítačového inžinierstva a aplikovanej informatiky
Fakulta informatiky a informačných technológií
Ilkovičová 2, 842 16 Bratislava

Školiteľ špecialista: **doc. Ing. Dominik Macko, PhD.**
Ústav počítačového inžinierstva a aplikovanej informatiky
Fakulta informatiky a informačných technológií
Ilkovičová 2, 842 16 Bratislava

Oponenti: **doc. Ing. Anton Baláž, PhD.**
Katedra počítačov a informatiky
Fakulta elektrotechniky a informatiky
Letná 9, 042 00 Košice

doc. Ing. Ondrej Ryšavý, PhD.
Ústav informačních systémů
Fakulta informačních technologií
Božetěchova 2, 612 00 Brno, ČR

Autoreferát bol rozoslaný .....

Obhajoba dizertačnej práce sa bude konať dňa ..... o ..... h na ústave aplikovanej informatiky Fakulty informatiky a informačných technológií v Bratislave (Ilkovičová 2, 842 16 Bratislava)

prof. Ing. Ivan Kotuliak, PhD.
Dekan FIIT STU Bratislava

Abstrakt

V súčastnosti je k Internetu pripojených čoraz viac vzájomne komunikujúcich zariadení označovaných ako Internet vecí. V práci sa venujeme klasifikácii zariadení pre IoT, komunikačných technológií a bezpečnostných protokolov so zreteľom na ich energetickú náročnosť. Ďalej sa v práci venujeme existujúcim optimalizáciám bezpečnostného protokolu HIP používaného v prostredí Internetu vecí, analyzujeme jednotlivé metódy a prístupy optimalizácie s dôrazom na energetickú náročnosť. Na základe vyhodnotenia súčasného stavu sme predstavili ciele práce, ktorých hlavnou myšlienkou je návrh energetického zefektívnenia zabezpečenia komunikácie kombináciou nových a existujúcich optimalizačných metód, pričom zameranie bude na koncové uzly Internetu vecí citlivé na energetickú spotrebu. Na základe rozboru jednotlivých prístupov optimalizácie a ich vhodnosti použitia v tejto práci sme navrhli ich zahrnutie do finálneho výsledku. Výsledný navrhnutý optimalizovaný protokol S-HIP pre prostredie Internetu vecí sme otestovali z rôznych hľadísk v testovacom prostredí a výsledky porovnali s existujúcimi riešeniami. Riešenie dokazuje jedny z najlepších dosiahnutých výsledkov optimalizovaných protokolov.

Obsah

1. Úvod

S postupným zavádzaním Internetu do každodenného života a automatizovaním jednotlivých procesov od priemyslu až po domácnosť, sa vytvoril priestor pre zavedenie pojmu Internet vecí (*IoT - z angl. Internet of Things*)[14]. Nie je to len prepojenie osobných počítačov, mobilov, či nositeľnej elektroniky, ale sú to inteligentné zariadenia, ktoré pomocou k nim pripojených senzorov a akčných členov pomáhajú ľuďom monitorovať úplne všetko, častokrát aj za hranice našich možností [8]. Internet vecí jednoducho prepája dva oddelené svety, a to skutočný svet s virtuálnym prostredím [19]. Tento pojem zahŕňa všetky jednoznačne identifikovateľné a adresovateľné inteligentné zariadenia, ktoré dokážu medzi sebou komunikovať [1, 12, 19]. Existujú rôzne využitia a aplikácie Internetu vecí, na ktoré sú kladené rôzne požiadavky. Jednou z najdôležitejších požiadaviek pre tieto zariadenia je ich mobilita a bezdrôtovosť. Preto sú tieto zariadenia napájané pomocou batérie s možnosťou prepnutia do stavu nečinnosti (*tzv. power down, idle režim*)[11].

Medzi ďalšie požiadavky na zariadenia IoT patrí [2] :

1. mobilita a bezdrôtovosť,
2. použiteľnosť s/vo vnorených systémoch,
3. rôznorodosť systémov pre jednotlivé možnosti využitia,
4. škálovateľnosť systémov z hľadiska veľkého nárastu,
5. energetická účinnosť,
6. iné.

Tu vstupuje do hry aj bezpečnosť, pretože každé takéto zariadenie, ale aj komuniká-

cia sú zraniteľné a mali by byť zabezpečené rôznymi bezpečnostnými technikami. Zabezpečenie nám však môže zvýšiť celkovú energetickú náročnosť riešenia, čo môže byť kritické. Preto sa mnohé výskumy v oblasti Internetu vecí venujú najmä rôznym optimalizáciám tak, aby čo najviac znížili celkovú energetickú náročnosť v rámci výpočtov a komunikácie.
Počet pripojených zariadení stúpa každým rokom čoraz viac, čím stúpa aj počet útokov, nie len na zariadenia Internetu vecí. V súčastnosti sa preto treba vo väčšej miere venovať oblasti bezpečnosti. Nasledujúce body, ako:

- aplikácia IoT zariadení v jednotlivých oblastiach života,
- maximalizácia výdrže batérie zariadenia,
- minimalizácia energetickej náročnosti bezpečnostného protokolu,
- zachovanie bezpečnostných vlastností,

môžme označiť ako hlavnú motiváciu pre dosiahnutie čo najlepšieho výsledku v tejto práci.

2. Bezpečnosť v IoT

V súčastnosti sa čoraz viac do pozornosti dostáva slovo bezpečnosť, ochrana osobných údajov a podobne. Napriek tomu, že práve bezpečnosť z pohľadu zisku nie je pre firmy zaujímavá, je veľmi dôležitá. To si však mnoho ľudí neuvedomuje, kým nepodľahnú určitému bezpečnostnému incidentu, čo môže koniec koncov stáť oveľa viac, ako práve samotné zabezpečenie.

V rámci odvetvia informatiky a informačných technológií existujú rôzne vetvy bezpečnosti, napríklad bezpečnosť webových aplikácií, bezpečnosť a zraniteľnosť databázových systémov, bezpečnosť operačných systémov, či v neposlednom rade bezpečnosť siete. Vzhľadom na to, že väčšina aplikácií komunikuje cez sieť, či internet, je bezpečnosť sietí jednou z najdôležitejších oblastí. O to viac je dôležitá pri kritických systémoch, ktoré môžu ohroziť aj samotnú bezpečnosť človeka. Do tejto skupiny môžme zaradiť aj zariadenia Internetu vecí. Na základe delenia IoT zariadení môžme určiť, že senzorové zariadenia patria k menej kritickým oproti zariadeniam spadajúcim do kategórie hybridov a reakčných, kde často dochádza ku konkrétnej činnosti daného zariadenia. Narušenie bezpečnosti využitím niektorej z bezpečnostných slabín takéhoto zariadenia môže v konečnom dôsledku spôsobiť obrovské škody ako napríklad odstávka distribúcie energie, napadnutie výrobných procesov v priemysle, spôsobenie environmentálnych škôd, či v neposlednom rade ohrozenie samotného človeka, či jeho zdravotného stavu a iné [13].

„Zariadenia pripojené k Internetu sú vystavené všetkým typom hrozieb, o to viac zariadenia Internetu vecí, ktoré sú častokrát fyzicky dostupné útočníkovi a na svoje zabezpečenie nemajú taký výpočtový, či energetický výkon, ktorý by zabezpečil maximálnu možnú ochranu [13].”

Ako sa vyvíja oblasť informatiky, postupne vznikali a vznikajú nové bezpečnostné protokoly pre použitie v klasických IP sieťach, ale aj ich odľahčené verzie pre použite v oblasti Internetu vecí, v ktorej stále existujú rôzne výzvy [16]:

- Interoperabilita, t.j. podpora protokolu rôznych pripojených zariadení a komunikačných protokolov.
- Obmedzenie zdrojov zariadenia, a to najmä obmedzenie procesora, pamäte, či spotreby energie.
- Dostupnosť - vysoká dostupnosť zariadenia.
- Schopnosť odolávať rôznym útokom a bezpečnostným incidentom.
- Ochrana súkromia.
- Škálovateľnosť, ktorá hovorí o množstve pripojených zariadení a ich manažmente.

Jednou taktiež z dôležitých vecí pre bezpečnosť je zvyšovanie bezpečnosti medzi dvoma nepriamo pripojenými zariadeniami, hovoríme o tzv. end-to-end spojení. Medzi rozšírené protokoly používané v sieťach a IoT oblasti podporujúce tento aspekt, patria hlavne tieto [17]:

1. TLS a jeho odľahčená verzia DTLS,
2. riešenia založené na protokoloch IKEv2 a IPSec,
3. protokol HIP a jeho optimalizácie.

Vo svojom predošlom výskume [10] sme sa v analýze venovali najmä spomenutým protokolom a ich existujúcim optimalizáciám. Na základe toho sme sa rozhodli pre výskum a optimalizáciu protokolu HIP.

2.1 Existujúce riešenia

Host Identity Protocol (skr. HIP) alebo aj HIP-BEX (*z angl. Basic Exchange*) je jedným z bezpečnostných protokolov používaných v sieťach. Pre použitie v oblasti Internetu vecí, vznikajú jednotlivé optimalizované, či jeho odľahčené verzie.

Vlastnosti protokolu [17, 18]:

- poskytuje rozlíšenie identifikátora a lokátora,
- poskytuje dohodu kľúčov,
- poskytuje vlastníctvo identifikátora,
- poskytuje anonymitu koncových používateľov,
- poskytuje mobilitu.

Protokol Slimfit [7] vznikol optimalizáciou protokolu HIP Diet Exchange (HIP-DEX) [15], ako kompresná vrstva pre zariadenia Internetu vecí založených na IP adrese. HIP-DEX v porovnaní s HIP-BEX používa dlhodobé verejné hodnoty Diffie-Hellman protokolu založené na Eliptických krivkách, čím redukuje energetickú náročnosť protokolu.

Na základe analýzy protokolu HIP a HIP-DEX, autori identifikovali ďalšie možnosti optimalizácia protokolu, a tak navrhli vytvorenie dodatočnej Slimfit kompresnej vrstvy, ktorá pozostáva z nasledujúcich častí:

1. integrácia do sieťového zásobníka,
2. kompresia HIP hlavičky,
3. kompresia HIP parametrov.

Skratka protokolu D-HIP [3] je odvodená od anglického slova *Distributed HIP*. Väčšina IoT zariadení má obmedzené výpočtové, či komunikačné zdroje, ktoré sú náročné na spotrebu energetickej energie. Inicializácia, či výmena správ v protokole HIP (HIP-BEX) je vysoko energeticky náročná. Naopak je to v protokole LHIP (*z angl. Lightweight HIP*), ten však nepodporuje väčšinu bezpečnostných funkcií pôvodného protokolu, aj keď je spätne kompatibilný a jeho bezpečnosť a zabezpečenie bezpečnostných vlastností je veľmi nízke. To bola hlavná motivácia pre autorov protokolu D-HIP. Optimalizovať protokol HIP tak, aby výpočtovo náročné úlohy boli realizované mimo IoT zariadenia.

HIP-TEX je ďalšou optimalizovanou verziou protokolu HIP (HIP-BEX) (*HIP-TEX z angl. Host Identity protocol Tiny Exchange*) [18]. Podobne ako iné, vznikol na základe motivácie optimalizácie náročného protokolu HIP, najmä DH výmeny a RSA podpisu. Návrh protokolu pozostáva z dvoch hlavných častí:

1. Nahradenie DH algoritmov pomocou kryptografie verejným kľúčom (PKC). Namiesto náročných výpočtov, autori navrhli výmenu dvoch tajných kľúčov zašifrovaných verejným certifikátom.
2. Kolaboratívny prístup výmeny kľúča, kde náročné výpočtové operácie sú vykonané na susedných zariadeniach, nazývaných ako proxy, rovnako ako pri D-HIP protokole.

CD-HIP (*z angl. Compressed and Distributed HIP*) v preklade znamená komprimovaný a distribuovaný protokol HIP [17]. Autori uvádzajú, že optimalizácia vzhľadom na energetickú náročnosť vyžaduje určité aplikačné alebo bezpečnostné zníženie nárokov.

Autori navrhli použitie komprimovanej hlavičky protokolu HIP s použitím protokolov 6LoWPAN a IPv6. Druhá časť optimalizácie je založená na distribúcii náročných úloh na menej energeticky kritické zariadenia, konkrétne časť náročného výpočtu pre algoritmus Diffie Hellman.

Protokol E-HIP [9, 10] bol zameraný na voľne dostupnú implementáciu OpenHIP. Pred samotným návrhom optimalizácie boli stanovené určité požiadavky na výsledný protokol ako zníženie energetickej náročnosti, zachovanie bezpečnostných vlastností, podpora IPv4 alebo IPv6, dosiahnutie podobných alebo lepších výsledkov ako podobné optimalizované riešenia a iné.
Prototypové riešenie nasadenia protokolu bolo založené na komunikačnej technológii Bluetooth, zariadenia Raspberry PI ako zástupcom zariadení IoT s operačným systémom Raspbian a osobného prenosného počítača pre reprezentáciu servera pripojeného v Internete.

P-HIP (*z angl. Lightweight and Privacy-Aware Host Identity Protoco*)[6]. Autori navrhli použitie kryptografie eliptických kriviek Qu-Vanstone (ECQV) [5]. Ve-

rejný a súkromný kľúč sú výpočítané z osobných údajov získaných od certifikačnej autority (CA). Následne sú vymenené verejné kľúče a pre vytvorenie bezpečnostnej asociácie SA je zachovaný algoritmus ECDH. Vygenerovanie nových kľúčov je možné bez dodatočného kontaktovania CA. Autori sa detailne venujú aj bezpečnostným vlastnostiam navrhnutého protokolu. Ukázali, že protokol dokáže odolať rôznym sieťovým útokom. Svoje riešenie otestovali vo virtuálnom prostredí pomocou operačného systému Contiki. Danou optimalizáciou dokázali odľahčiť pôvodný protokol o náročné kryptografické operácie a veľké certifikáty vďaka čomu dokázali ušetriť až 80% energie oproti pôvodným protokolom.

LC-DEX (*z angl. Lightweight Compressed HIP-DEX*)[4] je posledným aktuálnym protokolom (publikovaný November 2021), ktorý vznikol optimalizáciou protokolu HIP-DEX [15], podobne ako protokol Slimfit pre siete IoT. Hlavným cieľom autorov je optimalizovať protokol vzhľadom na energetickú náročnosť koncových uzlov. LC-DEX uvažuje o prepojení niekoľých rôznych optimalizačných techník, a to:

- Optimalizácia hlavičky, ale aj parametrov správ protokolu, založenú na princípoch protokolov C-HIP a E-HIP, konkrétne vynechaním niektorých polí, zadefinovaním pevného poradia jednotlivých parametrov, definovaním parametrov vopred.

- Redukcia počtu správ, kde okrem upravenia odstránenej správy ukončnia v protokole E-HIP, navrhli aj bezpečnostné dôvody nepoužívania správy typu *NOTIFY*.

- Distribúcia náročných kryptografických úloh na menej energeticky kritické zariadenia (príklad D-HIP), kde autori preniesli výpočty spojené s dohodou zdieľaného tajomstva.

V porovnaní s pôvodným protokolom HIP-DEX, autori LC-DEX dokázali získať okolo 60% energie, čím dosiahli jedno z najlepších riešení optimalizovaných protokolov.

2.2 Záver analýzy rôznych verzií HIP

V závere analyzovanej časti tejto práce sme porovnali dané protokoly z pohľadu použitého algoritmu, počtu potrebných správ pre vytvorenie spojenia, spôsobu optimalizácie, či dosiahnutého celkového zlepšenia v porovnaní s protokolom OpenHIP (viď Tabuľka 2.1, kde 'T' označuje prítomnosť danej technológie, '-' označuje nepoužitie, 'na' znamená nedostupná informácia).

Tabuľka 2.1: Porovnanie protokolov založených na HIP

Protokol	RSA	ECC	PSK	Spätná kompatibilita	DH	Počet prenesených správ (inicializačná fáza)	Kompresia správ	Redukcia výpočtovej zložitosti	Celková optimalizácia (v porovnaní s OpenHIP)
OpenHIP	T	T (HIPv2)	-		T	4	-	-	
HIP-DEX	-	T	-		T	4	-	T	
L-HIP	-	-	-	T	-	4	-	T	na
Slimfit	-	T	-	T	T	4	T	-	25 %
D-HIP	-	-	-	na	T	6	-	T	22 %
HIP-TEX	-	-	-	na	T	6	T	T	na
CD-HIP	-	T	T	na	T	7	T	T	48 %
E-HIP	T	T (HIPv2)	-	-	T	4	T	-	20 %
P-HIP	-	T (ECQV)	T	na	T	4	T	T	80 %

3. Ciele dizertačnej práce

Bezpečnosť a oblasť Internetu vecí sa čoraz viac dostáva do popredia a s tým súvisí aj výskum v tejto doméne informatiky a informačných technológií. Samotný výskum v tejto oblasti je založený na už existujúcom predošlom výskume [9], kde sa objavili možnosti jeho pokračovania.

Na základe vykonanej analýzy a predošlého výskumu v oblasti Bezpečnosť v Internete s dôrazom na energetickú náročnosť bezpečnostných protokolov Internetu vecí sme dospeli k nasledujúcim cieľom, respektíve tézam tejto práce:

- Návrh energetického zefektívnenia zabezpečenia komunikácie kombináciou nových a existujúcich optimalizačných metód, pričom zameranie bude na koncové uzly Internetu vecí citlivé na energetickú spotrebu.
- Overenie a vyhodnotenie riešenia (vzhľadom na energetickú náročnosť a stupeň zabezpečenia) s použitím operačného systému (OS) pre IoT zariadenia alebo implementáciou na zariadení bez používateľského prístupu k OS.
- Overenie a vyhodnotenie riešenia (vzhľadom na energetickú náročnosť a stupeň zabezpečenia) nasadením do niektorej modernej komunikačnej technológie určenej pre Internet vecí.
- Overenie funkčnosti protokolu a jeho bezpečnostných vlastností jednou z určených techník.
- Porovnanie navrhnutého protokolu s existujúcimi protokolmi.

4. Návrh optimalizovaného protokolu

V tejto časti práce sme navrhli použitie jednotlivých možností optimalizácie protokolov na nami navrhovaný nový protokol.

4.1 Výber optimalizačných metód

Na základe analyzovanej časti existujúcich riešení protokolu HIP, ale aj iných bezpečnostných protokolov pre použitie v oblasti Internetu vecí sme dospeli k niekoľkým možným spôsobom:

- Optimalizácia protokolov pomocou redukcie veľkosti prenášaných riadiacich správ, a to konkrétne:
  1. Redukcia veľkosti hlavičiek protokolu.
  2. Redukcia alebo odstránenie prenášaných parametrov protokolu.
- Optimalizácia pomocou distribúcie náročných úloh na iné menej kritické zariadenia Internetu vecí.
- Optimalizácia pomocou redukcie počtu riadiacich správ, čo súvisí aj s odstránením jednotlivých stavov protokolu.
- Optimalizácia pomocou prídavných kryptografických modulov.
- Optimalizácia použitím nových matematických spôsobov pre výpočet náročných úloh.

Tieto jednotlivé optimalizačné spôsoby majú za cieľ znížiť spotrebu elektrickej energie, a tým predĺžiť životnosť daných zariadení, či náklady spojené s ich údržbou ako je napríklad výmena batérie alebo jej samotné dobíjanie. V tabuľke 4.1 uvádzame prehľad použitých optimalizačných metód jednotlivými optimalizovanými verziami protokolu HIP. Ako môžme vidieť väčšina protokolov využíva jednu alebo kombináciu dvoch optimalizačných metód. Kryptografický hardvér nebol použitý ani v jednom protokole, preto ho v tabuľke neuvádzame.

Tabuľka 4.1: Prehľad použitých optimalizačných metód jednotlivými protokolmi

	Slimfit	**D-HIP**	**HIP-TEX**	**CD-HIP**	**E-HIP**	**P-HIP**	**Návrh**
Redukcia hlavičky	*			*			*
Redukcia parametrov	*				*		*
Distribúcia		*	*	*			*
Redukcia správ					*		*
Optimalizácia algoritmov			*			*	

Z predošlého výskumu a analyzovanej časti sa ukázalo, že autori protokolu CD-HIP dokázali správnou kombináciou ušetriť viac energie potrebnej na fungovanie protokolu. Vzhľadom na tento fakt, sme sa rozhodli prepojiť viacero týchto optimalizačných metód ako redukcia veľkosti hlavičky, redukcia parametrov, distribúcia náročných výpočtových úloh na menej kritické zariadenia a redukcia správ.

Distribuovaný prístup: Pri návrhu protokolov s použitím distribuovaného prístupu je potrebné uvažovať najmä o nasledujúcich veciach:

1. komunikácia s prídavným zariadením nazývaným Proxy,
2. distribúcia náročných výpočtových úloh, ktoré sa budú vykonávať na zariadení Proxy mimo svojho pôvodného zariadenia,
3. zabezpečenie komunikácie s Proxy zariadením v prípade citlivých údajov.

Návrh rozdelenia úloh - Pri danom prístupe je potrebné vedieť aké úlohy sa odohrávajú na jednotlivých stranách pôvodného protokolu HIP. Na základe toho je možné navrhnúť prenos niektorých, tých náročnejších, na zariadenie Proxy. V práci sa zameriame najmä na procesy prebiehajúce na senzorovom zariadení, ktoré

bude zohrávať rolu Iniciátora. Úlohy protokolu OpenHIP role iniciátora môžme rozdeliť do troch fáz podľa komunikujúcich správ, na úlohy spojené s odosielaním správy I1, prijatím správy R1 a odoslaním správy I2, prijatím správy R2. Nakoľko správa I1 je inicializačná, neobsahuje žiadne dôležité úlohy. Posledná prijatá správa R2 slúži na dokončenie inicializačnej výmeny a chráni protokol pred útokom replikácie (z angl. *Replay attack*). Nakoľko neobsahuje žiadne dôležité úlohy, našou snahou bude ju odstrániť, prípadne nahradiť jednoduchšou menšou správou v ďalšej časti práce. Najdôležitejšie úlohy sa odohrávajú v prostrednej fáze, kde je možné výpočet riešenia úlohy puzzle automaticky presunúť na zariadenie proxy. Výpočet DH tajomstva by bolo možné presunúť za predpokladu, že by sme zabezpečili bezpečnú komunikáciu medzi zariadením proxy a senzorom prostredníctvom jednoduchej kryptografie, napríklad vopred zdieľaným tajomstvom. To nám však môže pridať dodatočné správy a celková efektívnosť nemusí byť postačujúca. Podobne sa to týka aj ďalších výpočtových úloh, ktoré nie sú tak náročné a práve ich distribuovanie na iné zariadenie pomocou dodatočných správ môže mať nežiadúci účinok zvýšenia spotreby energie.

Návrh prebiehajúcich správ - Autori protokolov D-HIP a HIP-TEX dbali na to, aby počet prenesených inicializačných správ zostal zachovaný, narozdiel od protokolu CD-HIP. Jeho distribuovaná časť práve pridala niekoľko správ pre komunikáciu so zariadením Proxy, čo môže mať vplyv na energetickú náročnosť protokolu. Vzhľadom na to sme sa rozhodli v našom návrhu použiť podobný distribuovaný prístup protokolu D-HIP, ktorý nám zachová počet inicializačných správ tak, aby sme splnili požiadavky špecifikácie. Na základe analyzovanej časti existujúcich riešení protokolu HIP usudzujeme, že daný distribuovaný návrh je najefektívnejší. Pôvodná správa I2 je nahradená novými správami I2a/I2b.

Redukcia hlavičky: Pri redukcii hlavičky sme sa inšpirovali protokolom CD-HIP, ktorý zredukoval pôvodnú hlavičku o veľkosti 320 bitov na novú komprimovanú veľkosť 205 bitov. Táto veľkosť však nie je zarovnaná na veľkosť slova, preto sme navrhli veľkosť 208b, pričom narozdiel od CD-HIP protokolu nepridávali sme novú dodatočnú hlavičku LOWPAN_NHC_HIP, ale upravili sme pôvodnú. Zarovnanie veľkosti parametrov na veľkosť slova nie je nevyhnutné, ale vzhľadom

na pôvodnú verziu a ich spracovanie pri spracovaní jednotlivých správ vhodnejšie. Odstránili sme polia ako dĺžka hlavičky (hlavička je statická), verziu protokolu, nakoľko ide o optimalizovaný protokol, ktorý nie je spätne kompatibilný, kontrolnú sumu, či dodatočné pole kontrolných bitov. Naopak zachovali sme identifikátor ďalšej hlavičky, typ paketu / správy.

Redukcia parametrov: Podobne ako pri redukcii hlavičky, aj tu existuje niekoľko možností ako optimalizovať veľkosť prenášaných parametrov. Vhodným kompromisom je návrh protokolu E-HIP, kde boli vynechané niektoré parametre ako napríklad HI-R a zadefinovaním pevného poradia a veľkosti parametrov dokázal odstrániť polia obsahujúce informáciu o type a veľkosti parametra. Iný prístup je implementovaný v protokole Slimfit, ktorý používa základné nastavenia parametrov a jednotlivými nastavenými bitmi hovorí o ich použití. Zatiaľ čo protokol E-HIP použil jeho prístup len v jednej z prenášaných správ, rozhodli sme sa to implementovať na všetky základné typy správ protokolu.

Redukcia počtu správ: Protokol HIP okrem základných správ potrebných na základné otvorenie spojenia (*z angl. Base Exchange*), obsahuje ešte niekoľko doplnkových správ. Pri nami navrhovanom protokole sme sa inšpirovali protokolom E-HIP (časť 2.1), ktorý odstránil potvrdzovaciu správu ukončenia. Ostatné správy však zachoval. V prvotnom prototypovom riešení návrhu protokolu nebudeme uvažovať ani informačnú správu a správu aktualizácie dohodnutých parametrov, čo súvisí aj s navrhovaným prípadom použitia protokolu. Dané správy budú odstránené podobne ako správa pre potvrdenie ukončenia.

Návrh reálneho prostredia Finálne prostredie navrhnutého riešenia bude pozostávať z troch súčastí, a to: senzorového zariadenia alebo zariadenia Internetu vecí, zariadenia Proxy, ktoré môže zohrávať aj rolu brány (gateway), vzdialeného servera umiestneného v Internete. Na implementáciu prototypu sme vybrali nám dostupné zariadenia: ESP32-DEVKITC-32U / NodeMCU32S ESP32 - zariadenie bez OS, podporujúce Bluetooth 4.2 a WiFi priamo na doske, Raspberry PI 4 - rovnako podporuje BLE *z angl. Bluetooth Low Energy* a WiFi, Server Ubuntu 16 a vyššie - klasický zástupca serverového OS.

5. Implementácia prototypu S-HIP

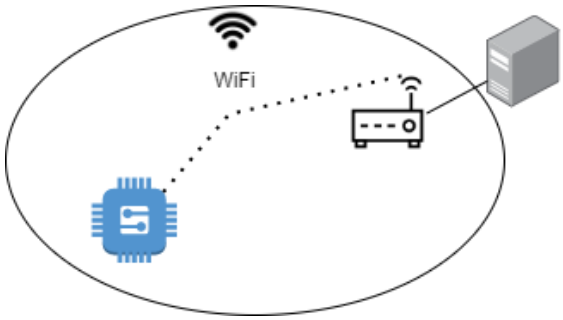
V tejto kapitole opisujeme konkrétnu implementáciu nami navrhovaného optimalizovaného protokolu S-HIP (*z angl. Simple HIP*). Samotné riešenie je naprogramované v jazyku C v prostredí Arduino. Vzhľadom na zameranie práce, a to optimalizácie bezpečnostného protokolu pre Internet vecí, hlavná implementácia je orientovaná na samotné zariadenie, konkrétne model ESP32-DEVKITC-32U. Samotný výber zariadenia môže závisieť od viacerých faktorov. Zvolené zariadenie ESP32-DEVKITC-32U poskytuje viacero požiadaviek pre testovanie nášho riešenia ako napríklad:

1. režim hlbokého spánku,
2. hardvérovú akceleráciu pre kryptografické operácie (AES, ECC, RNG),
3. dve komunikačné technológie na doske (WiFi, Bluetooth).

Implementácia jednotlivých častí protokolu: Priebeh protokolu je implementovaný na základe stavového automatu. Ako bolo spomínané, jednotlivé prechody medzi stavmi prebiehajú pri odosielaní a prijímaní jednotlivých správ. Po zapnutí zariadenia a jeho inicializácii, sa protokol nachádza v stave neasociovaný (označenie stavu "0"). Plne asociovaný protokol je v stave označovanom ako nadviazané spojenie (označenie stavu "3"), počas ktorého prebieha výmena správ. Pri objavení akéhokoľvek chybového stavu pri inicializácii spojenia, protokol automaticky prechádza do stavu neasociovaný, a je pripravený podľa konfigurácie používateľa prepnúť sa do stavu hlbokého spánku na vopred zadefinovaný časový

interval alebo opätovne sa pokúsiť o nadviazanie spojenia so serverom.

Testovacie prostredie: Hlavným cieľom práce je optimalizovať výslednú spotrebu energie protokolu na zariadení Internetu vecí, čím za predĺži jeho životnosť a nutnosť dodatočného nabíjania, či výmeny batérie. Z tohto dôvodu pre overenie nami navrhovaného riešenia môžme vynechať implementáciu Proxy zariadenia. Finálne testovacie prostredie je zobrazené na Obr. 5.1.



Obr. 5.1: Testovacie prostredie

Implementácia správ : Pôvodná implementácia protokolu HIP, z ktorej sme vychádzali pri návrhu optimalizácii protokolu E-HIP [9], a na základe jeho výsledkov a analýzy aj tohto protokolu S-HIP, je implementovaná nad protokolom UDP, t.j. všetky svoje správy enkapsuluje a prenáša pomocou tohto protokolu. V prostredí Arduino sme na to využili knižnicu *WiFiUdp.h*.

Implementácia kryptografických algoritmov / protokolov: Protokol HIPv2 implementuje všetky použité kryptografické operácie pomocou eliptických kriviek. Ich hlavnou výhodou oproti klasickým algoritmom sú menšie šifrovacie kľúče. Výber jednotlivých algoritmov pre zariadenie Internetu vecí v prostredí Arduino sme realizovali na základe ich porovnania[1] a dostupnej implementácie pre dané zariadenie.

Digitálny podpis (*Signature*) - na jeho implementáciu sme použili knižnicu *Ed25519.h* s použitím asymetrických (verejný, súkromný) kľúčov o veľkosti 256 bitov. Vygenerované kľúče pre samotné IoT zariadenie, ale aj verejný kľúč servera alebo proxy

[1] https://rweather.github.io/arduinolibs/crypto.html

zariadenia sú nahraté na zariadenie vopred. Podpísanie a verifikácia protokolov je implementované funkciami *sign()* a *verify()*.

HMAC - knižnica *BLAKE2s.h* bez použitia kľúča bola vybratá pre vytvorenie hašovaného autentifikačného kódu správy.

Diffie-Hellman výmena kľúčov - je implementovaná pomocou *Curve25519.h* knižnice. Samotná implementácia spočíva v zavolaní dvoch funkcií a výmene jednotlivých hodnôt medzi komunikujúcimi stranami. Výsledkom je zdieľané tajomstvo pre šifrovanie dátových správ pomocou agloritmu AES.

AES - implementovaný pomocou knižnice *AES.h*. Daná knižnica podporuje ECB mód s použitím 256 bitového šifrovacieho kľúča. Je potrebné nastavenie kľúča a následne šifrovanie prebieha v bloku.

Server - na implementáciu jednotlivých algoritmov na serveri v prostredí Python sme použili nasledovné knižničné ekvivalenty k už spomínaným knižniciam pre prostredie Arduino:

1. Diffie-Hellman - knižnica *donna25519*,
2. Digitálny podpis - knižnica *ed25519*,
3. HMAC - knižnica *hmac* s použitím *hashlib.blake2s*,
4. AES - knižnica *Crypto.Cipher*.

Hlboký spánok: Jedným z návrhov celkového riešenia je aj prechod zariadenia do stavu hlbokého spánku. Existuje niekoľko režimov šetrenia energie a možností návratu do normálneho stavu[2][3]. Rozhodli sme sa použiť hlboký spánok, a to z toho dôvodu, že okrem nastavenia časovača pre následný návrat do normálneho stavu, poskytuje aj externé prebudenie, či prebudenie tlačidlom. Pre prechod do hlbokého spánku sme nastavili časovač a zavolali funkciu na jeho spustenie. Finálne riešenie však môže byť prispôsobené podľa požiadaviek zákazníka.

[2]https://randomnerdtutorials.com/esp32-deep-sleep-arduino-ide-wake-up-sources/

[3]https://www.mischianti.org/2021/03/15/esp32-practical-power-saving-deep-sleep-and-hibernation-3/

6. Overenie riešenia

Overenie funkčnosti : Prvým krokom overenia je overenie protokolu na základe jeho funkčnosti odoslaním zašifrovaných dát.

Funkčnosť protokolu sme overili realizáciou jednotlivých výpisov zo zariadenia a odchytením odoslaných správ pomocou nástroja *Wireshark*. Výpisom jednotlivých krokov na oboch stranách protokolu zobrazených v ukážkach, ich vizuálnou kontrolou, odchytením správ programom Wireshark a dešifrovaním odoslaných správ môžme usúdiť, že funkčnosť prototypu je správna. Počas testovania a vývoja sme testovali aj rôzne chybové stavy. Protokol sa vždy správal podľa zadefinovanej špecifikácie, t.j. po vypršaní daného časovača sa vrátil do stavu neasociovaný.

Overenie zefektívnenia vzhľadom na veľkosť jednotlivých správ: Energetická náročnosť protokolu spočíva v dvoch základných typoch spotreby energie, a to:

1. spotreba energie potrebnej na komunikáciu,
2. spotreba energie potrebnej na výpočet jednotlivých úloh na procesore.

Z tohto dôvodu, jedným z dôležitých optimalizačných prístupov je redukcia celkových prenesených dát v správach. Optimalizácia protokolu S-HIP je založená na redukcii riadiacich správ, konkrétne inicializačné správy a správy ukončenia. V predošlom výskume [9] bola dosiahnutá určitá optimalizácia protokolu E-HIP vzhľadom na veľkosť a počet riadiacich správ. Pri protokole S-HIP je implementovaných viacero optimalizačných techník. Porovnanie pôvodného protokolu OpenHIP a jeho optimalizácie E-HIP s protokolom S-HIP uvádzame v tabuľke 6.1. Pri

tomto type porovnania nedisponujeme hodnotami ani implementáciou iných verzií optimalizovaných protokolov okrem vyššie spomenutých, ktorým sme sa venovali v našom predošlom výskume.

Tabuľka 6.1: Porovanie veľkosti prenášaných riadiacich správ protokolov OpenHIP, E-HIP a S-HIP (TX - odoslané, RX - prijaté)

	OpenHIP	E-HIP	S-HIP	OpenHIP / S-HIP	E-HIP / S-HIP
RX bajty	1184	760	358	70 %	53 %
TX bajty	1048	1032	485	54 %	53 %
Celkovo bajtov	2232	1792	843	62 %	53 %

Overenie z hľadiska energetickej náročnosti : Z pohľadu charakteru práce, a to konkrétne optimalizácie bezpečnostného protokolu pre Internet vecí so zameraním na energetickú náročnosť protokolu, ale aj celkového riešenia, najdôležitejším overením je meranie energetickej spotreby. Pre odmeranie spotreby energie na IoT zariadení sme sa rozhodli použiť zariadenie *UM25c* určené na tieto potreby. V prvej fáze merania sme realizovali meranie spotreby energie počas opakovanej inicializačnej výmeny správ protokolu. Na začiatok sme realizovali 20 meraní, počas ktorých sme merali napätie a prúd na zariadení pri frekvencii mikroprocesora 240 MHz. Dané hodnoty sme zapísali do tabuľky a vypočítali priemerné namerané hodnoty. Z nameraných hodnôt sme zistili, že priemerná spotreba energie potrebnej pre inicializačnú výmenu správ protokolu je 43,722 mJ. Je potrebné uviesť, že dané hodnoty sme dosiahli s použitím technológie WiFi. Pri použití inej, menej energeticky náročnej technológii, napríklad Bluetooth, ktorý sme uviedli ako jednu z možností pri návrhu protokolu, by sme mohli dosiahnuť vyššiu energetickú úsporu.

Overenie funkčnosti kryptografických operácií: Toto overenie bolo realizované najmä výpisom jednotlivých výstupov z kryptografických algoritmov. Diffie-Hellman a AES: overením, že dohoda zdieľaného kľúča pomocou tohto algoritmu funguje správne bolo vypísanie zdieľaného tajomstva na oboch stranách protokolu, a to IoT zariadenia a Servera, ale aj zašifrovaním a dešifrovaním odosielaných dát

pomocou algoritmu AES, čím sme overili aj jeho správnu implementáciu. AES šifrovanie bolo overené aj pomocou online dostupného nástroja. HMAC a Digitálny podpis: overenie realizované pomocou výpisov na jednotlivých stranách. Dôležité výpisy, pomocou ktorých je možné konštatovať, že kryptografické operácie fungujú správne sú *"Sign. verified"* a *"HMAC: match"*, resp. porovnaním vypísaných HMAC reťazcov na oboch stranách.

Overenie vzhľadom na frekvenciu procesora: Ďalším vplyvom na spotrebu energie, o ktorom sme neuvažovali pri návrhu riešenia, či analýze existujúcich riešení, je aj frekvencia mikroprocesora [1]. Pri prvých experimentoch (časť 6), mikroprocesora bežal na frekvencii 240 MHz. Túto frekvenciu je však možné znížiť, čím sa zabezpečí jeho menšia energetická spotreba. Vyskúšali sme teda experimenty pri použití 80 MHz a 160 MHz a porovnali ich s experimentom vykonaných s 240 MHz. Podobne ako pri pôvodných experimentoch sme odčítali spotrebu energie zariadenia pri jeho nulovom zaťažení. Výsledky uvádzame v Tabuľke 6.2.

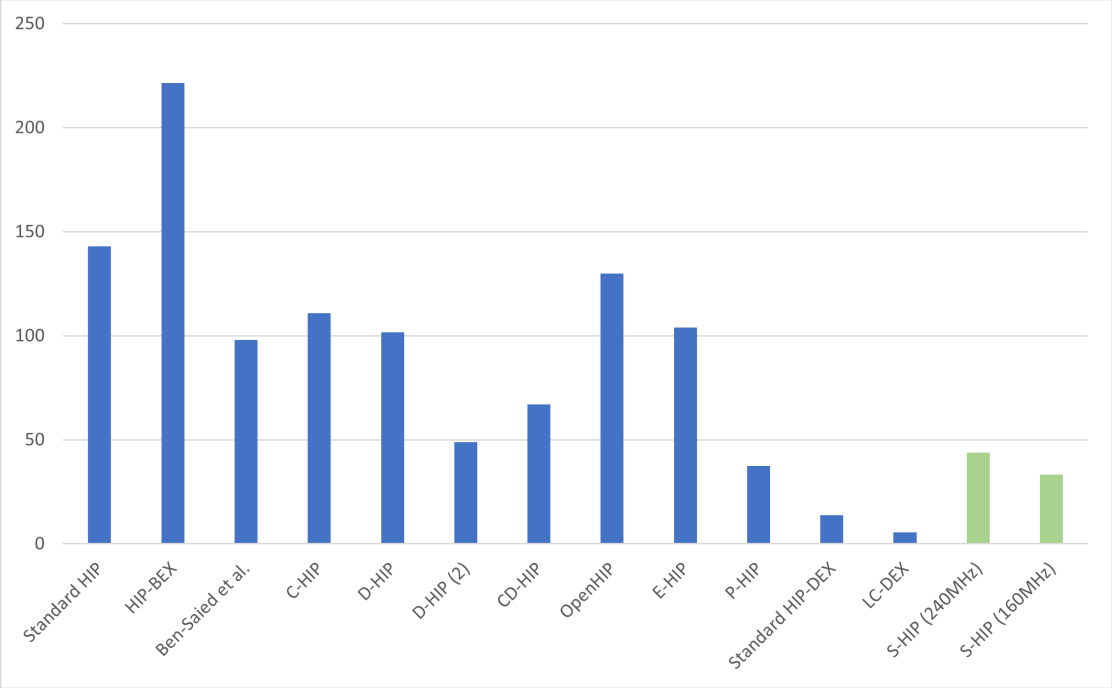
Tabuľka 6.2: Hodnoty namerané pri jednotlivých frekvenciách procesora

Frekvencia mikroprocesora (MHz)	**Priemerné napätie (V)**	**Priemerný prúd (A)**	**Prúd zariadenia bez záťaže (A)**	**Výkon (W)**	**Priemerný čas (s)**	**Energia (J)**
240	5.063730159	0.111564286	0.055	0.286426	0.1526464	0.04372194
80	5.063730159	0.08	0.045	0.177231	0.26	0.046079944
160	5.063730159	0.0968	0.055	0.211664	0.157	0.033231236

Z vykonaných experimentov možno usúdiť, že spotreba prúdu zariadenia pri jeho nulovom zaťažení klesla o 0,01 A pri frekvencii 80 MHz, zatiaľ čo pri 160 MHz ostala rovnaká ako pri 240 MHz. Taktiež sa zmenil aj čas potrebný na inicializačnú výmenu správ. Čas výrazne narástol pri znížení frekvencie o 2/3, no pri 160 MHz sa zmenil iba minimálne. Z toho vyplýva, že energia E spotrebovaná protokolom pri nižšej frekvencii je menšia, a to približne o 10 mJ (E = 33 mJ).

[1] https://www.mischianti.org/2021/03/06/esp32-practical-power-saving-manage-wifi-and-cpu-1/

Porovnanie s existujúcimi riešeniami: V analýze práce sme uviedli niekoľko rôznych optimalizačných techník, ktoré boli použité pri návrhoch jednotlivých optimalizácii protokolu HIP. Našim hlavným cieľom dizertačnej práce bolo vytvorenie optimalizovaného protokolu HIP pre použitie v prostredí Internetu vecí kombináciou viacerých optimalizačných metód. Vychádzali sme z predpokladu, že ich vzájomná kombinácia by mohla priniesť podobné výsledky ako dosiahli autori protokolu CD-HIP, ktorý vznikol spojením dvoch optimalizačných metód. Porovnanie výsledkov uvádzame na Obr. 6.1. Je možné vidieť, že pri porovnaní s ostatnými optimalizovanými protokolmi, ktoré vychádzali z originálneho protokolu HIP, S-HIP je najlepším riešením z hľadiska spotreby energie.



Obr. 6.1: Grafické porovnanie energetickej náročnosti (os y - energia E [mJ])

Nameranou priemernou spotrebou energie protokolu S-HIP ($E_{240MHz} = 43{,}722$ mJ, $E_{160MHz} = 33{,}231$ mJ) sme ukázali, že kombináciou viacerých optimalizačných techník je možné dosiahnuť jednu z najlepších energetických úspor v porovnaní s inými optimalizovanými protokolmi, čo dokazuje aj protokol LC-DEX, ktorý vznikol podobným prístupom ako nami navrhovaný protokol S-HIP.

7. Záver

Priemyselná revolúcia 4.0 a postupná automatizácia procesov spôsobili to, že sa postupne pripája čoraz viac a viac zariadení do siete. Takto vytvorenú sieť nazývame Internet vecí. Tieto zariadenia častokrát sú napájané pomocou batérií z dôvodu potreby ich mobility, či nedostupnosti napájania z elektrickej siete. Jedným z odvetví života, kde počet zariadení narastá, je aj oblasť zdravotníctva. Tieto zariadenia častokrát prenášajú dáta o zdravotnom stave pacienta. Z tohto dôvodu je dôležité zabezpečenie ich komunikácie proti odchyteniu a zneužitiu dát útočníkmi. Existuje niekoľko protokolov na zabezpečenie komunikácie medzi zariadeniami a serverom na rôznych vrstvách OSI alebo IoT modelu zásobníka. Mnohé protokoly však sú energeticky náročné pre použitie v prostredí Internetu vecí. To je jeden z dôvodov, prečo vznikajú nové optimalizované riešenia s použitím rôznych optimalizačných prístupov ako je optimalizácia správ vzhľadom na ich veľkosť, či prenášaný počet, optimalizácia kryptografických algoritmov alebo použitie menej energeticky náročných variantov, rozdelenie záťaže na zariadenia so stálym pripojením k zdroju, či v neposlednom rade aj vhodným výberom IoT zariadenia, či použitej komunikačnej technológie vzhľadom na aplikačné požiadavky používateľa.

Jedným z takto optimalizovaných protokolov je aj protokol HIP a jeho varianty. V práci sme analyzovali použité optimalizačné prístupy jednotlivými optimalizáciami protokolu. Na základe analýzy a nášho predošlého výskumu sme navrhli vytvorenie optimalizovaného protokolu HIP pre použitie v Internete vecí spojením existujúcich a nových optimalizačných metód. Na základe návrhu sme implementovali prototypové riešenie protokol S-HIP (*Simplified HIP for IoT networks*) a jeho implementáciu overili v testovacom prostredí s použitím zariadenia ESP32 a tech-

nológie WiFi. Dokázali sme, že prístupom kombinácie čo najviac optimalizačných metód, je možné dosiahnuť najlepšiu energetickú optimalizáciu pôvodného protokolu HIP vzhľadom na spotrebu energie. Podobným prístupom publikovaným v Novembri 2021 sa venovali aj autori LC-DEX [4], ktorý však určili jeho energetickú náročnosť na základe teoretického výpočtu spotreby energie procesora a komunikačnej technológie, avšak ich optimalizácia vychádzala z už optimalizovanej verzie HIP-DEX, čo vzhľadom na dostupnosť implementácie nebolo v našom prípade možné.
Vyriešením stanovených cieľov dizertácie sme dosiahli tieto pôvodné vedecké výsledky:

1. Teoretické:
    - Návrh pôvodnej modifikácie protokolu HIP, ktorá spĺňa stanovené optimalizačné kritériá.
    - Overenie funkčnosti a optimálnosti protokolu S-HIP.
2. Praktické:
    - Energetická úspora zabezpečenia IoT siete.
    - Ekvivalentná bezpečnosť pôvodného a optimalizovaného protokolu.

Protokol je navrhnutý pre aplikácie, ktoré nevyžadujú nepretržité otvorené spojenie, t.j. pre prípady použitia, kde IoT zariadenie môže väčšinu času byť v stave nečinnosti, prípadne vykonávať činnosť samostatne a komunikovať len v určitých zadefinovaných intervaloch alebo iný vstupný signál. Vytvorené spojenie je vždy z jedného uzla na druhé, s oddeleným identifikátorom od IP adresy. V ďalšej práci by bolo vhodné implementovať navrhnutý protokol s použitím menej energeticky náročnou komunikačnou technológiou ako WiFi, prípadne použiť novšie matematické varianty kryptografických protokolov.

8. Publikačná činnosť

ADM Vedecké práce v zahraničných časopisoch registrovaných v databázach Web of Science alebo SCOPUS

1. KAŇUCH, Peter - MACKO, Dominik. E-HIP: An Energy-Efficient OpenHIP-Based Security in Internet of Things Networks. In Sensors. Vol. 19, iss. 22 (2019), s. 1-17. ISSN 1424-8220 (2019: 3.275 - IF, Q1 - JCR Best Q, 0.653 - SJR, Q1 - SJR Best Q). V databáze: WOS: 000503381500100 ; SCOPUS: 2-s2.0-85074888000 ; DOI: 10.3390/s19224921.

AFC Publikované príspevky na zahraničných vedeckých konferenciách

1. KAŇUCH, Peter - MACKO, Dominik. Optimizing Energy Efficiency of Secured IoT Communication by OpenHip. In TSP 2019 : 42nd International conference on telecommunications and signal processing. Budapest, Hungary. July 1-3, 2019. Danvers : IEEE, 2019, S. 174-177. ISBN 978-1-7281-1864-2. V databáze: SCOPUS: 2-s2.0-85071067839 ; DOI: 10.1109/TSP.2019.8769096.

2. KAŇUCH, Peter - MACKO, Dominik - HUDEC, Ladislav. Survey: Classification of the IoT Technologies for Better Selection to Real Use. In TSP 2020 : 43 rd International Conference on Telecommunications and Signal Processing. 1. vyd. Piscataway : Institute of Electrical and Electronics Engineers, 2020, S. 500-505. ISBN 978-1728-1-6376-5. V databáze: SCOPUS: 2-s2.0-85090571677.

3. GAJDOŠÍK, Adam - KAŇUCH, Peter. Security vulnerability analysis of OpenHIP and E-HIP protocols. In TSP 2021 : 44th International conference

on telecommunications and signal processing. Virtual Conference. July 26-28, 2021.

AFD Publikované príspevky na domácich vedeckých konferenciách

1. KAŇUCH, Peter - MACKO, Dominik - HUDEC, Ladislav. HIP-Based Security in IoT Networks: A comparison. In ICETA 2020 : proceedings, 18th IEEE International conference on emerging elearning technologies and applications, November 12-13, 2020, virtual conference Košice, Slovakia. 1. vyd. Danvers : IEEE, 2020, S. 283-289. ISBN 978-0-7381-2366-0.
2. VANEK, Samuel; KAŇUCH, Peter; NEMČÍK, Ján. Statistical Metadata Analysis of Mobile Phone Communications. In: 2021 19th International Conference on Emerging eLearning Technologies and Applications (ICETA). IEEE, 2021. p. 414-419.

Akceptované príspevky na zahraničné vedecké konferencie

1. KAŇUCH, Peter - MACKO, Dominik - HUDEC, Ladislav. S-HIP: Simplified HIP Protocol for IoT Networks. TSP 2022 45th International Conference on Telecommunications and Signal Processing.

Literatúra

[1] Luigi Atzori, Antonio Iera a Giacomo Morabito. "The internet of things: A survey". In: *Computer networks* 54.15 (2010), s. 2787–2805.

[2] Sachin Babar et al. "Proposed security model and threat taxonomy for the Internet of Things (IoT)". In: *International Conference on Network Security and Applications*. Springer. 2010, s. 420–429.

[3] Yosra Ben Saied a Alexis Olivereau. "(k, n) threshold distributed key exchange for HIP based internet of things". In: *Proceedings of the 10th ACM international symposium on Mobility management and wireless access*. ACM. 2012, s. 79–86.

[4] Balkis Bettoumi a Ridha Bouallegue. "LC-DEX: Lightweight and Efficient Compressed Authentication Based Elliptic Curve Cryptography in Multi-Hop 6LoWPAN Wireless Sensor Networks in HIP-Based Internet of Things". In: *Sensors* 21.21 (2021), s. 7348.

[5] M Campagna. "SEC 4: Elliptic curve Qu-Vanstone implicit certificate scheme (ECQV)". In: *institution content-type= institution> Certicom Res</institution>., Mississauga, ON, Canada, Tech. Rep* (2013).

[6] Mahmud Hossain a Ragib Hasan. "P-HIP: A Lightweight and Privacy-Aware Host Identity Protocol for Internet of Things". In: *IEEE Internet of Things Journal* (2020).

[7] René Hummen et al. "Slimfit—A HIP DEX compression layer for the IP-based Internet of things". In: *2013 IEEE 9th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*. IEEE. 2013, s. 259–266.

[8] *Internet of Things (IoT)*. https://www.techopedia.com/definition/28247/internet-of-things-iot. Accessed: 2018-02-19.

[9] Peter Kaňuch a Dominik Macko. "E-HIP: An Energy-Efficient OpenHIP-Based Security in Internet of Things Networks". In: *Sensors* 19.22 (2019), s. 4921.

[10] Peter Kaňuch. "Zefektívnenie zabezpečenia komunikácie zariadení Internetu vecí pomocou OpenHip". https://opac.crzp.sk/?fn=detailBiblioForm&sid=A6067EBA35451301816FD90D9AAB. Dipl. pr. Slovenská technická univ. v Bratislave FIIT UPAI (FIIT), 2019.

[11] KRISTOPHER SANDOVAL. *OAuth 2.0 – Why It's Vital to IoT Security*. https://nordicapis.com/why-oauth-2-0-is-vital-to-iot-security/. Accessed: 2018-02-19.

[12] Gyu Myoung Lee et al. "The Internet of Things–Concept and Problem Statement. July 2011". In: *Internet Research Task Force* (2011).

[13] Milan Milenkovic. *Internet of Things: Concepts and System Design*. Springer, 2020.

[14] Jacob Morgan. "A Simple Explanation Of'The Internet Of Things'". In: *Retrieved November* 20 (2014), s. 2015.

[15] HIPWGR Moskowitz a M Komu. "HIP Diet EXchange (DEX) draft-ietf-hip-dex-18". In: *HIP* 5 (2020), s. 1.

[16] Kim Thuat Nguyen, Maryline Laurent a Nouha Oualha. "Survey on secure communication protocols for the Internet of Things". In: *Ad Hoc Networks* 32 (2015), s. 17–31.

[17] Somia Sahraoui a Azeddine Bilami. "Efficient HIP-based Approach to Ensure Lightweight End-to-end Security in the Internet of Things". In: *Comput. Netw.* 91.C (nov. 2015), s. 26–45. ISSN: 1389-1286. DOI: 10.1016/j.comnet.2015.08.002. URL: http://dx.doi.org/10.1016/j.comnet.2015.08.002.

[18] Yosra Ben Saied a Alexis Olivereau. "HIP Tiny Exchange (TEX): A distributed key exchange scheme for HIP-based Internet of Things". In: *Third International Conference on Communications and Networking*. IEEE. 2012, s. 1–8.

[19] Harald Sundmaeker et al. “Vision and challenges for realising the Internet of Things”. In: *Cluster of European Research Projects on the Internet of Things, European Commision* 3.3 (2010), s. 34–36.