VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

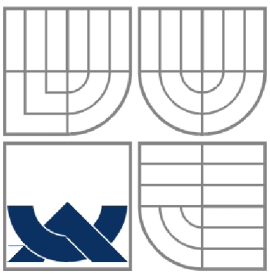
FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

ZABEZPEČENÍ BEZDRÁTOVÝCH SENZOROVÝCH SÍTÍ

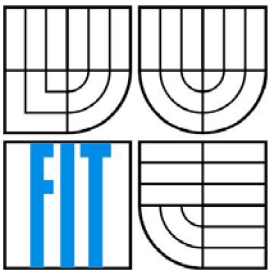
DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE Bc. JAN NAGY
AUTHOR

BRNO 2007



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

ZABEZPEČENÍ BEZDRÁTOVÝCH SENZOROVÝCH SÍTÍ

THE SECURITY OF THE WIRELESS SENSOR NETWORKS

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

Bc. JAN NAGY

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. PAVEL OČENÁŠEK

BRNO 2007

Vysoké učení technické v Brně - Fakulta informačních technologií

Ústav informačních systémů Akademický rok 2006/2007

Zadání diplomové práce

Řešitel: **Nagy Jan, Bc.**
Obor: Informační systémy
Téma: **Zabezpečení bezdrátových senzorových sítí**
Kategorie: Počítačové sítě

Pokyny:

1. Seznamte se s technologií ZigBee a vývojem jejího standardu.
2. Seznamte se s existujícími metodami zabezpečení v této technologii.
3. Proveďte analýzu požadavků na zabezpečení uvedené technologie. Proveďte srování existujících metod zabezpečení vzhledem k výsledkům analýzy.
4. Diskutujte aktuální stav zabezpečení uvedené technologie. Uveďte otevřené problémy/nedostatky a stav jejich řešení.
5. Bezpečnostní funkce prakticky odzkoušejte v laboratoři ZigBee, případně odsimulujte.
6. Navrhněte a implmentujte jednoduchou aplikaci demonstrující síťovou komunikaci a zaměřte se na její zabezpečení.
7. Diskutujte získané výsledky a pokračování vývoje zabezpečení této technologie.

Literatura:

- Enabling Wireless Sensors with IEEE 802.15.4, Low-Rate Wireless Personal Area Networks - by Jose A. Gutierrez, Edgar H. Callaway, Jr., and Raymond L. Barrett, Jr., IEEE Press, 2003
- Wireless Sensor Networks - Architectures and Protocols, by Edgar H. Callaway, Jr., Auerbach Publications, 2003
- www.zigbee.org

Při obhajobě semestrální části diplomového projektu je požadováno:

- Body 1 - 3.

Podrobné závazné pokyny pro vypracování diplomové práce naleznete na adrese http://www.fit.vutbr.cz/info/szz/

Technická zpráva diplomové práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap, které byly vyřešeny v rámci ročníkového a semestrálního projektu (30 až 40% celkového rozsahu technické zprávy).

Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním paměťovém médiu (disketa, CD-ROM), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Očenášek Pavel, Ing.**, UIFS FIT VUT
Datum zadání: 28. února 2006
Datum odevzdání: 22. května 2007

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav informačních systémů
612 66 Brno, Božetěchova 2

L.S.

doc. Ing. Jaroslav Zendulka, CSc.
vedoucí ústavu

LICENČNÍ SMLOUVA
POSKYTOVANÁ K VÝKONU PRÁVA UŽÍT ŠKOLNÍ DÍLO

uzavřená mezi smluvními stranami

1. **Pan**

Jméno a příjmení: **Bc. Jan Nagy**
Id studenta: 47061
Bytem: Bořetická 6, 628 00 Brno
Narozen: 12. 08. 1982, Chomutov
(dále jen "autor")

a

2. **Vysoké učení technické v Brně**

Fakulta informačních technologií
se sídlem Božetěchova 2/1, 612 66 Brno, IČO 00216305
jejímž jménem jedná na základě písemného pověření děkanem fakulty:

.....
(dále jen "nabyvatel")

Článek 1
Specifikace školního díla

1. Předmětem této smlouvy je vysokoškolská kvalifikační práce (VŠKP):
diplomová práce

Název VŠKP: Zabezpečení bezdrátových senzorových sítí
Vedoucí/školitel VŠKP: Očenášek Pavel, Ing.
Ústav: Ústav informačních systémů
Datum obhajoby VŠKP: .....

VŠKP odevzdal autor nabyvateli v:
tištěné formě počet exemplářů: 1
elektronické formě počet exemplářů: 2 (1 ve skladu dokumentů, 1 na CD)

2. Autor prohlašuje, že vytvořil samostatnou vlastní tvůrčí činností dílo shora popsané a specifikované. Autor dále prohlašuje, že při zpracovávání díla se sám nedostal do rozporu s autorským zákonem a předpisy souvisejícími a že je dílo dílem původním.
3. Dílo je chráněno jako dílo dle autorského zákona v platném znění.
4. Autor potvrzuje, že listinná a elektronická verze díla je identická.

Článek 2
Udělení licenčního oprávnění

1. Autor touto smlouvou poskytuje nabyvateli oprávnění (licenci) k výkonu práva uvedené dílo nevýdělečně užít, archivovat a zpřístupnit ke studijním, výukovým a výzkumným účelům včetně pořizování výpisů, opisů a rozmnoženin.
2. Licence je poskytována celosvětově, pro celou dobu trvání autorských a majetkových práv k dílu.
3. Autor souhlasí se zveřejněním díla v databázi přístupné v mezinárodní síti:
   - ☐ ihned po uzavření této smlouvy
   - ☐ 1 rok po uzavření této smlouvy
   - ☐ 3 roky po uzavření této smlouvy
   - ☐ 5 let po uzavření této smlouvy
   - ☐ 10 let po uzavření této smlouvy

   (z důvodu utajení v něm obsažených informací)
4. Nevýdělečné zveřejňování díla nabyvatelem v souladu s ustanovením § 47b zákona č. 111/ 1998 Sb., v platném znění, nevyžaduje licenci a nabyvatel je k němu povinen a oprávněn ze zákona.

Článek 3
Závěrečná ustanovení

1. Smlouva je sepsána ve třech vyhotoveních s platností originálu, přičemž po jednom vyhotovení obdrží autor a nabyvatel, další vyhotovení je vloženo do VŠKP.
2. Vztahy mezi smluvními stranami vzniklé a neupravené touto smlouvou se řídí autorským zákonem, občanským zákoníkem, vysokoškolským zákonem, zákonem o archivnictví, v platném znění a popř. dalšími právními předpisy.
3. Licenční smlouva byla uzavřena na základě svobodné a pravé vůle smluvních stran, s plným porozuměním jejímu textu i důsledkům, nikoliv v tísni a za nápadně nevýhodných podmínek.
4. Licenční smlouva nabývá platnosti a účinnosti dnem jejího podpisu oběma smluvními stranami.

V Brně dne: .....

....

Nabyvatel Autor

Abstrakt

Tato diplomová práce se zabývá problematikou zabezpečení bezdrátových senzorových sítí, konkrétně průmyslového standardu ZigBee. Cílem práce je seznámit se se standardem 802.15.4 a technologií ZigBee, seznámit se s existujícími metodami zabezpečení v této oblasti a analyzovat požadavky na zabezpečení uvedené technologie. Dalším výstupem práce je představení ZigBee kitu a popis a analýza implementace ZigBee protokolu od firmy Microchip, která je spojena s praktickým odzkoušením bezpečnostních funkcí v laboratoři ZigBee.

Klíčová slova

Bezdrátová síť, senzorová síť, IEEE 802.15.4, WPAN Low Rate, ZigBee, Microchip ZigBee stack, zabezpečení, bezpečnostní režimy, trust center, koordinátor, RFD, bezpečnostní klíč, CC2420, MRF24J40, ZENA analyzátor

Abstract

This thesis deals with the security of wireless sensor networks, mainly of the industrial standard ZigBee. The aim of the work is to familiarize with the 802.15.4 standard and the ZigBee technology, especially with present methods of security in this field. I have also analysed the requirements for the security of this technology. Further aim of this work is the introduction of the ZigBee kit and description of the Microchip's ZigBee stack. Analysis of the stack is connected with practical test of security functions in the ZigBee laboratory.

Keywords

Wireless network, sensor network, IEEE 802.15.4, WPAN Low Rate, ZigBee, Microchip ZigBee stack, security, security suites, trust center, coordinator, RFD, security key, CC2420, MRF24J40, ZENA analyzer

Citace

Jan Nagy: Zabezpečení bezdrátových senzorových sítí, diplomová práce, Brno, FIT VUT v Brně, 2007

Zabezpečení bezdrátových senzorových sítí

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením Ing. Pavla Očenáška.
Další informace mi poskytli Mgr. Roman Trchalík a Ing. Václav Šimek.
Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

……
Jan Nagy
18.5.2007

Obsah

1 Úvod

Bezdrátové komunikační technologie představují jednu z rychle se rozvíjejících oblastí telekomunikační technologie. Vznik bezdrátové komunikace sahá do období přelomu 19. a 20. století, kdy různí vědci prováděli pokusy s rádiovým přenosem. Aplikací technologie, která dovoluje využít rádiová pásma velmi vysokých frekvencí (stovky MHz až jednotky GHz), dochází k rozvoji bezdrátových komunikací a bezdrátových sítí.

Dnes existuje několik norem pro bezdrátovou komunikaci na krátkou vzdálenost pod souhrnným názvem *WPAN* (*Wireless Personal Area Network*). Tyto normy byly schváleny standardizační organizací IEEE pod souhrnným označením IEEE 802.15. Mezi schválené specifikace patří například Bluetooth (802.15.1), WPAN High Rate (vysokorychlostní malé sítě pro komunikaci na krátkou vzdálenost, 802.15.3) nebo WPAN Low Rate (nízkorychlostní malé sítě pro komunikaci na krátkou vzdálenost, 802.15.4).

Pomalé malé sítě (WPAN Low Rate) jsou často také nazývané senzorové sítě. Tyto sítě jsou určené pro malá a jednoduchá bezdrátová zařízení v průmyslu i v domácnosti a pracují v bezlicenčních rádiových pásmech. Až donedávna fungovaly monitorovací a řídící systémy výhradně na bázi přenosu dat z pevně umístěných zařízení (senzorů) spojených drátovým vedením. Pokračující vývoj v oblasti integrovaných čipů s malým příkonem přinesl zařízení, která integrují měřící a výpočetní schopnosti a bezdrátovou komunikaci. Nově vznikající aplikace zahrnují průmyslové a výrobní monitorování a údržbu nebo složitý sběr informací pro vnitřní i venkovní prostředí, včetně domovů, kanceláří, továren, skladů nebo zemědělské půdy.

V první části této diplomové práce se budu věnovat obecnému konceptu bezdrátových senzorových sítí a představím jeden ze standardů senzorových sítí – ZigBee. Popíši strukturu tohoto standardu, zmíním se o jeho jednotlivých vrstvách, typech zařízení v síti a topologii senzorové sítě ZigBee. V druhé části práce se budu zabývat problematikou zabezpečení senzorových sítí s důrazem na zabezpečení standardu ZigBee. Praktickou část budu provádět s implementací ZigBee protokolu od firmy Microchip. Na hardwarových modulech budu demonstrovat postup zabezpečení dané aplikace.

2 Senzorové sítě

Bezdrátová senzorová síť (*Wireless Sensor Network*) [3] je bezdrátová síť sestávající z prostorově distribuovaných autonomních zařízení používající senzory ke kooperativnímu monitorování fyzikálních nebo přírodních podmínek (jako třeba teplota, zvuk, vibrace, tlak, pohyb) v různých lokacích. Vývoj bezdrátových senzorových sítí byl původně motivován vojenskými aplikacemi (např. průzkum bojiště), přesto jsou dnes bezdrátové senzorové sítě používány v mnoha oblastech civilních aplikací, například monitorování prostředí, diagnostika zařízení, zdravotnické aplikace, domácí automatizace nebo řízení dopravy.

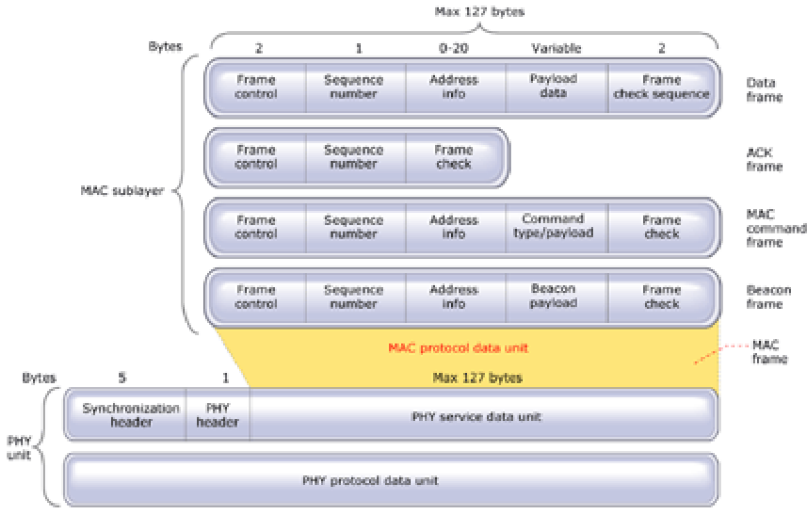
Každý uzel v senzorové síti je typicky vybaven rádiovým vysílačem s přijímačem nebo jiným bezdrátovým komunikačním zařízením, malým mikrokontrolerem a zdrojem energie (většinou baterií).

Velikost jednoho uzlu senzorové sítě se může pohybovat od velikosti krabice od bot po zařízení velikosti smítka prachu. Cena jednotlivých uzlů senzorové sítě je podobně variabilní, pohybuje se od několika tisíc korun po pár korun v závislosti na velikosti senzorové sítě a složitosti jednotlivých uzlů. Omezení vyplývající z velikosti a ceny uzlů senzorové sítě vyúsťují v odpovídající omezení týkající se zdrojů jako energie, paměť, výpočetní rychlost a šířka komunikačního pásma.

2.1 Technologie senzorových sítí

Základem pro sítě s malou propustností a malými nároky na napájení (tedy senzorové sítě) je norma IEEE 802.15.4. Tato norma definuje specifikaci rádiového přenosu (fyzickou vrstvu) a podvrstvu MAC linkové vrstvy pro spolehlivý bezdrátový přenos. Pro zvýšení spolehlivosti přenášených dat je vysílání na fyzické vrstvě prováděno technologií rozprostřeného spektra DSSS (*Direct Sequence Spread Spectrum*). Jednotlivé bity jsou nahrazeny početnější sekvencí bitů (čipů), které se pak vysílají. Signál je tak rozprostřen do větší části spektra a je více odolný vůči rušení. Uživatelům, kteří neznají mechanismus vytváření pseudonáhodné sekvence (čipovací sekvence), se přenášená data jeví jako šum.

Komunikační protokol linkové vrstvy definovaný standardem 802.15.4 je založen na přenosu datových rámců. Ve standardu 802.15.4 jsou definovány čtyři základní typy komunikačních rámců na linkové vrstvě pro přístup k médiu (obrázek 1): datový rámec, potvrzení ACK, příkaz MAC a *beacon* rámec.



Obrázek 1. Čtyři základní typy rámců definovaných v 802.15.4 – datový, ACK, MAC command a beacon. (Zdroj [5])

Datový rámec (*data frame*) poskytuje prostor užitečná data. Rámce jsou číslované, aby bylo možné dohledat případné chybějící rámce. Na konci rámce je kontrolní součet, pomocí kterého si příjemce zjistí bezchybnost doručení.

Rámec pro potvrzení (*acknowledgment frame*, ACK) poskytuje odesílateli zpětnou vazbu od příjemce, který tímto rámcem potvrzuje bezchybnost doručení dat. Příjemce využívá krátkého času mezi vysílanými pakety, aby potvrdil správnost právě přijatého paketu. Tento rámec lze využít pouze na vrstvě MAC pro potvrzovanou komunikaci a většinou není zabezpečen.

Rámec *MAC command* poskytuje mechanismus pro vzdálené řízení konfigurace zařízení. Centralizovaná řídicí jednotka sítě používá MAC ke konfiguraci jednotlivých klientů.

Rámec *beacon* slouží pro synchronizaci zařízení v síti, aniž by musela poslouchat po celou dobu vysílání, čímž se šetří energie. Rámec může probouzet uspané klienty, kteří jen poslouchali na síti, zda jim nepřišla zpráva (v *beacon-enabled* síti, viz kapitolu 3.3 – Topologie sítě).

Nad fyzickou a linkovou vrstvou, které definuje norma IEEE 802.15.4, jsou vrstvy síťová a aplikační, které jsou definovány jednotlivými standardy senzorových sítí a výrobci. Tyto vrstvy budují síť a zajišťují provoz zařízení. Síťová vrstva může implementovat různá schémata jako ZigBee (standard ZigBee), Wibree (plánovaný standard Wibree), IP verze 6 (standard 6lowpan) a množství proprietárních schémat. Aplikační software se také liší, ale obecně je to malý program, který umožní fungování senzoru, jeho monitorování a reakce na externí události a přijaté příkazy.

3 Standard ZigBee

Někdy se pomalé bezdrátové sítě s malým dosahem označují jménem průmyslové specifikace ZigBee. Specifikace ZigBee dostala své jméno podle polétavého, kmitavého pohybu včel, kterým samostatné jednoduché organizmy přistupují k řešení složitých úkolů. Včela je například schopná sdílet s ostatními členy kolonie informace o umístění, vzdálenosti a směru nově nalezeného zdroje potravy. Tento přístup chtějí tvůrci specifikace emulovat. ZigBee reaguje na problémy v síti. Pokud vyslaný rámec není příjemcem potvrzen, musí se opětovně vyslat. Pokud ani po několika pokusech není trámec přenesen úspěšně, např. při problému v síti nebo při zhoršení kvality bezdrátového spoje mezi vysílačem a přijímačem, ZigBee umožňuje autonomní nalezení a využití alternativních cest v síti (pokud existují). Směrovací protokol umožňuje realizaci více skoků v síti na cestě od zdroje k příjemci zprávy.

Aby byly u jednotlivých zařízení splněny všechny požadavky standardu, vznikla v roce 2002 ZigBee Alliance (www.zigbee.org), pracovní průmyslová skupina, která sdružuje přes dvě stě nadnárodních firem a společností (Philips, Samsung, Motorola, Honeywell, Cisco Systems, Freescale Semiconductors ad.). Aliance ZigBee vyvíjí standardizovaný aplikační software nad nejvyšší vrstvou bezdrátového standardu pro pomalé malé sítě 802.15.4. Aliance také spolupracuje s mezinárodní standardizační organizací IEEE, aby byla dosažena maximální kompatibilita mezi ZigBee zařízeními v rámci standardu 802.15.4.

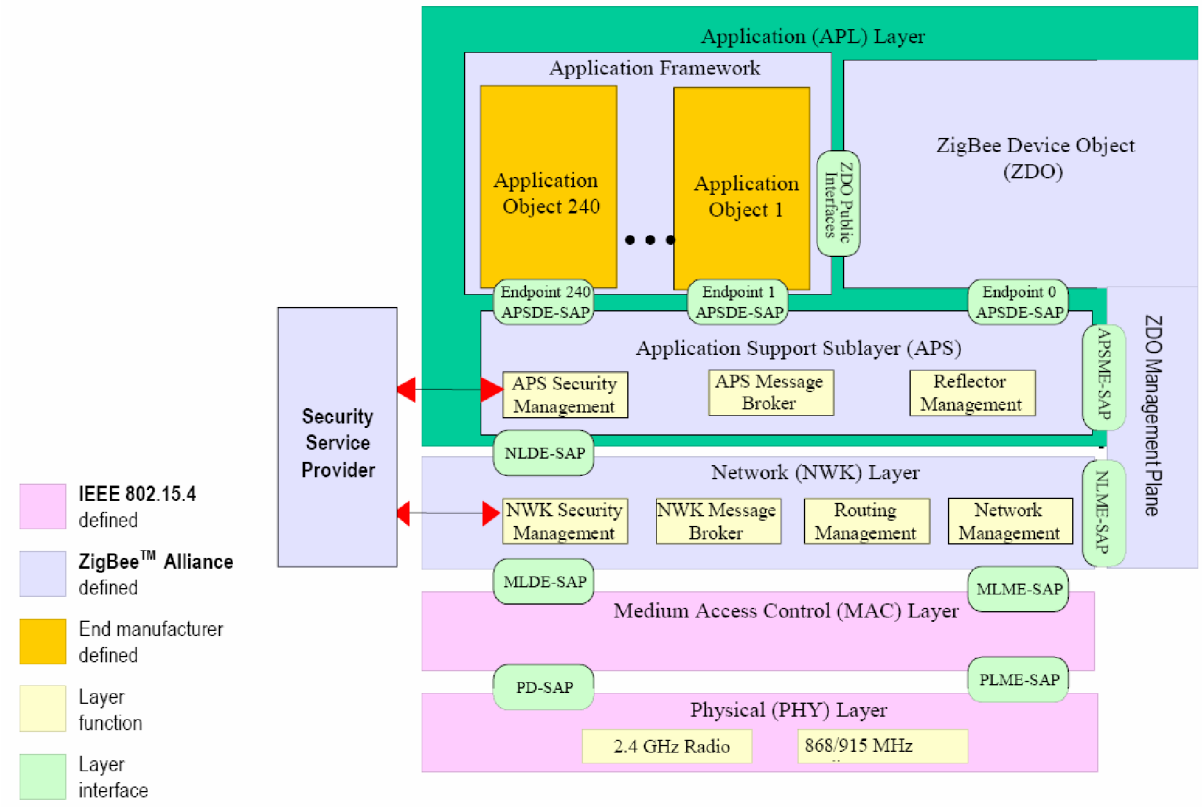
3.1 Struktura protokolu ZigBee

Architektura ZigBee sestává z množiny bloků zvaných vrstvy. Každá vrstva provádí specifickou množinu služeb pro vrstvu vyšší. Každá entita poskytující nějakou službu poskytuje rozhraní vyšší vrstvě přes tzv. přístupový bod služeb (*service access point, SAP*) a každý SAP podporuje množinu služeb, které nabízí požadovanou funkcionalitu.

Základem pro senzorové sítě je norma IEEE 802.15.4. Tato norma definuje specifikaci rádiového přenosu a podvrstvy MAC linkové vrstvy pro spolehlivý bezdrátový přenos. ZigBee pak nad těmito vrstvami definuje vyšší vrstvy, od síťové po aplikační. Doplňuje specifikaci sítě zejména o směrování a bezpečnost (autentizace, management klíčů). Nad samotnou síťovou architekturou pak ZigBee specifikuje profily pro vzájemnou spolupráci bezdrátových zařízení od různých výrobců.

Norma 802.15.4 nenabízí mechanismus pro distribuci šifrovacích klíčů, tuto službu musí zajišťovat ZigBee. ZigBee nabízí takové prostředky managementu klíčů, že lze síť bezpečně spravovat i na dálku.

Architektura ZigBee, jak je zobrazena na obrázku 2, je založena na standardním sedmivrstvém modelu Open Systems Interconnetion (OSI), ale definuje jen vrstvy relevantní pro funkcionalitu senzorové sítě.



Obrázek 2. Architektura vrstev protokolu ZigBee. (Zdroj [7])

3.1.1 Specifikace fyzické a linkové vrstvy standardu 802.15.4

První dvě vrstvy, fyzická (PHY) a linková pro přístup k médiu (MAC), jsou definovány standardem 802.15.4. Pracovní skupina IEEE schválila první koncept vrstev PHY a MAC v roce 2003 ve standardu 802.15.4-2003. Další rozšíření bylo schváleno v červnu roku 2006 ve standardu 802.15.4-2006. Tento modernizovaný standard zahrnuje specifické rozšíření a objasnění původního standardu 802.15.4-2003. Řeší některé nejasnosti, redukuje nadbytečné složitosti, zvyšuje flexibilitu v používání bezpečnostních klíčů a další.

Úkolem fyzické vrstvy je vysílání a příjem datových jednotek. Komunikace probíhá na jednom ze tří bezlicenčních radiových pásem ISM (*Industrial, Scientific, Medical*): 868/915 MHz (Evropa/Severní Amerika, Austrálie) a 2400 MHz (celosvětově).

Vrstva MAC provádí synchronizaci, zabezpečuje přístup na radiový kanál, ověřuje platnost rámce, potvrzuje příjem rámce, řídí spojení, generování a rozpoznání adres.

3.1.2 Specifikace vyšších vrstev protokolu ZigBee

Vyšší vrstvy protokolu ZigBee jsou definovány aliancí ZigBee. Aliance ZigBee staví na základních vrstvách (fyzické a linkové pro přístup k médiu) přidáním dalších vrstev – síťové (NWK) a aplikační (APL).

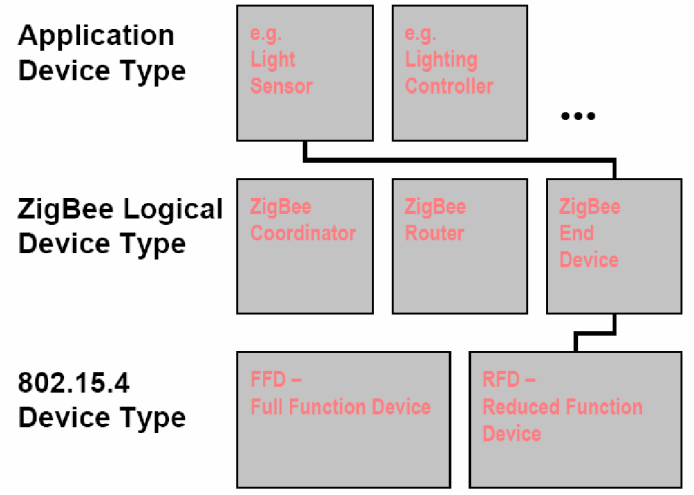
Mezi úkoly síťové vrstvy (NWK) patří zabezpečení rámců a jejich směrování k cílovým uzlům. Hledá přímé (tedy dostupné jedním přeskokem, tzv. *one-hop*) sousední uzly a ukládá si informace o nich. Síťová vrstva koordinátoru ZigBee (viz následující kapitolu) zajišťuje komunikaci a přiděluje

adresy novým zařízením. Síťová vrstva poskytuje funkcionalitu k zajištění správného fungování IEEE 802.15.4 MAC vrstvy a poskytuje vhodné rozhraní služeb aplikační vrstvě. Ke správné spolupráci síťové a aplikační vrstvy se síťová vrstva skládá ze dvou entit nabízejících potřebnou funkcionalitu. Tyto entity poskytují data a management. Datová entita - NWK layer data entity (NLDE) - poskytuje přenos dat prostřednictvím přístupového bodu služeb (SAP), NLDE-SAP, a entita managementu - NWK layer management entity (NLME) - poskytuje služby managementu přes příslušný SAP, NLME-SAP. NLME využívá NLDE k dosažení některých úkolů managementu a také spravuje databázi spravovaných objektů známých jako network information base (NIB).

Aplikační vrstva sestává z pomocné aplikační podvrstvy (*application support sub-layer,* APS), z objektů ZigBee (*ZigBee device object,* ZDO) a z aplikačních objektů definovaných výrobci koncových zařízení. Úkolem pomocné aplikační podvrstvy je udržovat vazební (*binding*) tabulky, které umožňují propojit dvě zařízení, a přeposílat zprávy mezi vzájemně vázanými zařízeními. Objekt ZigBee (ZDO) definuje roli zařízení v síti (např. ZigBee koordinátor nebo koncové zařízení), navazuje spojení a odpovídá na žádosti spojení a zřizuje zabezpečené spojení mezi zařízeními sítě. ZDO také zajišťuje hledání zařízení v síti a zjišťuje rozsah poskytovaných služeb.

3.2 Typy zařízení

V síti ZigBee rozlišujeme několik typů zařízení (viz obrázek 3).



Obrázek 3. Typy zařízení v ZigBee. (Zdroj [19])

V závislosti na různých úrovních pohledu můžeme rozlišit zařízení definovaná na základě normy IEEE 802.15.4, logická zařízení na základě standardu ZigBee a typy zařízení podle aplikace.

Na základě normy 802.15.4 rozlišujeme tyto hardwarové platformy ZigBee zařízení:

- Plně funkční zařízení (FFD, *Full Function Device*) podporuje všechny funkce a doplňky definované standardem 802.15.4. Tento typ zařízení může v síti ZigBee působit jako:
    - Síťový koordinátor (*ZigBee Coordinator*) - zná topologii sítě. Je to nejvíce sofistikované zařízení ze tří logických typů zařízení (koordinátor, směrovač a koncové zařízení) a vyžaduje nejvíce paměti a výpočetního výkonu. Obvykle plní také funkci tzv. trust center zařízení (viz kapitolu 4.1.3 – Bezpečnostní klíče).

- Síťový směrovač (*ZigBee Router)* – FFD s přídavnou pamětí a výpočetní silou. Směrovač předává data od ostatních zařízení.

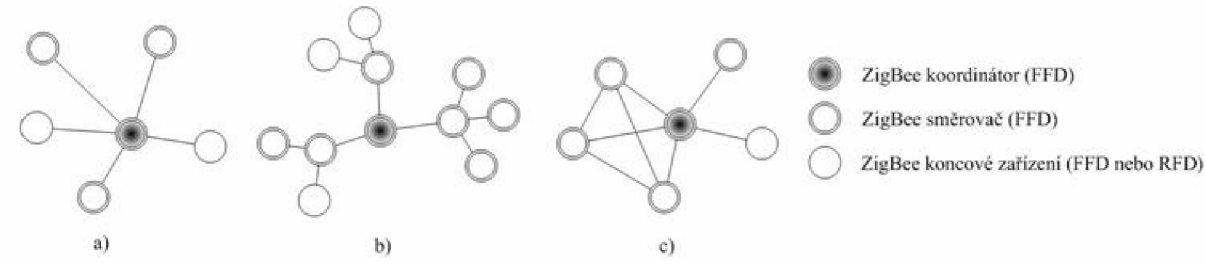
- Zařízení s redukovanou funkčností (RFD, *Reduced Function Device*) má implementovány pouze některé funkce, aby se snížila cena a složitost zařízení. V síti ZigBee se tomuto zařízení říká:
  - Koncové zařízení (ZED, *ZigBee End Device)*. Takové zařízení může pouze komunikovat s nadřazeným zařízením (buď s koordinátorem nebo směrovačem); neumí přeposílat data od ostatních zařízení.

Další dělení zařízení podle aplikační funkčnosti může být například na senzor světla, ovladač osvětlení apod.

3.3 Topologie sítě

Pro adresaci jednotlivých zařízení v síti lze použít dva druhy adres. Každé zařízení má unikátní 64-bitovou IEEE adresu (*extended address*). Po připojení k síti může koordinátor přidělit koncovému zařízení 16-bitovou zkrácenou adresu, která minimalizuje komunikační režii.

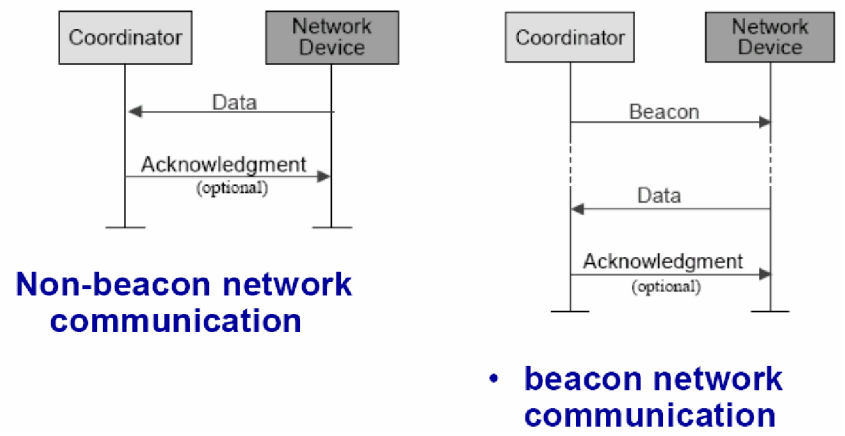
Každou síť lze jednoznačně určit pomocí 16-bitového identifikátoru PAN ID (*Personal Area Network Identificator*), který se používá v případě, kdy je v jednom prostoru provozováno více sítí podle standardu IEEE 802.15.4. Každá síť s jedinečným PAN ID je řízena síťovým koordinátorem (centrální stanicí). Síťová vrstva standardu ZigBee podporuje síťové topologie typu hvězda (*star*), strom (*tree*) a síť (*mesh*) (viz obrázek 4).



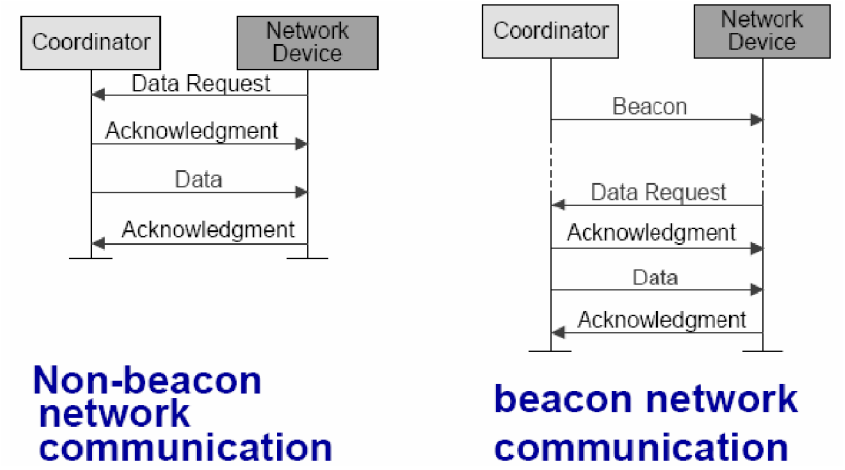
Obrázek 4. Topologie sítě ZigBee typu a) hvězda, b) strom, c) síť (mesh). (Zdroj [2])

Uzly sítě jsou buď plně funkční zařízení (FFD), která mohou vykonávat funkce koordinátora, směrovače, nebo koncového zařízení, a nebo redukovaná zařízení (RFD), která mohou pracovat pouze jako koncová zařízení. V topologii typu hvězda komunikují ostatní zařízení, označovaná jako koncová, přímo s koordinátorem. V topologii typu síť a strom koordinátor spouští komunikaci a stanovuje parametry sítě. Síť lze rozšířit použitím ZigBee směrovačů. [2]

Dále rozlišujeme dva typy sítí, a to *beacon-enabled* a *non-beacon* síť.



Obrázek 5a. Přenos dat k síťovému koordinátoru. (Zdroj [20])



Obrázek 5b. Přenos dat od síťového koordinátora. (Zdroj [20])

V *beacon-enabled* síti koordinátor pravidelně vysílá signál *beacon* (obrázky 5a, 5b), který koncová zařízení využívají k připojení se k síti a vlastní synchronizaci pro následný přenos dat (obrázek 5a) nebo žádost o data (*data request*) (obrázek 5b). V *non-beacon* síti koordinátor také periodicky vysílá signál, který však slouží pouze k jeho vlastní identifikaci a koncovým zařízením k detekci. Koncová zařízení komunikují s koordinátorem přímo pomocí zasílání dat (obrázek 5a) nebo pomocí požadavku na data a potvrzovacích rámců (obrázek 5b).

3.3.1 Síťová asociace

Nová ZigBee síť je sestavována koordinátorem. Ten po startu hledá další koordinátory pracující na povolených kanálech. Na základě množství nalezených sítí na jednotlivých kanálech sestaví vlastní síť a vybere unikátní 16-bitové číslo PAN ID. Po sestavení sítě se mohou routery a koncová zařízení připojovat k síti [22].

4 Zabezpečení senzorových sítí

Se schopnostmi a možnostmi moderních senzorových sítí jde ruku v ruce riziko zneužití sítě. Nebezpečí však nehrozí jen v podobě cílených útoků na senzorovou síť, ale často také v podobě nesprávné funkčnosti zařízení v důsledku kolize v síti. Takové události mohou vzbuzovat dojem, že nové technologie jsou nespolehlivé nebo špatně navržené: dálkové ovladače řídící špatná zařízení, hlásiče požáru, které se spustí bezdůvodně, senzory prostředí, které vykazují špatná nebo žádná data apod. Bezpečnostní metody samy o sobě nemohou podobným událostem zabránit, ale mohou omezit jejich výskyt. Pro nově vznikající bezdrátové sítě senzorů a řídících zařízení představuje jakýkoli nevyužitý či ztrátový čas v důsledku útoku na síť selhání zabezpečení sítě, jehož důsledek je pak snížení účinnosti aplikace a růst nákladů.

Abychom předešli těmto problémům, musí být senzorová síť zabezpečena proti neautorizovanému použití (náhodnému nebo zlomyslnému) - to vyžaduje autentizaci a řízení přístupu a proti odposlechu dat – to vyžaduje šifrování komunikace.

Dnes jsou tyto činnosti nejčastěji zajišťovány pomocí centrálně řízené výměny klíčů: centrální kontroler ověřuje identifikaci uzlů a distribuuje jim klíče, aby sestavil zabezpečená spojení. Po autentizaci může následovat šifrování, například pomocí algoritmu AES.

Takový přístup je vhodný pro malé a soběstačné sítě, ale nedovoluje snadné rozšiřování sítě. Vhodná analogie k centrálnímu kontroleru je dopravní policista: také musí být schopen řídit rušnou křižovatku, ale s narůstajícím počtem jízdních pruhů, ulic nebo počtu aut se mu může řízení vymknout z ruky [25].

S modelem centrálně řízené výměny klíčů jsou spojeny některé praktické výzvy, např. vytvoření spojení mezi sítěmi – protože uzly obecně nemají takovou inteligenci, aby věděly, s kterým kontrolerem mají komunikovat (poslouchat jeho příkazy). Centralizované řízení navíc vytváří centrální bod náchylný na selhání zabezpečení sítě.

V rozsáhlých sítích a kritických aplikacích je nutné implementovat technologii veřejných klíčů, která může být použita k jednoznačné identifikaci uzlu v síti a pak k bezpečnému přenosu dat z/do takového uzlu.

Flexibilní decentralizovaná síťová architektura (*mesh*) (viz kapitolu 3.3 – Topologie sítě) umožňuje růst senzorové sítě a snadnou spolupráci mezi uzly. V takové architektuře umožňuje autentizace a bezpečnost založená na veřejných klíčích uzlům pracovat nezávisle a kooperativně. Každé zařízení je vybaveno svými bezpečnostními klíči a bezpečnostní politikou. Identity a politiky mohou být vytvářeny centrálně a poté distribuovány jednotlivým uzlům, aby byla umožněna jejich spolupráce v síti.

S technologií veřejného klíče se pomocí prvního klíče, které zná pouze zařízení, spáruje zařízení se svou identitou v síti; a druhý klíč, který je matematicky vypočítán z prvního, použije síť pro ověření této identity. To umožňuje rychlou a spolehlivou identifikaci zařízení.

Veřejné klíče lze použít pro různé účely: k jednoznačné identifikaci osoby nebo zařízení, pro výměnu klíčů přes síť, umožňující dvěma zařízením bezpečně komunikovat, nebo pro vytvoření digitálního podpisu, který může být použit k ověření integrity zprávy.

Ne všechny senzorové aplikace vyžadují zabezpečení. Přesto tato zařízení pracují nezávisle a jsou náchylná na chyby v použití, zvláště při použití bezdrátové komunikace. Pro většinu aplikací, a obzvláště pro bezdrátová zařízení, kde je větší riziko útoku, je však bezpečnost vyžadována.

Cíle bezpečnostního návrhu senzorových sítí kopírují atributy těchto sítí. Jednotlivá zařízení (uzly senzorové sítě) jsou založena na malých mikrokontrolerech, což znamená, že šifrování musí být snadno implementovatelné. Zařízení v senzorové síti mají omezenou paměť, takže potřebujeme malou režii pro uložení klíčů. Pro senzorová zařízení se předpokládá nasazení v domácnostech a průmyslu, z čehož plyne, že některé aplikace budou vyžadovat dostatečnou pružnost pro odpověď (tzn. malou latenci, třeba v případě řídících systémů).

4.1 Bezpečnost v ZigBee

Bezpečnost a integrita dat jsou klíčové výhody technologie ZigBee. ZigBee využívá bezpečnostního modelu linkové podvrstvy přístupu k médiu MAC standardu 802.15.4. Tento standard definuje bezpečnost na úrovni linkové vrstvy a specifikuje čtyři bezpečnostní služby:

- Řízení přístupu, autentizace (*access control*) – zařízení si udržuje seznam důvěryhodných zařízení v síti
- Šifrování dat (*data encryption*), které používá symetrický 128-bitový klíč standardu AES
- Integrita rámce (*frame integrity*), která chrání rámce před modifikací třetí stranou
- Sekvenční posloupnost dat (*sequential freshness*), kdy řídicí prvek v síti odmítne data, která byla opakovaně poslána, aniž by byla změněna hodnota *freshness* v rámci. Tato služba je volitelná.

Později popíšu tyto základní bezpečnostní služby podrobněji.

ZigBee rozlišuje celkem osm úrovní zabezpečení (*security suite*) dostupných vrstvám MAC, NWK a APS (viz obrázek 6). Úrovně zabezpečení určují, zda existuje řízení přístupu, zda se šifrují data, délku integritního kódu rámce (*Message Integrity Code*, MIC)[1] a zda se hlídá sekvenční posloupnost dat.

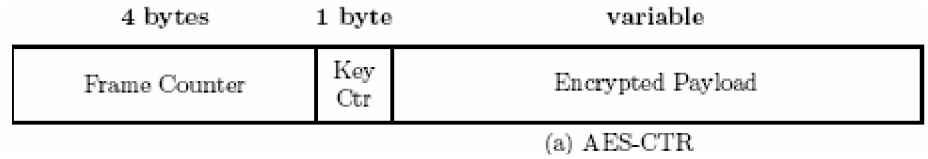
Security suite name	Access control	Data encryption	Frame integrity	Sequential freshness (optional)
None				
AES-CTR	x	x		x
AES-CCM-128	x	x	x	x
AES-CCM-64	x	x	x	x
AES-CCM-32	x	x	x	x
AES-CBC-MAC-128	x		x	
AES-CBC-MAC-64	x		x	
AES-CBC-MAC-32	x		x	

Obrázek 6. Bezpečnostní úrovně v protokolu ZigBee. (Zdroj [15])

[1] Kryptografická konvence hovoří o integritním součtu jako o MAC (*Message Authentication Code*), standard 802.15.4 však kvůli odlišení od vrstvy pro přístup k médiu (MAC, media access control) používá označení MIC [23]. V následujících obrázcích bude místo MIC uváděno MAC a myšlen bude integritní součet rámce.

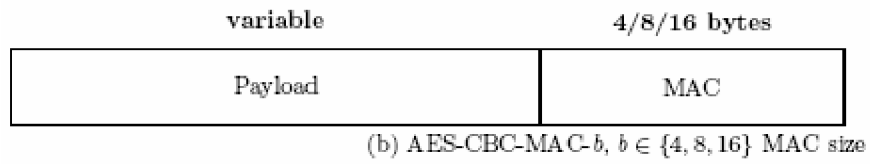
Nejjednodušší bezpečnostní úroveň je **None**. Všechny rádiové čipy ji musí implementovat. Nepracuje s žádným zabezpečením a chová se jako funkce identity. Neposkytuje žádné bezpečnostní záruky.

Další bezpečnostní úrovní je **AES-CTR**. Ta poskytuje důvěrnost dat pomocí blokové šifry AES v čítačovém režimu (counter mode). K zašifrování dat v tomto režimu odesílatel rozdělí text na 16-bajtové bloky a provede operaci xor (logický výlučný součet) s měnícím se čítačem. Příjemce pak také provede operaci xor na přijatá data a čítač, čímž dostane původní zprávu. Čítač se zde nazývá inicializační vektor (IV) a sestává adresy odesílatele, čítače rámce (který identifikuje paket), čítače klíče a čítače bloku. Čítač rámce je spravován hardwarem rádiového čipu. Odesílatel jej inkrementuje po zašifrování každého paketu. Po dosažení maximální hodnoty vrátí čip chybu a další šifrování již není možné. Čítač klíče je kontrolován aplikací. Může být inkrementován i když čítač rámce dosáhl svého maxima. Požadavkem je, aby se IV neopakoval během života konkrétního klíče a role obou čítačů (rámce a klíče) je zabránit znovupoužití IV. Čítač bloku zaručuje, že každý blok použije jinou hodnotu IV. Hodnota tohoto čítače může být vypočítána, takže ji odesílatel nemusí posílat. Výsledný paket pak ukazuje obrázek 7.



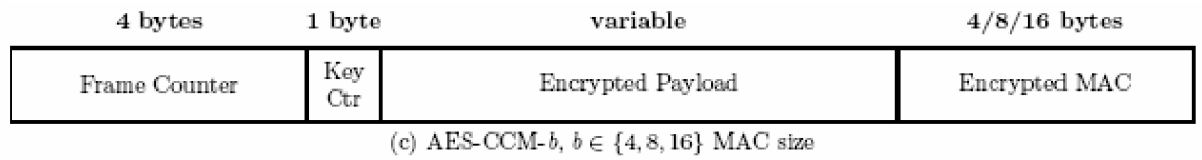
Obr. 7. Formátování paketu pro bezpečnostní režim AES-CTR. (Zdroj [23])

Bezpečnostní úroveň **AES-CBC-MAC** poskytuje datovou integritu pomocí CBC-MAC. Odesílatel může spočítat 4, 8 nebo 16-bajtový MIC, což vede na tři různé AES-CBC-MAC varianty (viz Obrázek 6). MIC může být spočítán stranami sdílejícími symetrický (stejný) klíč. MIC chrání jak hlavičku paketu, tak data. Odesílatel přidá k datům MIC (viz obrázek 8) a příjemce kontrolní součet MIC zkontroluje (tak, že si jej vypočítá a porovná s přijatým).



Obr. 8. Formátování paketu pro bezpečnostní režim AES-CBC-MAC. (Zdroj [23])

Poslední bezpečnostní úroveň **AES-CCM** se používá pro šifrování a autentizaci. Nejprve aplikuje integritní ochranu nad hlavičkou a daty spočítáním CBC-MAC a poté zašifruje data a spočítaný MIC pomocí AES-CTR režimu. Režim AES-CCM obsahuje pole z obou autentizačních a šifrovacích operací: MIC, čítač rámce (frame counter) a čítač klíče (key counter). Tak jako režim AES-CBC-MAC má tři varianty v závislosti na velikosti MIC, také AES-CCM může mít tři varianty. Formát paketu ilustruje obrázek 9.



Obr. 9. Formátování paketu pro bezpečnostní režim AES-CCM. (Zdroj [23])

Příjemce může volitelně umožnit ochranu proti znovuposlání již odeslaných rámců (replay protection, sequential freshness) použitím bezpečnostní úrovně, která poskytuje důvěrnost dat (provádí jejich šifrování). Sem patří AES-CTR a všechny varianty AES-CCM. Příjemce použije čítače rámce a klíče jako 5-bajtovou hodnotu „čítače opakování“ (replay counter), kde čítač klíče bude představovat nejvýznamnější bajt této hodnoty. Příjemce porovná hodnotu čítače opakování z příchozího paketu s dosud nejvyšší hodnotou tohoto čítače, kterou si zaznamenává jako položku v přístupovém seznamu ACL. O tomto seznamu se zmíním v zápětí. Pokud má příchozí paket větší hodnotu čítače opakování než je uložená hodnota, pak je paket přijat a nová hodnota je uložena. Pokud však má příchozí paket menší hodnotu, je odmítnut a aplikace je o tom zpravena. Tomuto čítači se říká čítač odmítnutí, ačkoliv je to stejný čítač jako inicializační vektor. Pouze slouží k jinému úkolu. IV má za úkol chránit důvěrnost. Čítač odmítnutí není veřejně dostupný aplikaci k použití.

Řízení přístupu je podporováno všemi bezpečnostními úrovněmi vyjma *None* a dává zařízení možnost vybrat si jiné zařízení, se kterým chce komunikovat. Každé zařízení si uchovává tzv. přístupový seznam (*Access Control List*, ACL) ve struktuře zvané *MAC sublayer PAN Information Base* (MPIB), který lze využít pro bezpečnostní účely. Přístupový seznam (ACL) může obsahovat až 255 záznamů, pro každé cílové zařízení (se kterým uzel komunikuje) jeden záznam. Každý záznam v ACL sestává z adresy cíle (IEEE adresa nebo volitelná zkrácená adresa), identifikátoru bezpečnostní úrovně a dalších bezpečnostních informací.

Implicitně není bezpečnost ve standardu 802.15.4 povolena. Pro povolení bezpečnosti musí vyšší vrstvy specifikovat bezpečnostní úroveň jinou než *None* v příslušném záznamu ACL, který koresponduje s cílovým zařízením. Výjimkou je potvrzovací rámec (*acknowledgment frame*, ACK), který musí mít nastavenu bezpečnostní úroveň *None*, tedy není nijak chráněn.

Aby se zjednodušila spolupráce mezi zařízeními, měla by všechna zařízení v síti používat stejnou bezpečnostní úroveň.

Bezpečnostní služby poskytované ZigBee zahrnují metody pro generování klíče, přenosu klíče, ochranu datového rámce a management zařízení. Tyto služby formují stavební prvky pro implementaci bezpečnostních politik na zařízení ZigBee.

4.1.1 Bezpečnostní architektura a návrh

Úroveň bezpečnosti poskytované bezpečnostní architekturou ZigBee závisí na úschově symetrických klíčů, na ochraně zabudovaných mechanismů a na korektní implementaci šifrovacích mechanismů a k tomu příslušných bezpečnostních politik. Poskytované bezpečnostní šifrovací služby chrání rozhraní mezi různými zařízeními; oddělení různých vrstev architektury v rámci jednoho zařízení se z pohledu zabezpečení neuvažuje.

Takový model, zvaný *open trust model*, dovoluje znovupoužití stejných klíčů mezi různými vrstvami jednoho zařízení, čímž se dosáhne bezpečnosti mezi dvěma koncovými zařízeními, namísto bezpečnosti mezi koncovými vrstvami v rámci jednoho zařízení.

Architektura vrstev, jak byla popsána v kapitole 3.1 – Struktura protokolu ZigBee, zahrnuje bezpečnostní mechanismy na třech vrstvách. Vrstvy MAC, NWK a APS jsou zodpovědné za bezpečný přenos svých rámců. APS podvrstva navíc poskytuje služby pro ustavení a údržbu bezpečnostních vztahů. Objekty ZigBee (ZDO) spravují bezpečnostní politiky a bezpečnostní konfiguraci zařízení.

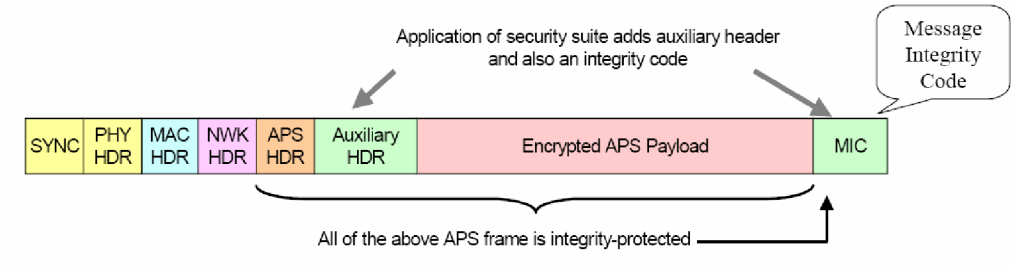
Při návrhu architektury z hlediska bezpečnosti se zohledňovala otázka zneužití sítě zlomyslným zařízením, které by využívalo síť k přenášení dat bez povolení. Jak jsem již dříve uvedl, podobný problém představuje neúmyslná interference mezi komunikujícími zařízeními. Proto byl jako hlavní bezpečnostní princip zvolen takový, že *vrstva, která vytvoří datový rámec je zodpovědná za jeho zabezpečení*. Například pokud vrstva MAC vytvoří rámec, který vyžaduje ochranu, použije se bezpečnost na úrovni vrstvy MAC. Podobně pokud vyžaduje ochranu rámec síťové vrstvy (NWK) nebo aplikační (APS) vrstvy, využije se zabezpečení na úrovni síťové nebo aplikační vrstvy (viz následující kapitolu).

Pokud je potřeba chránit síť před zneužitím poskytovaných služeb (např. ze strany zlomyslného síťového zařízení), musí být použita bezpečnost na úrovni síťové vrstvy NWK pro všechny rámce mimo těch, které představují komunikaci mezi routerem a nově připojeným zařízením (dokud toto zařízení neobdrží síťový klíč *Network key*). Tedy platí, že pouze zařízení, které se připojilo do sítě a úspěšně obdrželo síťový klíč, má možnost posílat své zprávy přes víc jak jeden hop v síti.

Použitím *open trust modelu* může být bezpečnost založena na znovupoužití klíčů každou vrstvou. Například aktivní síťový klíč bude použit pro zabezpečení všesměrových (broadcastových) rámců na aplikační vrstvě (APS), rámců na síťové vrstvě (NWK) nebo příkazů na vrstvě MAC. Znovupoužití klíčů pomáhá snižovat paměťové nároky zařízení.

4.1.2 Zabezpečení ZigBee rámců

Na obrázku 10 je znázorněn ZigBee rámec aplikační vrstvy se zabezpečením. ZigBee může přidávat hlavičky k datovým rámcům na vrstvách MAC, NWK a APS. Na každé z těchto vrstev se k hlavičce příslušné vrstvy (*MAC HDR* / *NWK HDR* / *APS HDR*) při aplikaci zabezpečení přidá hlavička zabezpečení (tzv. pomocná hlavička – *auxiliary HDR*). Hlavička příslušné vrstvy, hlavička se zabezpečením (pomocná) a data se pak mohou zakódovat do integritního kódu MIC pro ověření integrity dat.



Obrázek 10. Zabezpečený rámec aplikační vrstvy standardu ZigBee. (Zdroj [12])

4.1.3 Bezpečnostní klíče

Ve standardu ZigBee je bezpečnost mezi zařízeními založena na klíčích. V této souvislosti rozeznáváme tři typy klíčů:

- master klíč (*master key*)
- linkový klíč (*link key*)
- síťový klíč (*network key*)

ZigBee provádí centralizované řízení bezpečnosti přes tzv. *trust center* zařízení. V každé zabezpečené síti je právě jedno trust center zařízení. Trust center je zodpovědný za distribuci síťového klíče všem zařízením ve své síti a jeho údržbu, stejně jako za propojení dvou aplikací a umožnění koncové (*end-to-end*) bezpečnosti mezi zařízeními (např. distribucí *master* klíčů nebo *linkových* klíčů). Přes trust center zařízení také probíhá žádost o autentizaci nového zařízení při připojení do sítě. Trust centrem bývá většinou síťový koordinátor (viz kapitolu 3.2 - Typy zařízení).

Master klíč tvoří základ pro dlouhodobou bezpečnost mezi dvěma zařízeními. Master klíč může být na klientu přednastaven při výrobě nebo může být klientovi poslán *trust center* zařízením. Z master klíče se odvozuje linkový klíč, který představuje aktuální zabezpečení mezi dvěma komunikujícími zařízeními. Síťový klíč slouží pro zabezpečení sítě, chrání tedy před útoky zvenku. Linkový a síťový klíč mohou být periodicky aktualizovány.

Unicastová komunikace mezi aplikačními vrstvami APL dvou zařízení je zabezpečena 128-bitovým linkovým klíčem sdíleným oběma zařízeními, zatímco broadcastová komunikace (komunikace se všemi zařízeními v síti) je zabezpečena 128-bitovým síťovým klíčem sdíleným všemi zařízeními v konkrétní síti. Příjemce je vždy obeznámen s přesnými bezpečnostními opatřeními komunikace (příjemce tedy ví, jestli je rámec chráněn linkovým, nebo síťovým klíčem).

Master klíč může zařízení získat

- pomocí techniky *key-transport* nebo
- může být přednastaven při výrobě (*pre-installation*)

Linkový klíč může zařízení získat

- pomocí techniky *key-transport*,
- *key-establishment* nebo
- přednastavením při výrobě (*pre-installation*)

Síťový klíč může zařízení získat

- pomocí techniky *key-transport* nebo
- přednastavením při výrobě (*pre-installation*).

Technika *key-transport* je založena na přenosu (master, linkového nebo síťového) klíče od trust center zařízení ke koncovému zařízení.

Technika *key-establishment* pro získání linkového klíče je založena na *master* klíči. Z master klíče se technikou SKKE (*symmetric-key key establishment*) handshake vytvoří linkový klíč. Technika SKKE zahrnuje např. ověření použitelnosti master klíče v síti a navázání na konkrétní adresu – identifikaci zařízení.

Bezpečnost mezi zařízeními závisí na bezpečnostní inicializaci a instalaci těchto klíčů.

V zabezpečené síti je poskytována řada bezpečnostních služeb. Opatrnost velí, že bychom se měli vyhnout opakovanému používání stejných klíčů mezi různými bezpečnostními službami, což by mohlo vést k narušení bezpečnosti (při odhalení klíče). Tyto rozdílné služby používají klíč odvozený z linkového klíče použitím jednocestné funkce. Použití nekorelujících klíčů zajišťuje logické oddělení provádění jednotlivých bezpečnostních protokolů.

Síťový klíč může být použit vrstvami linkovou (MAC), síťovou (NWK) a aplikační (APL). Vrstvy sdílí stejný síťový klíč a mají k dispozici stejný mechanismus čítače příchozích i odchozích rámců (kvůli ochraně před podvrženými rámci). Linkové a *master* klíče mohou být použity pouze podvrstvou APS aplikační vrstvy, z čehož plyne, že linkové a *master* klíče jsou dostupné pouze aplikační vrstvě (APL).

4.1.4 Bezpečnost na úrovni jednotlivých vrstev

4.1.4.1 Bezpečnost na úrovni fyzické vrstvy (PHY)

Pro zvýšení spolehlivosti přenášených dat je vysílání prováděno technologií rozprostřeného spektra DSSS (*Direct Sequence Spread Spectrum*). Jednotlivé bity jsou nahrazeny početnější sekvencí bitů (*chipů*), které se pak vysílají. Signál je tak rozprostřen do větší části spektra a je více odolný vůči rušení. Uživatelům, kteří neznají mechanismus vytváření pseudonáhodné sekvence (čipovací sekvence), se přenášená data jeví jako šum.

4.1.4.2 Bezpečnost na úrovni linkové podvrstvy MAC

Když je potřeba zabezpečit rámec sestavený na linkové podvrstvě pro přístup k médiu (MAC), ZigBee použije zabezpečení specifikované na vrstvě MAC. Podvrstva MAC sama o sobě nemůže vyřešit všechny bezpečnostní problémy nízkorychlostních osobních sítí. Nemůže například poskytnout koncovou bezpečnost (*end-to-end*), která je důležitá pro některé aplikace, kde citlivé informace nesmí být vyzrazeny žádnému z mezilehlých uzlů. Koncové zabezpečení poskytuje trust center a je možné díky specifikaci open trust modelu (viz kapitolu 4.1.1 – Bezpečnostní architektura).

Při požadavku na ověření integrity je vytvořen kryptografický kontrolní součet MIC (*Message Integrity Code*) o délce 32, 64 nebo 128 bitů a je zahrnut do vysílaného MAC rámce. Na přijímací straně se provádí stejná operace a hodnota součtu se porovnává s přijatou. Pokud se zpráva během přenosu změnila, budou se hodnoty lišit a rámec je odmítnut. Integritu rámce lze zabezpečit i na dalších vrstvách, síťové a aplikační.

Pokud je nutné zajistit důvěrnost MAC rámce, je k němu přidána informace o pořadí rámce a klíče (*Frame Count*, *Key Sequence Count*). Na vysílací a přijímací straně je udržována aktuální

informace o čísle rámce. Pokud obdrží přijímací zařízení rámec s neplatným číslem, je detekováno narušení bezpečnosti.

Vrstva MAC je tedy zodpovědná za zabezpečení svého rámce, ale vyšší vrstvy musí vědět, jakou úroveň bezpečnosti mají použít.

4.1.4.3 Bezpečnost na síťové vrstvě (NWK)

Síťová vrstva je zodpovědná za provedení kroků vedoucích k bezpečnému přenosu odchozích rámců a přijetí příchozích rámců příkazů síťové vrstvy (požadavky na spojení a odpovědi na tyto požadavky). Vyšší vrstvy řídí bezpečnostní operace tím, že nastaví příslušné klíče a čítače rámců a ustaví, kterou bezpečnostní úroveň použít.

4.1.4.4 Bezpečnost na podvrstvě aplikační vrstvy (APS)

Vrstva APS je zodpovědná za provedení kroků vedoucích k bezpečnému přenosu odchozích rámců, přijetí příchozích rámců a správu kryptografických klíčů. Vyšší vrstvy řídí správu šifrovacích klíčů tím, že posílají vrstvě APS bezpečnostní příkazy (např. k vytvoření linkového klíče, přenosu klíče od jednoho zařízení k jinému nebo odpojení zařízení od sítě). Vyšší vrstvy také určují, jakou bezpečnostní úroveň použít k zabezpečení odchozích rámců.

4.1.5 Základní poskytované bezpečnostní služby

V této kapitole se podrobněji zmíním o základních bezpečnostních službách poskytovaných standardem ZigBee.

4.1.5.1 Sekvenční posloupnost dat, odmítnutí opakujících se rámců (*Freshness*)

Na rozdíl od univerzálních sítí jsou nízkorychlostní osobní sítě (LR-WPAN) většinou specifické pro určitou aplikaci. Informace přenášející se v LR-WPAN jsou často citlivé na čas. V takových sítích nestačí zaručit důvěrnost a autentizaci. Opakování starých (i když zabezpečených a autentizovaných) zpráv může podstatně narušit práci sítě a způsobit nějakou nepříjemnost. *Freshness* zajišťuje, že přijatá zpráva je nová (čerstvá) a tedy platná v daném kontextu aplikace.

Sekvenční posloupnost dat (*freshness*) také předchází útokům typu replay, kdy útočník vyšle již jednou poslanou zprávu, aby získal odpověď a mohl ji později zneužít. Zařízení ZigBee udržují čítače posloupnosti příchozích i odchozích rámců. Čítač je resetován při vytvoření nového klíče.

4.1.5.2 Integrita přenášených zpráv

Jedním ze základních požadavků na bezpečnou komunikaci je, aby zpráva byla přijata v takové podobě, v jaké byla odeslána. Přesto se může stát, například kvůli cíleným útokům nebo kvůli různým poruchám (kolize při přenosu či zhoršení přenosových podmínek), že zpráva se při přenosu poškodí. Integrita garantuje, že je zpráva přenesena bez podvržení, smazání, vložení, přeskládání nebo jakékoli jiné modifikace. Proto se ke každému paketu přikládá integritní kód MIC, který zajišťuje integritu zprávy. MIC můžeme chápat jako kryptograficky bezpečný kontrolní součet zprávy. Její spočítání vyžaduje od autorizovaných odesílatelů a příjemců sdílení tajného kryptografického klíče a tento klíč je součástí vstupu při počítání. Odesílatel spočítá MIC nad paketem a přiloží jej k paketu. Příjemce sdílící stejný tajný klíč přepočítá MIC a porovná s hodnotou v paketu. V každém případě musí být obtížné spočítat MIC bez znalosti tajného klíče.

Integritní kód MIC může mít velikost 32, 64 nebo 128 bitů. Zabezpečení integrity dat je kompromisem mezi nezabezpečenou komunikací bez dodatečné režie a šifrovanou komunikací, která vyžaduje vyšší režii. Samotná integrita nezahrnuje šifrování zprávy.

4.1.5.3 Autentizace (řízení přístupu)

Autentizace je používána uzlem, který si ověřuje identitu jiného uzlu, se kterým komunikuje (autentizace uzlu), nebo původ zprávy (autentizace původu dat). Autentizace je důležitá v pomalých osobních sítích zejména při administrativních činnostech jako je připojování, odpojování, výměna koordinátora sítě (trust center), vysílání rámce *beacon* nebo řešení konfliktu mezi identifikátory osobních sítí (PAN).

Autentizace poskytuje ujištění o odesílateli zprávy, znemožňuje útočníkovi modifikovat zařízení a tím se vydávat za někoho jiného. Legitimní uzly by měly být schopné detekovat zprávy od neautentizovaných uzlů a odmítnout je. Autentizace je možná na úrovni sítě, nebo na úrovni zařízení. Autentizace na úrovni sítě je dosažena pomocí běžného síťového klíče a je prevencí proti útokům zvnějšku sítě s minimálními paměťovými nároky. Autentizace na úrovni zařízení je dosažena pomocí unikátního linkového klíče sdíleného dvojicí zařízení a je prevencí proti útokům zevnitř i zvnějšku sítě, ale vyžaduje vyšší paměťové nároky.

4.1.5.4 Důvěrnost (Šifrování dat)

Důvěrností se rozumí uchování informace v tajnosti před neautorizovanými uživateli. Hlavním cílem důvěrnosti je ujištění, že citlivá data nebudou odhalena nikomu jinému než určenému příjemci. Typicky se toho dosahuje šifrováním. Šifrování dat tedy zamezuje útočníkovi odposlechnout komunikaci na síti. Šifrování by mělo v lepším případě zabránit útočníkovi dozvědět se i částečnou informaci o zašifrované zprávě. Tato silnější vlastnost se označuje jako sémantická bezpečnost. Jedna vlastnost plynoucí ze sémantické bezpečnosti je ta, že zašifrování jedné zprávy dvakrát musí dát dvě rozdílné zašifrované zprávy. Běžná technika pro dosažení sémantické bezpečnosti je použití unikátní výzvy pro každé provedení algoritmu. Výzvu si lze představit jako vedlejší vstup šifrovacího algoritmu. Hlavním účelem přidání náhodné výzvy je vnesení odchylky do šifrovacího procesu. Výzvy jsou typicky posílány nezašifrované a přiložené k paketu se zašifrovanými daty.

ZigBee používá 128-bitový klíč šifrovací metody AES. Šifrovací standard AES (*Advanced Encryption Standard*) nahrazuje svého předchůdce - standard DES. Výhodou tohoto nového způsobu šifrování je, že nehrozí útok hrubou silou (tj. vyzkoušení všech možných klíčů). Výpočetní zdroje potřebné pro provedení útoku hrubou silou rostou exponenciálně s délkou šifrovacího klíče, nikoli lineárně (standard DES používal 56-bitové klíče).

Šifrování může probíhat (stejně jako autentizace) na úrovni sítě, nebo na úrovni zařízení. Šifrovat na úrovni sítě můžeme pomocí standardního síťového klíče a předcházet tak útokům zvnějšku. Šifrování na úrovni zařízení je možné použitím linkového klíče mezi dvěma zařízeními, takové šifrování představuje ochranu proti útokům zevnitř i zvnějšku sítě

Šifrování lze vypnout bez dopadu na sekvenční posloupnost rámců, integritu nebo autentizaci. Některé aplikace totiž nevyžadují ochranu šifrováním a vypnutí šifrování pomůže urychlit komunikaci (snížit latenci) v případě potřeby (např. pro účely regulace).

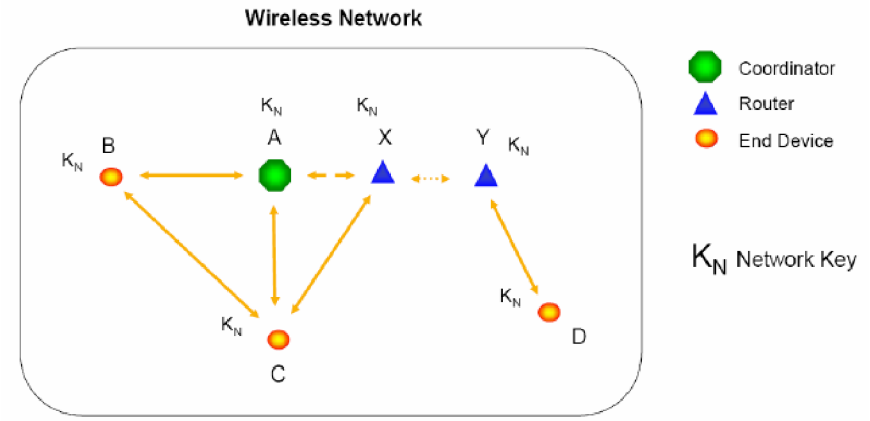
4.1.6 Bezpečnostní režimy

ZigBee rozlišuje dva bezpečnostní režimy, ve kterých se může nacházet trust center:

- rezidenční režim (residential mode)
- komerční režim (commercial mode)

4.1.6.1 Rezidenční režim

Rezidenční režim (obrázek 11) se používá pro zabezpečenou bezdrátovou síť, která nevyžaduje žádné další znalosti o bezpečnosti (typicky domácí síť). Vlastník sítě nemusí hrát žádnou aktivní roli při udržování bezpečnosti sítě. Trust center umožňuje zařízením připojit se k síti (pokud mají přednastavený síťový klíč), ale nevytváří a nedistribuuje žádné další klíče. Trust center také neprovádí údržbu klíčů, takže je ani pravidelně neaktualizuje. Tento režim vyžaduje minimum paměti, která neroste se sítí, protože trust center potřebuje uchovávat pouze jediný síťový klíč.

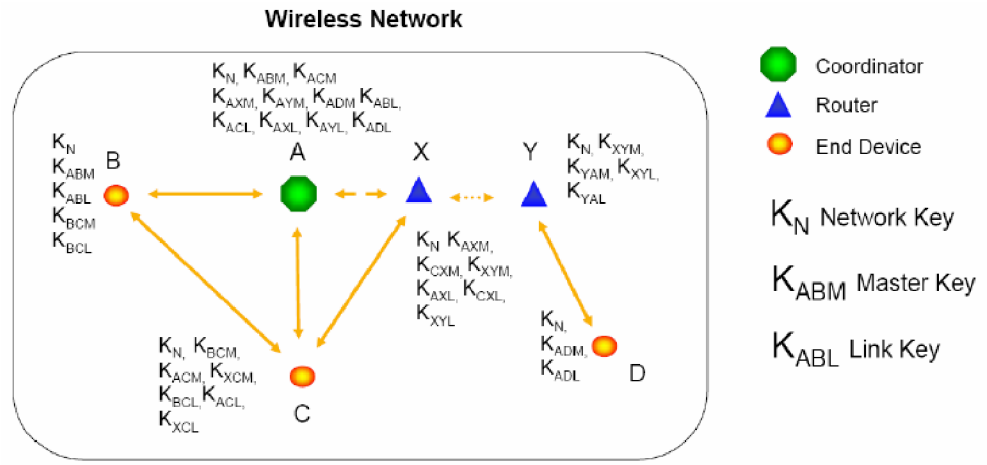


Obrázek 11. Bezdrátová síť v rezidenčním módu (pouze se síťovým klíčem). (Zdroj [12])

Všechna zařízení musí vlastnit pouze síťový klíč a čítač rámců, rezidenční režim poskytuje zabezpečení na úrovni sítě (tedy chrání proti útokům zvnějšku sítě). K šifrování se používá pouze síťový klíč.

4.1.6.2 Komerční režim

Naproti tomu komerční režim (obrázek 12) se používá pro bezdrátovou síť, která řídí kritické aplikace (osvětlení firem, alarmy, monitoring nebo řízení výroby). Taková bezdrátová síť je aktivně monitorována a spravována (pravidelné aktualizace klíčů, kontrolované připojování zařízení).
Trust center v komerčním režimu umožňuje připojit pouze zařízení, která byla ručně povolena (nastavena jejich adresa). Trust center pravidelně aktualizuje síťový klíč a linkové klíče. Šifrování se provádí pomocí linkového klíče, což zajišťuje mnohem větší bezpečnost než při šifrování síťovým klíčem v rezidenčním módu.



Obrázek 12. Bezdrátová síť v komerčním módu (se všemi typy klíčů). (Zdroj [12])

Zařízení vlastní mimo sdíleného síťového klíče také linkové klíče potřebné k bezpečné unicastové komunikaci uvnitř sítě. Zařízení jsou tak chráněna proti útokům jak zvnějšku sítě (sdílený síťový klíč), tak i proti útokům zevnitř (unikátní linkové klíče pro dvojici zařízení).

4.2 Bezpečnostní problémy

V původním bezpečnostním návrhu senzorových sítí i standardu ZigBee jsou některá slabá místa [15], [23]. Nemožnost zabezpečit potvrzovací rámec podvrstvy MAC nutně narušuje celou bezpečnostní architekturu senzorových sítí. Útočník má možnost podvrhnout potvrzovací rámec, čímž může provést různé útoky. Například může ochromit mechanismus opakovaného přenosu paděláním potvrzovacího rámce pro data nebo příkaz poškozeného kvůli kolizi, šumu nebo záměrného rušení od samotného útočníka. Pomocí vydávání se za neexistující zařízení může útočník také donutit zařízení přenášet rámce k neexistujícímu zařízení. Jedno praktické řešení tohoto problému je umožnit zdroji zjistit, zda je potřeba zabezpečený nebo nezabezpečený potvrzovací rámec.

Dalším problémem je přetečení čítače. Jak IEEE 802.15.4, tak ani ZigBee neposkytují mechanismus k prevenci útoku typu odmítnutí služby (denial of service, DoS). Takový útok lze provést přetečením čítače rámců: zfalšováním rámce IEEE 802.15.4 a nastavením jeho čítače na maximální hodnotu $2^{32} - 1$.

Mezi problémy standardu 802.15.4 lze zařadit [23]: management IV (inicializačního vektoru), management klíčů a nedostatečná ochrana integrity. Specifikace 802.15.4 ustanovuje individuální klíč a formát IV, který vede ke zranitelnostem. Bezpečnostní protokol na úrovni linkové vrstvy zajišťuje čtyři základní bezpečnostní služby: kontrolu přístupu, integritu dat, důvěrnost dat a ochranu proti znovuposlání. Nyní se o některých problémech zmíním podrobněji.

4.2.1 Problémy při managementu inicializačního vektoru

Jedním z možných bezpečnostních problémů standardu 802.15.4 je výskyt dvou či více stejných klíčů v položkách ACL. ACL může mít až 255 položek, které slouží k uchování různých klíčů a k nim příslušných inicializačních vektorů. Odesílatel zvolí patřičný ACL záznam na základě cílové adresy. Pokud však použijeme stejný klíč ve dvou různých položkách, je pravděpodobné, že omylem znovupoužijeme inicializační vektor. Při použití režimu AES-CCM, který využívá režimu CTR, lze

snadno narušit důvěrnost zpráv, protože pokud dvě zprávy budou xorovány pomocí stejného inicializačního vektoru, lze operací xor aplikovanou na obě zašifrované zprávy získat obě původní zprávy. Přestože se v těchto případech poruší důvěrnost zpráv, jejich integrita zůstane neporušena.

Odesílatel může bezpečně poslat dvě zprávy dvěma rozdílným příjemcům za použití stejného klíče, pokud bude pečlivě hlídat stav inicializačního vektoru. Jednou z možností je použití jednoho záznamu v ACL. Odesílatel pak může poslat první zprávu, změnit cíl v záznamu ACL a pak poslat druhou zprávu. Obecným principem tedy je, že aktuální stav inicializačního vektoru by neměl být oddělen od bezpečnostního klíče.

Dalším problémem může být ztráta stavu ACL při výpadku napájení. Zařízení postavená podle standardu 802.15.4 jsou většinou napájena bateriově. Návrháři čipů by měli dbát na to, aby stav ACL byl náležitě ošetřen i při výpadku napájení nebo ve stavu provádění nízkopříkonových (low-power) operací. Pokud se po návratu z výpadku napájení vynuluje ACL tabulka, obvykle lze reinicalizovat bezpečnostní klíče pro jednotlivá zařízení, ale již není jasné, co bude s inicializačními vektory. Pokud přejdou do předem známého stavu (tedy vynulují se), některé hodnoty IV budou znovu opakovány, čímž dojde k bezpečnostní kompromitaci. Řešením může být vytvoření nových bezpečnostních klíčů po výpadku napájení a tak nedojde k znovupoužití IV se stejným klíčem; nebo lze ukládat hodnoty čítače do elektricky nezávislé flash paměti. Druhá možnost je však časově i energeticky náročná.

Stejný problém (uchování stavu inicializačních vektorů) může nastat při přechodu zařízení do tzv. low-power módu. Je to stav, kdy zařízení funguje na nejmenší možný příkon, například pouze pro přijímání paketů. Pokud se pak dostane do běžného stavu (kdy může data i vysílat, k čemuž potřebuje ACL) s vynulovanou ACL tabulkou, znovu použije již jednou použité hodnoty IV a dojde k porušení podmínky důvěrnosti.

4.2.2 Další bezpečnostní problémy

Další oblast problémů vzniká z neadekvátní podpory v ACL tabulce pro více modelů použití klíčů a z nedostatečně vyřešené ochrany integrity. Podrobnosti lze nalézt v [23].

5 Praktická část

Praktickou část jsem prováděl se ZigBee kitem PICDEM DM163027-2 od společnosti Microchip. Měl jsem k dispozici dva moduly (jedno RFD zařízení a jeden koordinátor), každý modul měl vlastní anténu, která slouží jako transceiver. Moduly jsou napájené 9V baterií a obsahují rozhraní RJ-12 a RS-232. Pomocí rozhraní RJ-12 se připojuje programátor, kterým se do modulu nahraje konkrétní program, jehož funkce pak modul vykonává, a pomocí něj je možné také provádět ladění programu. Pomocí sériového rozhraní RS-232 lze připojit modul k počítači a tak prostřednictvím terminálu sériové linky a programových výpisů sledovat akce, které modul vykonává (například připojení k síti, posílání zpráv atp.).

Pracoval jsem s implementací protokolu ZigBee od firmy Microchip ve verzi 1.0-3.8. Toto označení se týká verze ZigBee protokolu (1.0) a verze implementace firmy Microchip (3.8). Vrstvový architektura ZigBee protokolu je označována jako ZigBee stack. Zmiňovaná implementace zahrnuje také demonstrační aplikaci, která je však napsána pro jiný typ transceiveru (konkrétně MRF24J40, http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1406&dDocName=en520396). Kit, se kterým jsem pracoval, obsahoval transceiver Chipcon CC2420. Podle dokumentů firmy Microchip a příspěvků na fóru zmíněné firmy je napsaná demo aplikace alespoň částečně kompatibilní s oběma transceivery, jen je nutné ji mírně modifikovat. Kompatibilita se týká pouze komunikace v nezabezpečeném režimu. Pokud jde o zabezpečený režim, tak Microchip upozorňuje, že je nutné modifikovat kód. Ale ani tak není od Microchipu zajištěna kompatibilita s transceiverem CC2420, protože tento typ transceiveru již firma Microchip nepodporuje.

Implementace protokolu ZigBee firmou Microchip v 1.0-3.8 nově přidává podporu všech tří síťových konfigurací (viz kapitola 3.3 – Topologie sítě). Moduly od Microchipu mohou pracovat v jakékoli z těchto tří síťových konfigurací, záleží vždy jen na nahraném programu, jakou topologii bude podporovat. V mém případě (dva ZigBee moduly, jeden jako RFD zařízení a druhý jako koordinátor) se jednalo o topologii typu hvězda, kdy RFD zařízení komunikuje s ostatními zařízeními pouze prostřednictvím koordinátora. V praxi to znamená, že RFD periodicky žádá koordinátora o data, která mu někdo přes koordinátora poslal nebo pocházející přímo od koordinátora. Přidaná podpora funkce routeru umožňuje sestavit ze zmíněných modulů firmy Microchip konfiguraci typu síť, kdy FFD zařízení může komunikovat přímo s ostatními FFD zařízeními, díky čemuž je umožněno dynamické směrování.

Demonstrační aplikace implementuje komunikaci mezi koordinátorem sítě a koncovým bodem (RFD zařízením). Jako profil je použit dodávaný „HCLighting“, což je profil pro ovládání domácího osvětlení. Definici pojmu profil a dalších souvisejících pojmů naleznete v kapitole 5.1. Po spárování obou zařízení (provádí se stiskem tlačítka RB5 na koordinátoru a následném stisku tlačítka RB5 na RFD) lze posílat zprávy mezi moduly stiskem tlačítka RB4 na obou zařízeních. Úspěšně přijatá zpráva se projeví rozsvícením nebo zhasnutím diody. Podrobný popis funkce aplikace je v [22]. K překladu kódu jsem použil překladač MCC18 a k práci s kódem vývojové prostředí MPLAB IDE v poslední verzi 7.52. Toto prostředí se používá i ke komunikaci s programátorem, kterým byl v mém případě MPLAB ICD 2. Programátor lze připojit k počítači přes USB nebo sériový COM port, z druhé strany k modulu se připojuje přes síťové rozhraní RJ-12.

Dalším přístrojem, se kterým jsem pracoval byl síťový analyzátor ZENA Network Analyzer (Product# DM183023, http://www.microchipdirect.com/productsearch.aspx?Keywords=DM183023) a kromě nastavení bezpečnostních a funkčních parametrů (výstupní přijímací výkon, povolené kanály,

povolené clustery na vybraném aplikačním profilu apod.) umožňuje také zobrazovat a analyzovat (filtrovat) odchycené pakety. Omezená verze programu je k dispozici bezplatně ke stažení na stránkách výrobce. Omezení spočívá v nemožnosti sledovat zachytávanou síťovou komunikaci v reálném čase. Po skončení komunikace je pak možné uložený soubor zobrazit a analyzovat až zpětně. V průběhu mé práce s tímto analyzátorem došlo k aktualizaci programu z verze 1.1 na 2.0, takže všechny vyobrazené komunikační pakety pocházejí z verze 2.0.

ZENA umí dešifrovat zabezpečený přenos (pokud má k dispozici síťový klíč) a zobrazit jej v okně zachytávaných paketů. Tato funkce je podle výrobce určena pro podporu vývojových účelů. Síťová komunikace nemůže být dešifrována pokud není znám síťový klíč a použitá bezpečnostní úroveň. V současné době ZENA podporuje dešifrování zabezpečené komunikace na úrovni vrstev MAC a NWK. Aplikační vrstva (APS) zatím není podporována. Dešifrování paketů v reálném čase je výpočetně náročné a proto není vyloučeno, že při hustém síťovém provozu může docházet k vypadávání paketů. V takovém případě je lepší provádět dešifrování až při prohlížení zachycených paketů.

5.1 Terminologie protokolu ZigBee

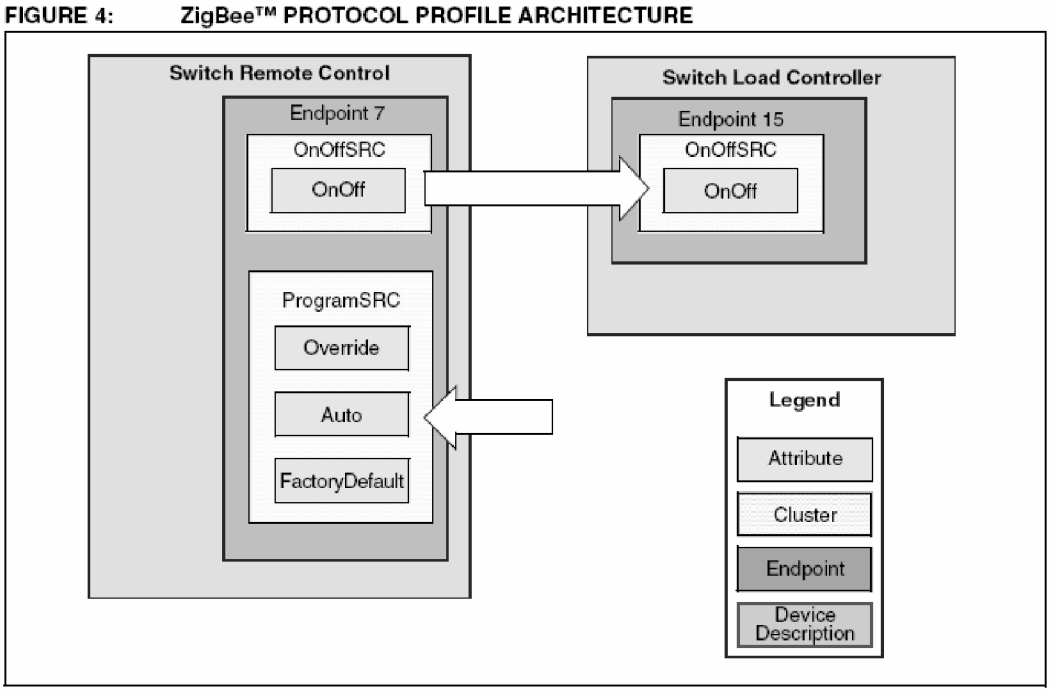
Profil protokolu ZigBee je popis logických komponent (zařízení) a jejich rozhraní. S profilem není obvykle asociován žádný kód. Každý kousek dat, který může být přenesen mezi zařízeními, jako např. popis stavu nebo hodnoty nějakého měření, je nazýván atribut. Každý atribut je přiřazen k unikátnímu identifikátoru. Tyto atributy jsou shlukovány v clusterech. Každému clusteru náleží jedinečný identifikátor. Rozhraní jsou specifikována na úrovni clusterů, nikoli atributů (viz Obrázek 13).

Profil definuje hodnoty ID atributu a ID clusteru, stejně jako formát každého atributu. Například pro profil Ovládání osvětlení domácnosti cluster OnOffSRC na dálkovém ovládaní osvětlení obsahuje jeden atribut, OnOff, kterým musí být 8-bitová hodnota (0xFF – zapnuto, 0x00 – vypnuto a 0xF0 – změň stav). Tato hodnota Profil také definuje, které clustery jsou povinné a které jsou volitelné pro každé zařízení. Navíc může profil definovat některé volitelné služby ZigBee protokolu jako povinné.

Uživatel může vzít tyto definice a napsat podle nich svůj kód. Může napsat jakýkoli kód, tzn. jakkoli seskupovat funkce, stačí když implementuje všechny povinné clustery a služby a použije atributy definované v profilu. Tímto způsobem pak mohou různí výrobci vytvořit zařízení, která budou vzájemně spolupracovat (pokud budou vyrobena podle stejného profilu).

Jako příklad profilu lze uvést Ovládání domácího osvětlení, které specifikuje šest zařízení. Implementace protokolu ZigBee od firmy Microchip poskytuje podporu pro tento profil ve formě hlavičkových souborů s následujícími informacemi: ID profilu, ID zařízení a jeho verze, ID clusteru, ID atributů a datové typy atributů. Každý blok kódu s určitou funkcí, který podporuje jeden nebo více clusterů, se nazývá koncový bod (endpoint). Rozdílná zařízení komunikují prostřednictvím svých koncových bodů a clusterů, které podporují.

Obrázek 13 graficky znázorňuje jak spolu souvisí pojmy jako atribut, cluster, endpoint a zařízení. Na obrázku jsou dvě zařízení pracující v profilu Ovládání domácího osvětlení. Každé zařízení má pouze jeden koncový bod. Světlo (Switch Load Controller) obsahuje jeden vstupní cluster. Přepínač/ovladač (Switch Remote Control) má ve svém koncovém bodě jeden vstupní a jeden výstupní cluster. Přepínač by mohl být také implementován tak, že clustery by byly ve dvou oddělených koncových bodech. Tok dat probíhá na úrovni clusterů.



Obrázek 13. Profil demonstrační aplikace. (Zdroj [22])

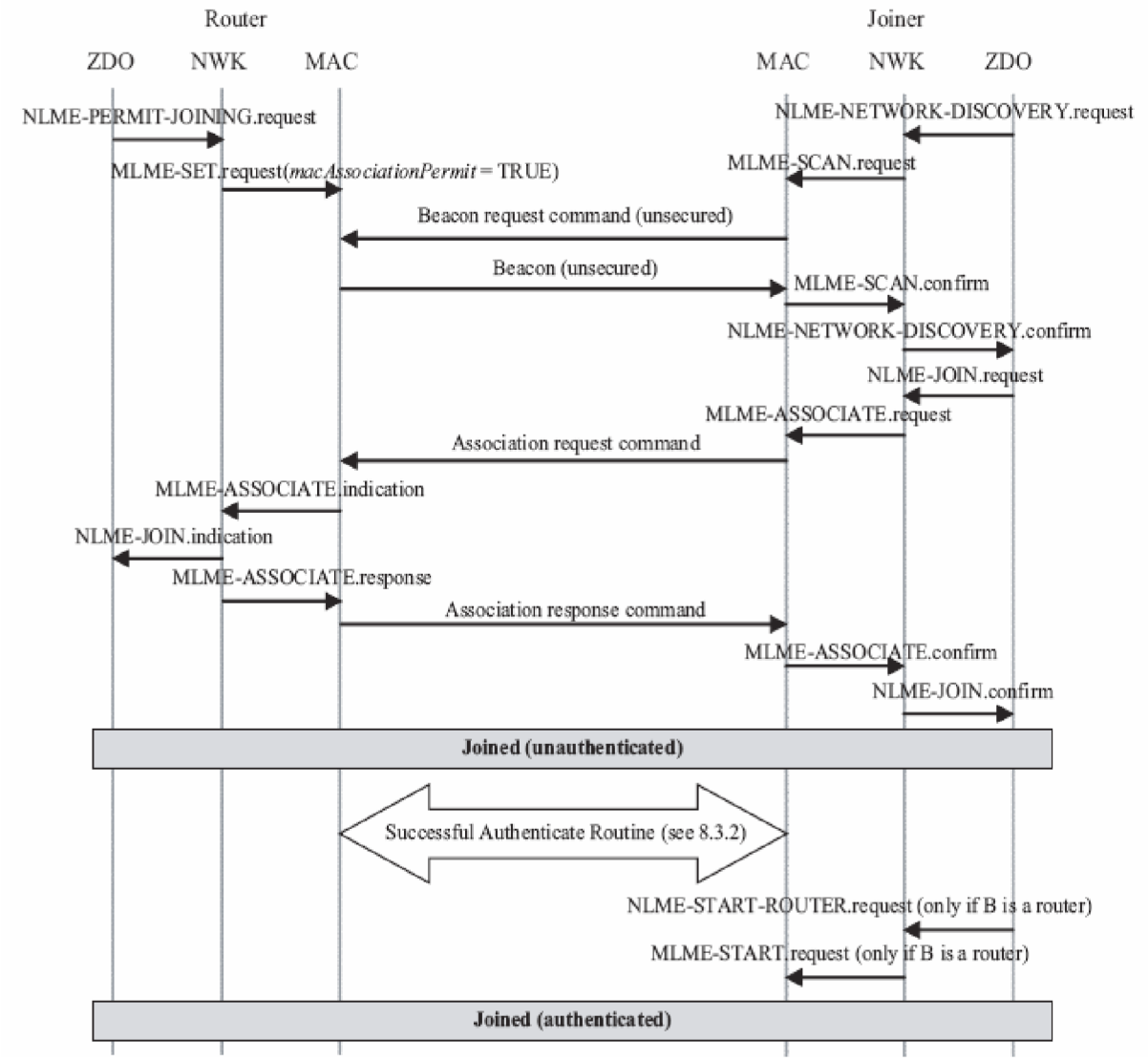
5.1.1 Typy zpráv a párování (binding)

Zařízení může komunikovat s jinými zařízeními v síti pokud zná jejich síťovou adresu. Takové zprávy se nazývají přímé zprávy (*direct messages*), v síťové terminologii je tento typ komunikace znám jako *unicast*. Pokud zařízení nezná adresu nebo chce poslat zprávu všem okolním zařízením, pošle tzv. nepřímou zprávu (*indirect messages*), známé také jako *broadcast*.

K posílání přímých zpráv je potřeba velká režie obsažená v objevování (*discovering*) okolních zařízení a správě adres objevených zařízení v paměti. Protokol ZigBee proto nabízí vlastnost nazvanou párování (*binding*), která má zjednodušit posílání zpráv. Párování vytvoří logickou vazbu mezi aplikačními objekty. Koordinátor si vytvoří tabulku záznamů (*binding table*) na úrovni clusterů nebo koncových bodů. Párování může být iniciováno buď ze strany koncového zařízení nebo může být vytvořeno koordinátorem či jiným zařízením. Po vytvoření „páru“ spolu mohou komunikovat dvě zařízení přes koordinátora. Zdroj zprávy zasílá svoji zprávu koordinátoru, který identifikuje zdroj zprávy, příjemce a ID clusteru a podle toho ji dále přepošle jednomu nebo více cílovým zařízením. Tyto zprávy se nazývají nepřímé (indirect). Výhodou takové komunikace je, že koncová zařízení nemusí znát adresy ostatních zařízení, se kterými komunikují.

5.2 Bezpečnostní procedury

V této kapitole se zmíním o připojování zařízení k zabezpečené síti, autentizaci nově připojeného zařízení, práci s bezpečnostním klíčem a odpojování od zabezpečené sítě.



Obrázek 14. Příklad připojování k zabezpečené síti. (Zdroj [7])

Po zapnutí začne připojované zařízení (Joiner) připojovací proceduru vysláním primitivy **NLME-NETWORK-DISCOVERY.request** z vrstvy ZDO. Tato primitiva vyvolá v síťové (NWK) vrstvě **MLME-SCAN.request** primitivu, která způsobí **přenos** nezabezpečeného **beacon request** rámce, kterým zařízení žádá o ohlášení již existujících sítí. Připojované zařízení pak obdrží beacon rámce od nejbližších routerů nebo koordinátorů a síťová (NWK) vrstva vyšle **NLME-NETWORK-DISCOVERY.confirm** primitivu. Parametr NetworkList této primitivy indikuje všechny blízké sítě (PAN) společně s jejich atributy *nwkSecurityLevel* a *nwkSecureAllFrames*. Router (přesněji zařízení, ke kterému se připojujeme, v tomto případě se jedná o koordinátora sítě) na obrázku 14 je nakonfigurován do stavu, kdy všechny jeho rámce typu beacon mají položku "association permit" nastavenu na "1" (tedy dovoluje připojení dalších zařízení ke své síti).

Připojované zařízení se rozhodne, ke které síti (PAN) se připojí (např. na základě bezpečnostních atributů obdržených v primitivě NLME-NETWORK-DISCOVERY.confirm) a vybrané síti pošle primitivu **NLME-JOIN.request**, kterou žádá o připojení. Pokud již připojované zařízení má síťový klíč pro tuto vybranou síť, parametr SecurityEnable v primitivě NLME-JOIN.request bude nastaven na TRUE; jinak bude nastaven na FALSE. V mém případě jsem klíče na

obou zařízeních přednastavil (v konfiguračním souboru *zigbee.def*), takže připojení již probíhalo zabezpečeně. Jak je vidět na obrázku 14, primitiva NLME-JOIN.request způsobí zaslání žádosti o připojení (Association request command) směrem k routeru. Po přijetí žádosti o připojení vyšle router primitivu **MLME-ASSOCIATE.indication** s parametrem SecurityUse nastaveným na TRUE nebo FALSE, podle toho, jestli byla žádost o připojení poslána zabezpečeně či nikoli. Poté síťová (NWK) vrstva routeru předá vrstvě ZDO primitivu **NLME-JOIN.indication**. Router v této chvíli zná adresu připojovaného zařízení a to, jestli byl použit síťový klíč k zabezpečení žádosti o připojení. Router poté pošle připojovanému zařízení primitivu **MLME-ASSOCIATE.response** s parametrem SecurityEnable nastaveném na TRUE nebo FALSE, podle toho, zda žádost o připojení byla poslána zabezpečeně či nikoli. Tato primitiva způsobí zaslání odpovědi na žádost o připojení (Association response command).

Po přijetí odpovědi na žádost o připojení vyšle připojované zařízení primitivu **NLME-JOIN.confirm**. Připojované zařízení je nyní ve stavu "připojeno, ale neautentizováno". Autentizační rutina (viz další kapitolu) bude následovat.

Pokud připojované zařízení není router, dostane se do stavu "připojeno a autentizováno" ihned po úspěšném dokončení autentizační routiny.

Pokud je připojované zařízení router, dostane se do stavu "připojeno a autentizováno" pouze po úspěšném dokončení autentizační rutiny následované dalšími inicializačními operacemi. Tyto operace jsou inicializovány vrstvou ZDO připojovaného zařízení vyvoláním primitivy **NLME-START.request**, která způsobí předání primitivy **MLME-START.request** k vrstvě MAC připojovaného zařízení.

Pokud router odmítne připojované zařízení, jeho odpověď na žádost o připojení obsahuje pole "association status" s hodnotou jinou než "0x00" a poté, co se tento parametr dostane k ZDO připojovaného zařízení v primitivě NLME-JOIN.confirm, připojované zařízení už nezačíná autentizační rutinu.

5.2.1 Autentizace

Jakmile se zařízení připojí k zabezpečené síti a je ve stavu „připojeno, ale neautentizováno", musí být autentizováno podle následujících pravidel.

5.2.1.1 Router

Pokud router není trust centrem, začne s autentizací hned po obdržení primitivy NLME_JOIN.indication, a to vyvoláním primitivy APSME-UPDATE-DEVICE.request s parametrem *DestAddress* nastaveném na *apsTrustCenterAddress* v AIB a parametrem *DeviceAddress* nastaveném na adresu nově připojeného zařízení. Parametr *Status* bude nastaven na 0x00 (tzn. zabezpečené připojení), pokud nově připojené zařízení zabezpečilo žádost o připojení, jinak bude nastaven na 0x01 (tzn. nezabezpečené připojení).

Pokud je router trust centrem, začne autentizační proceduru jednoduše tak, že se bude chovat jako trust center.

5.2.1.2 Trust center

Role trust centra v autentizační proceduře se aktivuje po přijetí příkazu update-device nebo po přijetí primitivy NLME-JOIN.indication (v případě, že router je trust centrem). Trust center se chová různě v závislosti na těchto okolnostech:

- zda trust center umožňuje novým zařízením připojit se k síti

- jestli je trust center v rezidenčním nebo komerčním režimu (viz kapitolu 4.1.6)
- pokud je v rezidenčním režimu, tak jestli se zařízení připojuje zabezpečeně nebo nezabezpečeně (indikuje to položka *Status* v příkazu update-device)
- pokud je v komerčním režimu, tak jestli má trust center master klíč korespondující s nově připojeným zařízením
- parametr *nwkSecureAllFrames* v NIB (Network Information Base)

Pokud se kdykoli během autentizační procedury trust center rozhodne nevpustit nové zařízení do sítě (např. na základě definované bezpečnostní politiky nebo kvůli chybě při výměně klíče), započne akce k odstranění zařízení ze sítě, a to vyslání primitivy APSME-REMOVE-DEVICE.request nadřazenému uzlu (tzn. routeru) se žádostí o odpojení nového zařízení nebo (pokud je sám routerem nového zařízení) vysláním primitivy NLME-LEAVE.request.

5.2.1.3 Připojování zařízení

Poté, co bylo připojované zařízení úspěšně asociováno k zabezpečené síti, bude také spolupracovat při níže popsané autentizační proceduře. Připojované zařízení po úspěšné autentizační proceduře nastaví atributy *nwkSecurityLevel* a *nwkSecureAllFrames* v NIB na hodnoty obsažené v beacon rámci od routeru.

Připojené a autentizované zařízení v zabezpečené síti s parametrem *nwkSecureAllFrames* s hodnotou TRUE bude vždy aplikovat zabezpečení síťové (NWK) vrstvy na odchozí i příchozí rámce (pokud nebudou směrovány od/k novému dosud neautentizovanému zařízení).
Míra účasti na autentizační proceduře závisí na stavu zařízení. Mohou nastat tři případy:

- zařízení s přednastaveným síťovým klíčem (rezidenční režim)
- zařízení s přednastaveným trust centrem master klíčem a adresou (komerční režim)
- bez přednastavení (dosud neurčený režim – může být rezidenční nebo komerční)

Protože ZigBee stack (v 1.0-3.8) od společnosti Microchip podporuje zatím pouze rezidenční režim s přednastaveným síťovým klíčem, budu se dále zabývat pouze první variantou.
Pokud bylo připojující se zařízení přednastaveno síťovým klíčem (a asociace byla úspěšná), nastaví čítač odchozích i příchozích rámců pro tento klíč na nulu a čeká na přijetí síťového klíče od trust centra se samými nulami. Po přijetí primitivy APSME-TRANSPORT-KEY.indication s parametrem, *KeyType* nastaveném na 0x01 (tzn. síťový klíč), zařízení nastaví parametr *apsTrustCenterAddress* ve své AIB na parametr *SrcAddress*, který obdrželo v primitivě APSME-TRANSPORT-KEY.indication. Zařízení je od této chvíle považováno za autentizované a může provádět operace běžné pro rezidenční režim. Pokud zařízení připojující se do zabezpečené sítě nebude autentizováno v předem daném čase, musí opustit síť.

5.3 Zabezpečení demonstrační aplikace

V následující části práce budu popisovat postup při seznamování se s demonstrační aplikací dodávané spolu s kitem a zabezpečení komunikace. Oba moduly byly připojené na sériový port počítače, a tak bylo možné sledovat i jejich výpisy. Zároveň jsem všechnu komunikaci zachytával síťovým analyzátorem ZENA, o němž se podrobněji zmíním později.

5.3.1 Inicializace a počáteční problémy

Nejprve se zmíním o počátečních problémech se zprovozněním modulů. Následující text budu prokládat výpisy z terminálu a obrázky odchycené komunikace.

Po zapnutí koordinátora:

```
**********************************
Microchip ZigBee(TM) Stack - v1.0-3.8
ZigBee Coordinator
Transceiver-CC2420

Hardware initialized
ZigBee initialized

Trying to start network...
PAN 1BD5 started successfully.
Turning on joining...
Joining status changed.
```

Frame	Time(us)	Len	MAC Frame Control					Seq	Dest	Dest	Beacon	FCS		
	+25704736		Type	Sec	Pend	ACK	IPAN	Num	PAN	Addr	Request	RSSI	Corr	CRC
00001	=25704736	10	CMD	N	N	N	N	0x00	0xFFFF	0xFFFF		-12	0x6B	OK

Obrázek 15. Koordinátor vyhledává již vytvořené sítě.

Koordinátor vytvořil síť PAN (Personal Area Network) s identifikátorem 1BD5 a zahájil proces hledání existující sítě: vyslal rámec typu Beacon Request a hledá další router nebo koordinátora (s již vytvořenou sítí), ke kterým by se připojil.

Po stisknutí tlačítka RB5:

```
Trying to perform end device binding.
Receiving ZDO cluster A0
End device bind/unbind invalid response.
```

Koordinátor se pokouší spárovat s dalším zařízením. V tomto případě není žádné k dispozici, protože RFD měl počáteční problémy (popíšu je později), takže neúspěšně.

Po stisknutí tlačítka RB4:

```
Trying to send light switch message.
Error 01 sending message.
```

Koordinátor se pokouší poslat zprávu (s příkazem zapnutí/vypnutí diody), ale protože nemá k dispozici žádné další zařízení, tak opět neúspěšně.

Při zachytávání ZigBee síťové komunikace jsem občas zaznamenal i WiFi komunikaci na stejné frekvenci (2.4GHz) a stejném kanálu. Takto přijatá data jsou většinou označena jako neplatná a vždy nesouhlasí CRC kód. Někdy je však paket alespoň částečně rozpoznán (viz obrázek 16, ze kterého lze zjistit PAN ID komunikujících sítí a také to, že je paket šifrován).

Frame	Time(us)	Len	Invalid Data				FCS		
	+53715808		0x67	0xCE	0xCC	0x5D	RSSI	Corr	CRC
00005	=426332368	10	0x17	0x73	0xF2	0xBF	-54	0x23	Bad

Frame	Time(us)	Len	MAC Frame Control					Seq	Dest	Dest	Encrypted Data	FCS		
	+696876128		Type	Sec	Pend	ACK	IPAN	Num	PAN	Addr	0x50	RSSI	Corr	CRC
00006	=1123208496	11	ACK	Y	Y	Y	Y	0x48	0x76DD	0xE17D	0x1B	-53	0x1B	Bad

Obrázek 16. Zachycená data pocházející z jiné sítě.

Nyní popíšu stejnou situaci z pohledu RFD.

Po zapnutí RFD:

```
**********************************
Microchip ZigBee(TM) Stack - v1.0-3.8
ZigBee RFD
Transceiver-CC2420

Trying to join network as a new device...
EA Error finding network.  Trying again...
```

RFD vypisoval stále nemožnost nalezení existující sítě. Tyto problémy (v anglické terminologii pojmenované jako NO_BEACON on RFD node) jsou způsobeny novou verzí ZigBee stacku 1.0-3.8. Na fóru Microchipu (http://forum.microchip.com/tm.aspx?m=226117) hlásilo tento problém více lidí.

Problém ve stacku se objevuje při použití CC2420 Transceiveru. Tato chyba postihuje pouze uzly RFD. Úprava kódu spočívala v odstranění jednoho příkazu MACDisable z MAC vrstvy, který odpojoval MAC vrstvu. RFD mohl posílat beacony, ale odpověď od koordinátora nebyla přijata, protože MAC vrstva byla odpojena. Takže výsledkem byl "no beacon".

Kód jsem upravil, přesto docházelo k vypisování podobné chyby, tentokrát s připojením k síti:

```
**********************************
Microchip ZigBee(TM) Stack - v1.0-3.8
ZigBee Coordinator
Transceiver-CC2420

Hardware initialized
ZigBee initialized

Trying to start network...
Error forming network.  Trying again...
```

Při dalším hledání na fóru firmy Microchip (http://forum.microchip.com/printable.aspx?m=199304) jsem narazil na poznámku, že se občas oběma typům zařízení nedaří provést inicializaci (tzn. zformovat síť v případě koordinátora a nalézt síť v případě RFD). Tento stav je nejspíš způsoben nedostatečně inicializovanou pamětí, ve které mohla zůstat data od posledního zapnutí přístroje. Nejen pro tyto případy je na ZigBee modulech tlačítko MCLR (memory clear, vymazání paměti).
Z výše uvedeného řešení vyplývá následující postup:
Při zapnutí koordinátora, pokud se mu nedaří zformovat síť, zmáčknout MCLR.
Při zapnutí RFD, pokud se mu nedaří nalézt síť, zmáčknout na více jak 1 sekundu MCLR. Pak již začne posílat rámce typu beacon a vyhledávat tak síť.

Pro úplnost uvádím kompletní průběh komunikace mezi oběma moduly včetně hlášení vypisovaných oběma zařízeními na sériový port.
Koordinátor vytvoří síť s ID 1006 a začne čekat na připojení dalších zařízení. Poté připojí nalezený RFD uzel s ID 796F (viz následující obrázky).

```
**********************************
Microchip ZigBee(TM) Stack - v1.0-3.8
ZigBee Coordinator
Transceiver-CC2420

Hardware initialized
ZigBee initialized

Trying to start network...

PAN 1006 started successfully.
Turning on joining...
Joining status changed.
Node 796F just joined.
```

Nyní ta samá situace z pohledu RFD. Zařízení hledá síť a snaží se připojit jako nové zařízení. Našlo síť s PANID 1006 a podařilo se mu k ní připojit. Ihned poté začne v pravidelných intervalech dotazovat svůj nadřazený uzel (tedy koordinátora), zda pro něj nemá nějaká data. Poté, co koordinátor odpoví, že žádná data nemá, RFD se „uloží ke spánku“ a šetří tak baterii. Proces se po nějakém čase opakuje.

```
**********************************
Microchip ZigBee(TM) Stack - v1.0-3.8
ZigBee RFD
Transceiver-CC2420

Trying to join network as a new device...

Network(s) found. Trying to join 1006.

Join successful!

Requesting data...
No data available.
Going to sleep...
Requesting data...
No data available.
Going to sleep...
```

Frame	Time(us)	Len	MAC Frame Control					Seq	Source	Source	SuperFrame Specification					
	+3424		Type	Sec	Pend	ACK	IPAN	Num	PAN	Addr	BO	SO	CAP	Batt	Coord	Assoc
00011	=2847907728	16	BCN	N	N	N	N	0x01	0x1006	0x0000	None	None	0xF	N	Y	Y

GTS Specification		PendAddr Spec		Beacon Payload				
Permit	Count	ExtAddr	ShortAddr	DevCap	Depth	RtrCap	NWKVer	St
N	0x0	0x0	0x0	Y	0x0	Y	0x1	

Obrázek 17. Koordinátor pošle odpověď na beacon rámec vyslaný koncovým zařízením.

V tomto speciálním beacon rámci koordinátor specifikuje informace o sobě a o spojení.

Frame	Time(us)	Len	MAC Frame Control					Seq	Dest	Dest	Source
	+120576		Type	Sec	Pend	ACK	IPAN	Num	PAN	Addr	PAN
00012	=2848028304	21	CMD	N	N	Y	N	0xD7	0x1006	0x0000	0xFFFF

Source Address	Association Request						FCS		
	Alloc	Sec	RxOn	Power	Dev	AltCoord	RSSI	Corr	CRC
0x0004A30000000067	Y	N	Off	Batt	END	N	-08	0x6B	OK

Frame	Time(us)	Len	MAC Frame Control					Seq	FCS		
	+1312		Type	Sec	Pend	ACK	IPAN	Num	RSSI	Corr	CRC
00013	=2848029616	5	ACK	N	Y	N	N	0xD7	-09	0x6A	OK

Obrázek 18. RFD posílá žádost o připojení k nalezené síti.

Frame	Time(us)	Len	MAC Frame Control					Seq	Dest
	+6464		Type	Sec	Pend	ACK	IPAN	Num	PAN
00016	=2848051408	27	CMD	N	N	Y	Y	0x02	0x1006

Destination Address	Source Address	Association Response		FCS		
		Status	Address	RSSI	Corr	CRC
0x0004A30000000067	0x0004A30000000054	Success	0x796F	-14	0x6C	OK

Frame	Time(us)	Len	MAC Frame Control					Seq	FCS		
	+1632		Type	Sec	Pend	ACK	IPAN	Num	RSSI	Corr	CRC
00017	=2848053040	5	ACK	N	N	N	N	0x02	-06	0x6B	OK

Obrázek 19. Koordinátor odpovídá na žádost o připojení.

Stav odpovědi je specifikován v poli Status (v tomto případě Success) a v poli Address je uvedena ID uzlu, kterého síť připojila (v tomto případě 0x796F).

Frame	Time(us)	Len	MAC Frame Control					Seq	Dest	Dest	Source		
	+854832		Type	Sec	Pend	ACK	IPAN	Num	PAN	Addr	Addr	Data Request	RSSI
00096	=13014544	12	CMD	N	N	Y	Y	0x38	0x1006	0x0000	0x796F		-08

Frame	Time(us)	Len	MAC Frame Control					Seq	FCS		
	+848		Type	Sec	Pend	ACK	IPAN	Num	RSSI	Corr	CRC
00097	=13015392	5	ACK	N	Y	N	N	0x38	-15	0x6A	OK

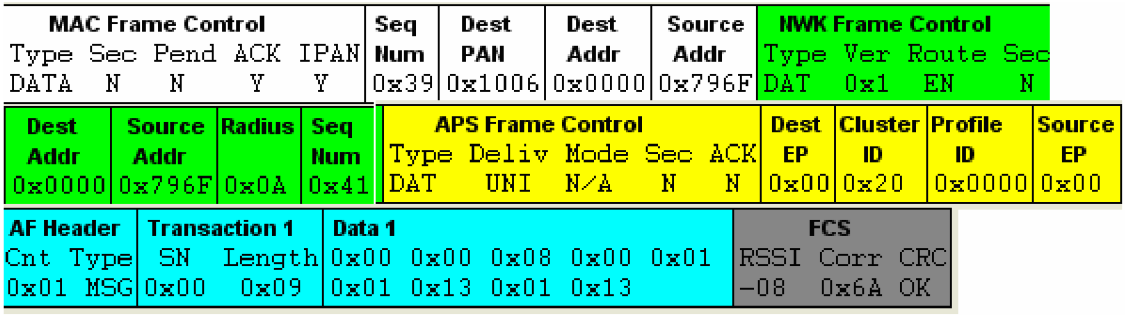
Frame	Time(us)	Len	MAC Frame Control					Seq	Dest	Dest	Source	FCS		
	+1984		Type	Sec	Pend	ACK	IPAN	Num	PAN	Addr	Addr	RSSI	Corr	CRC
00098	=13017376	11	DATA	N	N	N	Y	0x62	0x1006	0x796F	0x0000	-15	0x6A	OK

Obrázek 20. RFD pošle žádost o data a koordinátor mu odpoví.

Rámec 96: RFD (*Source Addr 796F*) posílá příkaz (*Type CMD*) koordinátorovi (*Dest Addr 0000*) se žádostí o data (*Data Request*).
Rámec 97 je potvrzení (Type ACK) vyslaného rámce se stejným sekvenčním číslem (Seq Num 38).
Rámec 98: Koordinátor (*Source Addr 0000*) odpovídá RFD (*Dest Addr 796F*) a vrací data (*Type DATA*), pokud pro něj nějaká má. V tomto případě je datová položka paketu prázdná, tedy žádná data nejsou k dispozici.

Následuje demonstrace párování koncových zařízení (RB5) a posílání zpráv diodám (RB4):

Po zmáčknutí tlačítka RB5 na RFD (Source Addr 796F) dojde k vyslání zprávy (Type DATA) ke koordinátorovi (Dest Addr 0000):

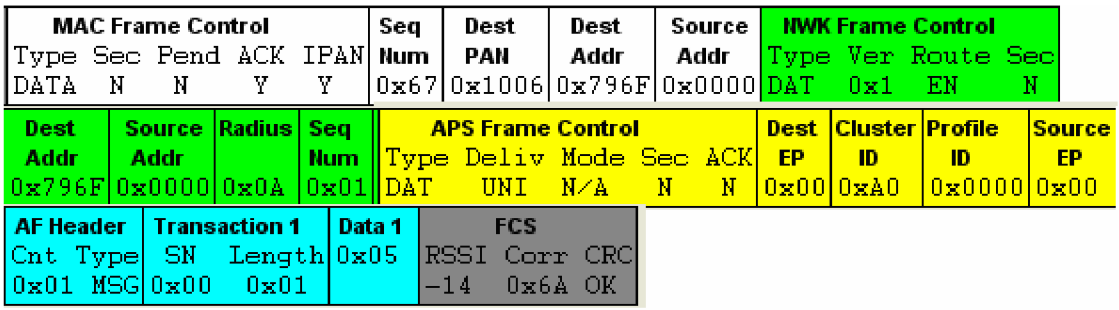


Obrýzek 21. Vyslání žádosti o připojení (bind request) od RFD ke koordinátorovi

a na terminálu se objeví:

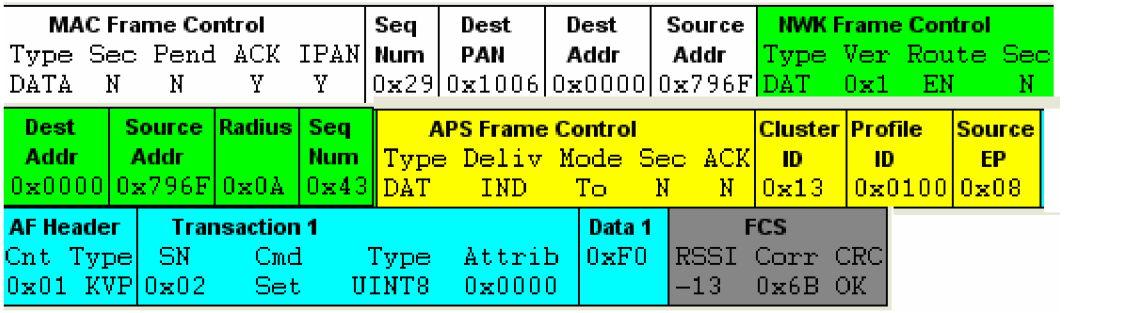
> Trying to send END_DEVICE_BIND_req.
> Message sent successfully.

A koordinátor na to odpoví:



Obrázek 22. Odpověď koordinátora na žádost o připojení

Po zmáčknutí tlačítka RB4 na RFD (Source Addr 796F) dojde k vyslání zprávy (Type DATA) ke koordinátorovi (Dest Addr 0000) s daty „prohoď stav diody“ (F0 – Toggling light):



Obrázek 23. Vyslání zprávy (projeví se změnou stavu diod) od RFD ke koordinátorovi

Na terminálu se zároveň objeví:

> Trying to send light switch message.
> Message sent successfully.

Po vypnutí koordinátora i RFD a následném zapnutí se RFD (Source Address 67) snaží zjistit, zda již nebyl dříve součástí nějaké sítě. Pokud ano (jako v tomto případě), tak vyšle příkaz (Type CMD) koordinátorovi s žádostí o znovu připojení „Orphan Notification“:

MAC Frame Control					Seq	Dest	Dest	Source	Source Address	Orphan	FCS		
Type	Sec	Pend	ACK	IPAN	Num	PAN	Addr	PAN		Notification	RSSI	Corr	CRC
CMD	N	N	N	N	0xD4	0xFFFF	0xFFFF	0xFFFF	0x0004A30000000067		-04	0x6A	OK

Obrázek 24. RFD žádá o znovupřipojení s odkazem na to, že byl již dříve součástí této sítě

Koordinátor (Source Address 54) mu odpoví, že se může připojit do jeho sítě s PANID 2786, na kanálu 0C (tedy 12):

MAC Frame Control					Seq	Dest	Destination Address	Source	Source Address
Type	Sec	Pend	ACK	IPAN	Num	PAN		PAN	
CMD	N	N	Y	N	0x4B	0xFFFF	0x0004A30000000067	0x2786	0x0004A30000000054

Coordinator Realignment				FCS		
PANID	ParentAddr	Channel	ShortAddr	RSSI	Corr	CRC
0x2786	0x0000	0x0C	0x796F	-15	0x6C	OK

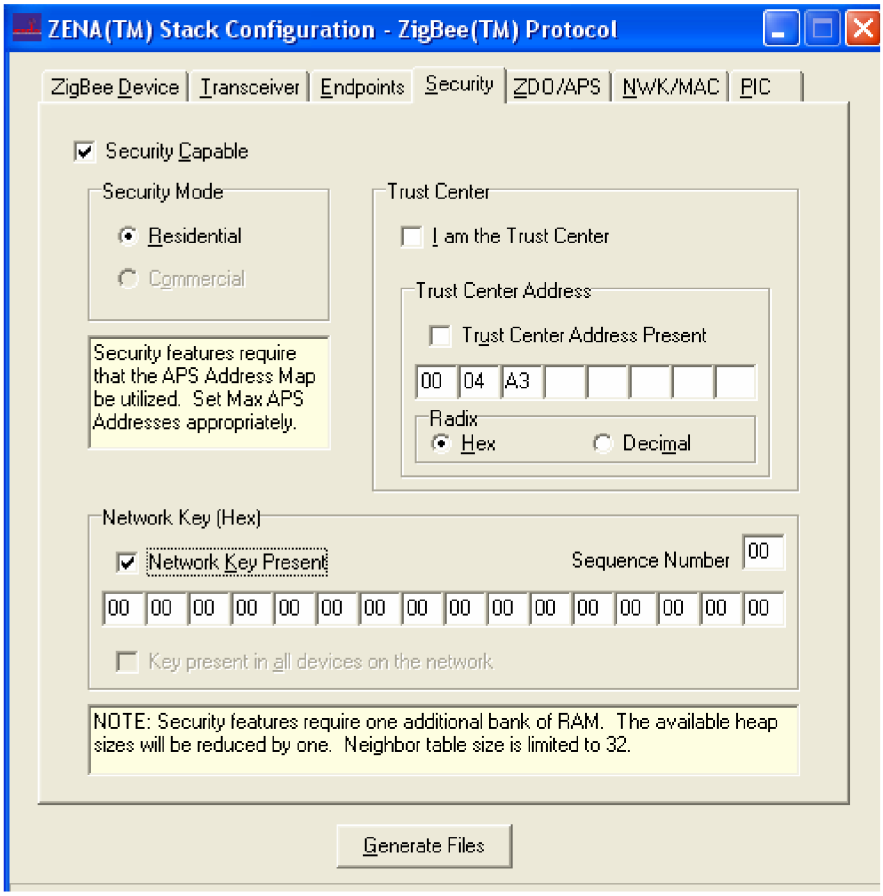
MAC Frame Control					Seq	FCS		
Type	Sec	Pend	ACK	IPAN	Num	RSSI	Corr	CRC
ACK	N	N	N	N	0x4B	-06	0x6A	OK

Obrázek 25. Koordinátor povoluje znovupřipojení koncového uzlu do své sítě

5.3.2 Zabezpečení aplikace

Klíčovým předpokladem pro implementaci bezpečnosti bylo podrobné pochopení práce programu. Musel jsem vysledovat co všechno se při provádění programu děje. K tomuto účelu jsem používal tzv. in-circuit debugger MPLAB ICD 2. Jde o programátor, kterým se do modulu jednak nahraje program a následně je možné jej debugovat, tedy sledovat posloupnost prováděných akcí. Vždy po stisku příslušného tlačítka procesor provede jeden výpočetní krok a program se posune k dalšímu nejbližšímu příkazu. Zároveň je možné si stanovit tzv. breakpointy, tedy místa, před kterými se provádění programu zastaví. Je tak možné začít sledovat program až od vybraného místa. Práce s debuggerem však byla, alespoň zpočátku, docela zdlouhavá. Breakpointy nelze libovolně vybrat (například v některých větvích či příkazech typu if), v jednu dobu mohou aktivní jen tři různé a každou chvíli se spouští vyřizování přerušení, které se při krokování vyřizuje poměrně dlouho. Některé situace jsou v ZigBee stacku ošetřeny tak, že jejich vyřízení musí proběhnout do předem stanoveného limitu (nastavuje se většinou v konfiguračním souboru *zigbee.def*) a pokud se tak nestane, dojde k odmítnutí paketu a případně i restartu celého stacku a tím i komunikace. Proto jsem musel některé hodnoty záměrně zvýšit, abych mohl provádět testování a ladění. Později jsem už znal kód natolik přesně, že jsem vystačil jen s jedním breakpointem, který jsem pak už přesouval na předem známá místa, kterými program prochází a před kterými jsem potřeboval ověřit stav proměnných. Z tohoto zkoumání vycházejí sekvenční diagramy komunikace, které jsem uvedl v příloze na konci práce.

Samotnou implementaci bezpečnosti jsem začal jejím povolením v konfiguračním souboru (*zigbee.def*). Konfigurační soubor je jedinečný pro každé zařízení, jsou zde nastaveny například MAC adresa zařízení a další parametry. Pro povolení bezpečnosti je potřeba definovat konstantu I_SUPPORT_SECURITY. Dále jsem zde pro zjednodušení přednastavil síťový klíč, aby se zařízení mohla jednoduše spárovat. K nastavování těchto parametrů se používá konfigurační nástroj ZENA Wireless Network Analyzer (viz obrázek 26), který jsem měl spolu s vývojovým kitem k dispozici. Tento nástroj podporuje bezpečnou komunikaci, ale pouze pro nově uvedený transceiver Microchipu MRF24J40. Mnou používaný transceiver Chipcon CC2420 sice hardwarově podporuje bezpečnostní operace na úrovni vrstvy MAC, ale pomocí ZENY bezpečnost povolit nelze. ZENA pak vygeneruje konfigurační soubor, podle kterého se budou překládat všechny soubory příslušného projektu. Můj postup byl takový, že jsem ručně dopsal části, které by ZENA vygenerovala ohledně bezpečnosti (jde například o specifikaci bezpečnostní úrovně nebo přednastavený síťový klíč), do konfiguračního souboru *zigbee.def*.



Obrázek 26. V konfiguračním nástroji Zena lze povolit bezpečnost, vybrat bezpečnostní mód (zatím je podporován jen rezidenční), nastavit MAC adresu trust centeru a nastavit přítomnost a hodnotu síťového klíče. Toto nastavení však není proveditelné pro transceiver CC2420.

Vlastní šifrování vyžaduje volání funkcí provádějící šifrování a dešifrování na příslušných vrstvách. Mimo to je ale potřeba v rámci standardu udělat změny v parametrech odchozích zpráv, a to zpravidla nastavení různých atributů v příslušných primitivách. Všechny provedené změny lze nalézt ve zdrojových souborech na přiloženém CD.

Musel jsem upravit stávající kód tak, aby platil i pro transceiver CC2420. Při kontrole typu čipu na začátku programu (aby se zjistilo, zda je schopný zabezpečné komunikace), jsem tedy doplnil

mezi možnosti podporovaných transceiveru i CC2420, aby program procházel požadovanou větev. Tato úprava ovšem předpokládá podobnost uspořádání obou čipů. Při překládání takto upraveného kódu však vyšlo najevo, že komunikace na fyzické úrovni je řešena odlišně a některé funkce použité při bezpečné komunikace transceiveru MRF24J40 zde chyběly. Také sada registrů (zejména v oblasti bezpečnosti) není podobná. Musel jsem doplnil některé funkce i do kódu pro CC2420. Při úpravách jsem odhalil několik nepřesností v kódu.

Například proměnná *currentAPSAddress* definovaná v základním ZigBee stacku vyžadovala podmínku *#if MAX_APS_ADDRESSES > 0*. Což znamená, že pokud je definováno více aplikačních adres, lze použít proměnnou *currentAPSAddress*. Při definici proměnné je tato podmínka dodržena, na mnoha dalších místech v kódu je však vložena pouze druhotná podmínka *#if defined(I_SUPPORT_SECURITY)*, takže pokud není vyžadována bezpečnost, na tyto nedostatky v kódu se nepřijde. Po upravení několika dalších podmínek (pro další proměnné), jsem narazil na problém fyzické nekompatibility obou transceiverů (MRF24J40 a CC2420). Příslušné soubory (zPHY_CC2420.c, zPHY_MRF24J40.c) se odlišují a proto některé potřebné proměnné a funkce jsem musel doplnit. Většinu úprav v kódu jsem okomentoval svým jménem, aby bylo zřejmé, kde a co jsem upravoval.

Po prvotních úpravách, přeložení a nahrání programu do modulů jsem spustil bezdrátový analyzátor a zachytával. Zde jsou výsledky:

```
**********************************
Microchip ZigBee(TM) Stack - v1.0-3.8
ZigBee Coordinator
Transceiver-CC2420

Trying to start network...
PAN 36AC started successfully.
Turning on joining...
Joining status changed.
Node 796F just joined.
secure join with Preconfigured Key
```

Frame	Time(us)	Len	MAC Frame Control					Seq	Dest	Dest	Beacon	FCS		
	+10496208		Type	Sec	Pend	ACK	IPAN	Num	PAN	Addr	Request	RSSI	Corr	CRC
00002	=16783760	10	CMD	N	N	N	N	0x00	0xFFFF	0xFFFF		-12	0x6B	OK

Obrázek 27. Koordinátor poslal rámec Beacon, kterým hledá dostupné sítě.

Frame	Time(us)	Len	MAC Frame Control					Seq	Dest	Dest	Beacon	FCS		
	+65934368		Type	Sec	Pend	ACK	IPAN	Num	PAN	Addr	Request	RSSI	Corr	CRC
00003	=82718128	10	CMD	N	N	N	N	0x40	0xFFFF	0xFFFF		-12	0x6B	OK

Obrázek 28. RFD poslal rámec Beacon, kterým hledá již vytvořené sítě.

Frame	Time(us)	Len	MAC Frame Control					Seq	Source	Source	SuperFrame Specification					
	+3584		Type	Sec	Pend	ACK	IPAN	Num	PAN	Addr	BO	SO	CAP	Batt	Coord	Assoc
00004	=82721712	16	BCN	N	N	N	N	0x01	0x36AC	0x0000	None	None	0xF	N	Y	Y

GTS Specification		PendAddr Spec		Beacon Payload				
Permit	Count	ExtAddr	ShortAddr	DevCap	Depth	RtrCap	NWKVer	St
N	0x0	0x0	0x0	Y	0x0	Y	0x1	

Obrázek 29. Koordinátor nenašel existující síť a tak vytvořil svoji síť s PAN ID 36AC.

Frame	Time(us)	Len	MAC Frame Control					Seq	Dest	Dest	Source	Source Address
	+119024		Type	Sec	Pend	ACK	IPAN	Num	PAN	Addr	PAN	
00005	=82840736	21	CMD	N	N	Y	N	0x41	0x36AC	0x0000	0xFFFF	0x0004A30000000067

Association Request						FCS		
Alloc	Sec	RxOn	Power	Dev	AltCoord	RSSI	Corr	CRC
Y	N	Off	Batt	END	N	-12	0x6B	OK

Frame	Time(us)	Len	MAC Frame Control					Seq	FCS		
	+1312		Type	Sec	Pend	ACK	IPAN	Num	RSSI	Corr	CRC
00006	=82842048	5	ACK	N	Y	N	N	0x41	-17	0x6B	OK

Obrázek 30. RFD poslal žádost o připojení k síti s Dest PAN 36AC.

Frame	Time(us)	Len	MAC Frame Control					Seq	Dest	Dest	Source Address	Data Request
	+12688		Type	Sec	Pend	ACK	IPAN	Num	PAN	Addr		
00007	=82854736	18	CMD	N	N	Y	Y	0x42	0x36AC	0x0000	0x0004A30000000067	

Frame	Time(us)	Len	MAC Frame Control					Seq	FCS		
	+1152		Type	Sec	Pend	ACK	IPAN	Num	RSSI	Corr	CRC
00008	=82855888	5	ACK	N	Y	N	N	0x42	-17	0x6C	OK

Obrázek 31. RFD poslal koordinátorovi žádost o data.

Frame	Time(us)	Len	MAC Frame Control					Seq	Dest	Destination Address
	+5904		Type	Sec	Pend	ACK	IPAN	Num	PAN	
00009	=82861792	27	CMD	N	N	Y	Y	0x02	0x36AC	0x0004A30000000067

Source Address	Association Response	
	Status	Address
0x0004A30000000054	Success	0x796F

rame	Time(us)	Len	MAC Frame Control					Seq	FCS		
	+1632		Type	Sec	Pend	ACK	IPAN	Num	RSSI	Corr	CRC
00010	=82863424	5	ACK	N	N	N	N	0x02	-12	0x6C	OK

Obrázek 32. Koordinátor poslal odpověď na žádost o připojení.

Frame	Time(us)	Len	MAC Frame Control					Seq	Dest	Dest	Source	Data Request
	+32208		Type	Sec	Pend	ACK	IPAN	Num	PAN	Addr	Addr	
00011	=82895632	12	CMD	N	N	Y	Y	0x43	0x36AC	0x0000	0x796F	

Frame	Time(us)	Len	MAC Frame Control					Seq	FCS		
	+832		Type	Sec	Pend	ACK	IPAN	Num	RSSI	Corr	CRC
00012	=82896464	5	ACK	N	Y	N	N	0x43	-17	0x6C	OK

Obrázek 33. RFD poslal další žádost o data.

Frame	Time(us)	Len	MAC Frame Control					Seq	Dest	Dest	Source	Encrypted Data	FCS		
	+4512		Type	Sec	Pend	ACK	IPAN	Num	PAN	Addr	Addr		RSSI	Corr	CRC
00013	=82900976	11	DATA	Y	N	N	Y	0x03	0x36AC	0x796F	0x0000		-17	0x6C	OK

Obrázek 34. Koordinátor odpověděl na žádost o data - zašifrovaně. Protože je datová část prázdná, odpověď znamená „žádná data“.

Frame	Time(us)	Len	MAC Frame Control					Seq	Dest	Dest	Source
	+1060928		Type	Sec	Pend	ACK	IPAN	Num	PAN	Addr	Addr
00014	=83961904	21	DATA	N	N	Y	Y	0x44	0x36AC	0x0000	0x796F

NWK Frame Control				Dest	Source	Radius	Seq	Leave	
Type	Ver	Route	Sec	Addr	Addr		Num	R/I	RemChild
CMD	0x1	SUP	N	0x0000	0x796F	0x01	0x8A	Ind	Y

rame	Time(us)	Len	MAC Frame Control					Seq	FCS		
	+1312		Type	Sec	Pend	ACK	IPAN	Num	RSSI	Corr	CRC
00015	=83963216	5	ACK	N	Y	N	N	0x44	-17	0x6B	OK

Obrázek 35. RFD (z adresy 796F) poslal příkaz typu LEAVE, protože nerozpoznal primitivu.

rame	Time(us)	Len	MAC Frame Control					Seq	Dest	Destination Address
	+38240		Type	Sec	Pend	ACK	IPAN	Num	PAN	
00016	=84001456	27	CMD	N	N	Y	N	0x45	0xFFFF	0x0004A30000000054

Source PAN	Source Address	Disassoc Notification Reason
0x36AC	0x0004A30000000067	0x02

rame	Time(us)	Len	MAC Frame Control					Seq	FCS		
	+1616		Type	Sec	Pend	ACK	IPAN	Num	RSSI	Corr	CRC
00017	=84003072	5	ACK	N	Y	N	N	0x45	-17	0x6A	OK

Obrázek 36. RFD pošle důvod odpojení (0x02) od sítě.

Tady je výpis z hyperterminálu:

> Requesting data...
> No data available.
> 3C Unhandled primitive.
> ZigBee Stack has been reset.

Zároveň došlo k resetu Zigbee stacku. Tato situace byla způsobena počátečním stavem kódu při vykonávání zabezpečeného režimu. Postupně jsem tedy musel doplňovat jednotlivé primitivy o bezpečnostní atributy, aby komunikace probíhala správně a nedocházelo k výše uvedenému stavu.

Implementace protokolu ZigBee od firmy Microchip podporuje všech sedm bezpečnostních režimů, které jsou ve specifikaci protokolu ZigBee na ochranu odchozích paketů (osmý *None* se nepočítá). Bezpečnostní režimy lze kategorizovat do třech skupin:

- Režim používající MIC (Message Integrity Code), který zajišťuje integritu paketu.
- Režim používající šifrování (ENC), který šifruje data a tak zajišťuje jejich důvěrnost.
- Kombinace obou režimů – samotná zpráva je zašifrována, integrita hlavičky a zprávy je chráněna MIC připojeným na konec paketu.

Použitý transceiver Chipcon CC2420 podporuje také všechny tyto režimy. Všechna další následující nastavení a postupy jsou hardwarově závislá a platí pro transceiver CC2420. K nastavení bezpečnostních parametrů slouží bezpečnostní registry SECCTRL0 (adresa 0x19) a SECCTRL1 (0x1A). Tyto registry se nastavují při inicializaci fyzické vrstvy ZigBee protokolu. V použité aplikaci se standardně nastavují na nezabezpečený režim. Při pokusu o změnu režimu při inicializaci docházelo k vysílání neplatných dat, takže jsem změny provedl až před samotným šifrováním.

V souboru zPHY_CC2420.c se ve funkci PHYinit inicializují registry transceiveru CC2420. Nastavování registrů (stejně tak i práce s pamětí) probíhá na úrovni fyzické vrstvy přes SPI rozhraní.

Pro práci s SPI sběrnicí jsou nadefinovány příkazy SPIPut a SPIGet. SPIPut vystaví bajt na sběrnici, SPIGet zase přečte vystavený bajt dat.

Zápis do registru probíhá tak, že se nejprve na sběrnici vystaví adresa požadovaného registru a poté se vystaví hodnota, kterou chceme zapsat do registru. Zde je příklad inicializace bezpečnostního registru:

```
SPIPut(REG_SECCTRL0);        // budeme zapisovat do bezpečnostního registru
SPIPut(0x01);          // zapíšeme MSB
SPIPut(0xC4);          // zapíšeme LSB
```

Význam jednotlivých bitů registru je podle tabulky na obrázku 37.

Čtení registru pak probíhá podobně, jen je potřeba při adresaci registru změnit jeden bit, který slouží pro výběr mezi zápisem nebo čtením do registru. To se provádí tak, že na hodnotu adresy aplikuje operace disjunkce s hodnotou 0x40. Tato operace zajistí nastavení sedmého bitu na 1, což je právě bit pro čtení registru. Příklad čtení výše zmíněného registru by tedy vypadal takto:

```
SPIPut(REG_SECCTRL0 | 0x40)          // budeme číst z bezpečnostního registru
MSB = SPIGet();          // přečteme MSB
LSB = SPIGet();          // přečteme LSB
```

SECCTRL0 (0x19) - Security Control Register

Bit	Field Name	Reset	R/W	Description
15:10	-	0	W0	Reserved, write as 0
9	RXFIFO_PROTECTION	1	R/W	Protection enable of the RXFIFO, see description in the RXFIFO overflow section on page 33. Should be cleared if MAC level security is not used or is implemented outside CC2420.
8	SEC_CBC_HEAD	1	R/W	Defines what to use for the first byte in CBC-MAC (does *not* apply to CBC-MAC part of CCM): 0 : Use the first data byte as the first byte into CBC-MAC 1 : Use the length of the data to be authenticated (calculated as (the packet length field – SEC_TXL – 2) for tx or using SEC_RXL for rx) as the first byte into CBC-MAC (before the first data byte). This bit should be set high for CBC-MAC 802.15.4 inline security.
7	SEC_SAKEYSEL	1	R/W	Stand Alone Key select 0 : Key 0 is used 1 : Key 1 is used
6	SEC_TXKEYSEL	1	R/W	TX Key select 0 : Key 0 is used 1 : Key 1 is used
5	SEC_RXKEYSEL	0	R/W	RX Key select 0 : Key 0 is used 1 : Key 1 is used
4:2	SEC_M[2:0]	1	R/W	Number of bytes in authentication field for CBC-MAC, encoded as (M-2)/2 0 : Reserved 1 : 4 2 : 6 3 : 8 4 : 10 5 : 12 6 : 14 7 : 16
1:0	SEC_MODE[1:0]	0	R/W	Security mode 0 : In-line security is disabled 1 : CBC-MAC 2 : CTR 3 : CCM

Obrázek 37. Možnosti nastavení bezpečnostního registru SECCTRL0 u transceiveru CC2420. (Zdroj [24])

Implicitní nastavení registru SECCTRL0 bylo v demonstrační aplikaci 000000 0 1 110 001 **00**. První dva **nejnižší** bity určují bezpečnostní režim, další tři bity určují počet bajtů vyhrazených pro počítání MAC hodnoty při režimu CBC-MAC, další tři bity jednotlivě určují výběr šifrovacího klíče (celkem lze zadat dva šifrovací klíče) pro samostatné šifrování nebo in-line šifrování ve vysílacím (TX, transmit) a přijímacím buferu (RX, receive). Bit 8 určuje co se bude autentizovat při počítání MIC v režimu CBC-MAC (jestli první bajt nebo délka datové části paketu). Bit 9 povoluje ochranu přijímací fronty (RX FIFO).

Mnou navrhované nastavení je 000000 1 1 000 001 **10**. Nastaven je režim CTR (tedy šifrování dat), MIC se nepočítá (tedy hodnota bitů 2-4 není důležitá), pro všechny operace je použit klíč 0 (ten jediný je zapsán do bezpečnostního buferu). Bit 8 se opět neuplatní a bit 9 je nastaven na 1 z důvodů kompatibility, jak je uvedeno v [24] na straně 33.

Transceiver CC2420 nabízí dva bezpečnostní mechanismy. In-line bezpečnostní mechanismus provádí bezpečnostní operace (šifrování, dešifrování a autentizace) na rámcích přímo ve vstupní (TXFIFO) a výstupní frontě (RXFIFO). Stand-alone bezpečnostní mechanismus provádí tyto operace na vyhrazeném místě v paměti, tzv. bezpečnostním buferu (SABUF).

Samotné zabezpečení dat pak probíhá následovně: Před vysíláním dat do paměti (buď do výstupní fronty nebo do bezpečnostního buferu) zapíšu délku hlavičky paketu, délku paketu, hlavičku paketu, pomocnou bezpečnostní hlavičku (viz obrázek 39) a nakonec samotná data. Výběr místa v paměti závisí na použitém bezpečnostním mechanismu. Dále do paměti zapíšu bezpečnostní klíč (na vyhrazené místo – tedy KEY0 nebo KEY1, CC2420 podporuje dva různé klíče) a kryptografické slovo. Protokol ZigBee specifikuje kryptografické slovo (tzv. *nonce*), které je složeno ze tří částí: čítače rámce, zdrojové adresy a sekvenčního čísla (pro vrstvu MAC) nebo bezpečnostního kontrolního bajtu (pro vrstvy NWK a APL). Nakonec pomocí bezpečnostních registrů (SECCTRL0 a SECCTRL1) nastavím požadovaný bezpečnostní režim (podle standardu ZigBee, viz obrázek 6).

Výběr bezpečnostního mechanismu (in-line nebo stand-alone) je dán zapsáním do příslušného stavového registru, čímž započne samotná šifrovací operace.

V případě in-line bezpečnosti je to registr STXENC (provedení bezpečnostních operací ve výstupní frontě TXFIFO), nebo STXON resp. STXONCCA (provedení bezpečnostních operací ve výstupní frontě TXFIFO a jejich následné odeslání). Podobné nastavení je i při příjmu dat, pro in-line bezpečnost jde o registr SRXDEC (první rámec v přijímací frontě RXFIFO bude dešifrován).

V případě stand-alone režimu je to pak registr SAES. Po zapsání do tohoto stavového registru se použije zvolený klíč v registru SECCTRL0 k zašifrování textu v bezpečnostním buferu SABUF v paměti. Po skončení šifrování je šifrovaný text zapsán zpět do bezpečnostního buferu, takže přepíše původní nezašifrovaná data.

Implementace ZigBee stacku od Microchipu podporuje pouze rezidentní režim (viz kapitolu 4.1.6.1), což znamená že pro celou síť existuje jeden síťový klíč, kterým se zabezpečují všechny pakety. Tento klíč lze jednak přednastavit, nebo vyměnit až za běhu. Klíč lze v demonstrační aplikaci přednastavit v konfiguračním souboru *zigbee.def* a pro účely šifrování je pak nutné jej nahrát do vyhrazeného místa v paměti transceiveru CC2420 (tedy na vyhrazené pozice KEY0 a KEY1, viz obrázek 38).

Na obrázku 38 je mapa paměti transceiveru CC2420. Informace lze do paměti zapisovat a číst buď pomocí proměnných v programu nebo na fyzické úrovni pomocí SPI sběrnice. Zápis na sběrnici zajišťuje funkce SPIPut, čtení zase SPIGet. Pro adresaci místa v paměti se posílají dva bajty. První se

pošle nejvýznamnější bit značící práci s pamětí (1) nebo registrem (0). Dalších 7 bitů indexuje místo v paměti. Druhý bajt se skládá z 2 bitů (B1:B0) určujících jeden ze tří paměťových banků (00 pro TXFIFO, 01 pro RXFIFO a 10 pro bezpečnostní bank) a jednoho bitu značícího zápisovou (0) nebo čtecí operaci (1). Po vystavení adresy lze poslat nebo číst požadovaná data. Další podrobnosti o přístup do paměti a informace o časování SPI sběrnice lze nalézt v [24].

Address	Byte Ordering	Name	Description
0x16F – 0x16C	-	-	Not used
0x16B – 0x16A	MSB LSB	SHORTADR	16-bit Short address, used for address recognition.
0x169 – 0x168	MSB LSB	PANID	16-bit PAN identifier, used for address recognition.
0x167 – 0x160	MSB LSB	IEEEADR	64-bit IEEE address of current node, used for address recognition.
0x15F – 0x150	MSB LSB	CBCSTATE	Temporary storage for CBC-MAC calculations
0x14F – 0x140	MSB (Flags) LSB	TXNONCE / TXCTR	Transmitter nonce for in-line authentication and transmitter counter for in-line encryption.
0x13F – 0x130	MSB LSB	KEY1	Encryption key 1
0x12F – 0x120	MSB LSB	SABUF	Stand-alone encryption buffer, for plaintext input and ciphertext output
0x11F – 0x110	MSB (Flags) LSB	RXNONCE / RXCTR	Receiver nonce for in-line authentication or receiver counter for in-line decryption.
0x10F – 0x100	MSB LSB	KEY0	Encryption key 0
0x0FF – 0x080	MSB LSB	RXFIFO	128 bytes receive FIFO
0x07F – 0x000	MSB LSB	TXFIFO	128 bytes transmit FIFO

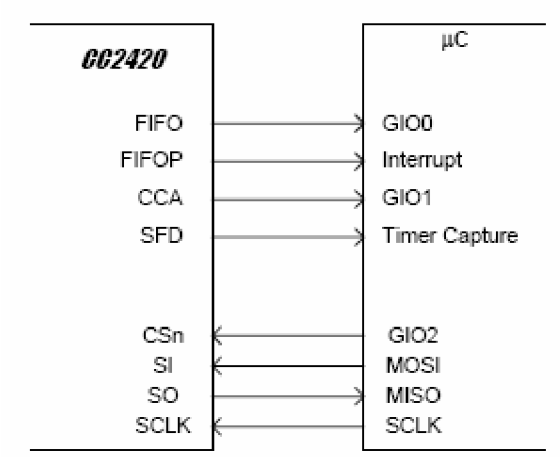
Table 6. *CC2420* RAM Memory Space

Obrázek 38. Paměťová mapa transceiveru CC2420. (Zdroj [24])

Problémy s implementací bezpečnosti v demonstrační aplikaci spočívaly v částečné nekompatibilitě obou čipů, zejména v oblasti registrů. Demonstrační aplikace je psaná pro transceiver MRF24J40 a tak některé registry přítomné v tomto transceiveru nejsou dostupné v CC2420.

Další komplikace představovaly přednastavené hodnoty, jejichž nevhodné nastavení vyplynulo až při podrobném krokování kódem. Příkladem je bezpečnostní režim - přednastaven byl režim ENC-MIC-32. Pro zjednodušení situace jsem MIC součet nepočítal a pracoval pouze s šifrováním, změnil jsem tedy režim na ENC.

Během implementace jsem narazil na problém s komunikací mezi transceiverem CC2420 a mikrokontrolerem po SPI sběrnici. Obecně používaný model komunikace, tedy pomocí proměnných v aplikaci, nebyl v tomto případě možný. Bylo nutné zapsat na předem určenou adresu v paměti, kde se následně provede šifrování a data se vyčtou ze stejného místa zpět. Podle technické podpory Microchipu, na kterou jsem se v této souvislosti obrátil, by bylo vhodné prověřit, že signály posílané transceiveru jsou podle specifikace, tzn. zda jsou data na sběrnici vystavena ve chvílích, kdy je transceiver čeká. To by vyžadovalo připojení logického analyzátoru nebo osciloskopu k pinům transceiveru připojeným na SPI sběrnici, tedy SI, SO a SCLK (viz obrázek 39).



Obrázek 39. Rozhraní transceiver – mikrokontroler, spodní čtyři piny tvoří rozhraní SPI sběrnice. (Zdroj [24])

Bezpečnost lze implementovat na třech vrstvách architektury – linkové (MAC), síťové (NWK), aplikační vrstvě (APL), záleží na požadavcích aplikačního profilu.

Implementace protokolu ZigBee přidává pomocnou bezpečnostní hlavičku před zabezpečenou zprávu každého zabezpečeného paketu v příslušné vrstvě. Formát pomocné bezpečnostní hlavičky ukazuje obrázek 40.

TABLE 11: ZigBee™ AUXILIARY SECURITY HEADER FORMAT

Security Location	Packet Header Feature			
	Security Control (1 Byte)	Frame Counter (4 Bytes)	Source Extended Address (8 Bytes)	Key Sequence Number (1 Byte)
MAC Layer Security		X		X
NWK Layer Security	X	X	X	X
APL Layer Security	X	X		X

Obrázek 40. Formát pomocné bezpečnostní hlavičky. (Zdroj [22])

Implementace je schopna zabezpečit sekvenční posloupnost (sequential freshness) hlídáním čítače přenesených rámců. Kontrolovány jsou pouze čítače paketů od „příbuzných“ uzlů sítě (tzn. v terminologii stromu – děti nebo rodiče), protože pouze členové „rodiny“ vědí, kdy se zařízení připojilo k síti. Pakety přijaté od členů rodiny, které nesplní požadavky na sekvenční posloupnost, jsou zahazovány.

Maximální délka přenášené zprávy je 127 bajtů. Při zapnuté bezpečnosti vzroste režie na přenášenou bezpečnostní hlavičku a kontrolní součet MIC o 5 až 29 bajtů, v závislosti na kombinaci bezpečnostního režimu (viz kapitolu 4.1) a zabezpečené vrstvy [22].

Na následujících obrázcích ukazuji odchycené pakety komunikace s bezpečnostními atributy.

MAC Frame Control					Seq	Dest	Dest	Source	Encrypted Data	FCS		
Type	Sec	Pend	ACK	IPAN	Num	PAN	Addr	Addr	0x04	RSSI	Corr	CRC
CMD	Y	N	Y	Y	0xB3	0x2412	0x0000	0x796F		-11	0x6B	OK

MAC Frame Control					Seq	FCS		
Type	Sec	Pend	ACK	IPAN	Num	RSSI	Corr	CRC
ACK	N	Y	N	N	0xB3	-28	0x6B	OK

Obrázek 41. RFD posílá zabezpečeně žádost o data (0x04 = data request)

MAC Frame Control					Seq Num	Dest PAN	Dest Addr	Source Addr	NWK Frame Control			
Type	Sec	Pend	ACK	IPAN					Type	Ver	Route	Sec
DATA	N	N	Y	Y	0x03	0x2412	0x796F	0x0000	DAT	0x1	SUP	Y

Dest Addr	Source Addr	Radius	Seq Num	Security Control			Frame Counter	Source Address
				ExtN	Key	Lvl		
0x796F	0x0000	0x0A	0x1C	Y	TC	1	0x00000105	0x0000000000000000

Encrypted Data												FCS		
0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x67	0x00	0x00	0x00	0x00	RSSI	Corr	CRC
0xA3	0x04	0x00	0x54	0x00	0x00	0x00	0x00	0xA3	0x04	0x00		-29	0x6B	OK

Obrázek 42. Koordinátor odpovídá na žádost o data – v paketu lze vysledovat použití rozšířené kryptografické výzvy (Ext N), čítače rámců a dat představujících MAC adresy obou zařízení.

MAC Frame Control					Seq Num	Dest PAN	Dest Addr	Source Addr	NWK Frame Control			
Type	Sec	Pend	ACK	IPAN					Type	Ver	Route	Sec
DATA	N	N	Y	Y	0xC3	0x2412	0x0000	0x796F	DAT	0x1	EN	Y

Dest Addr	Source Addr	Radius	Seq Num	Encrypted Data						FCS		
										RSSI	Corr	CRC
0x0000	0x796F	0x0A	0x15	0x34	0x13	0x00	0x01	0x08	0x11	-17	0x6B	OK

Obrázek 43. RFD zařízení posílá koordinátorovi zprávu.

6 Závěr

V této diplomové práci jsem se zabýval senzorovými sítěmi a způsoby jejich zabezpečení. O bezpečnost se zajímám již delší dobu a senzorové sítě jsou rychle se rozvíjející technologií, takže skloubení obou témat bylo důvodem mého výběru. Tato oblast je zatím málo prozkoumaná, neboť se jedná o poměrně mladou technologii.

Nejprve jsem zasadil problematiku senzorových sítí do širšího kontextu, vysvětlil důvod jejich zavedení a zdůraznil potřebu zabezpečení. Nastínil jsem základy fungování senzorových sítí, z hlediska jejich výstavby na standardu IEEE 802.15.4. Ve větší části práce jsem se věnoval bezpečnosti senzorových sítí obecně, jejich způsobu zabezpečení v závislosti na požadované aplikaci a velikosti sítě.

Poté jsem se zaměřil na významný standard senzorových sítí – ZigBee. Uvedl jsem základní principy, na kterých tento protokol pracuje, jeho strukturu, typy zařízení v síti a různé topologie sítě. Popsal jsem poskytované bezpečnostní mechanismy a bezpečnost na úrovni jednotlivých vrstev architektury standardu ZigBee. Protože je bezpečnost v ZigBee založena na bezpečnostních klíčích, věnoval jsem se i tomuto tématu. Ke konci jsem se zmínil i o některých otevřených bezpečnostních problémech senzorových sítí.

V praktické části práce jsem se nejprve seznamoval s vývojovým ZigBee kitem firmy Microchip a jejím ZigBee stackem. Podrobně jsem popsal a analyzoval fungování demonstrační aplikace, která postupně prochází jednotlivé stavy od nejnižší k nejvyšší vrstvě architektury a podle požadované služby nastavuje příslušné primitivy a jejich parametry.

Poté jsem se věnoval odzkoušení bezpečnostních funkcí ZigBee, které jsem provedl upravením jednotlivých vrstev aplikace a jejich připojením na bezpečnostní funkce hardwarových modulů. V demonstrační aplikaci jsem implementoval bezpečnostní prvky nutné pro zabezpečený přenos dat. Pracoval jsem s nejnovější verzí implementace ZigBee protokolu 1.0-3.8, která však není úplně doladěná, a proto jsem musel řešit vzniklé problémy. Pomocí síťového analyzátoru jsem sledoval komunikaci na úrovni jednotlivých paketů a získával tak zpětnou vazbu o fungování obou modulů.

V návaznosti na můj projekt se nabízí další náměty na jeho pokračování. Praktické uplatnění najde otestování zabezpečené komunikace více modulů. Při takové konfiguraci bude hrát větší roli autentizace. Dalším rozšířením tématu je vyzkoušení podpory pro transport klíčů. Zařízení bez přednastaveného klíče budou muset nejprve získat klíč od síťového koordinátora a teprve poté budou moci bezpečně komunikovat.

Současná verze ZigBee stacku (1.0-3.8) od Microchipu podporuje pouze rezidenční režim s jedním síťovým klíčem pro všechny bezpečnostní operace. Logickým námětem na další vývoj projektu je přidání podpory komerčního režimu s linkovými klíči, která přiblíží aplikaci senzorových sítí reálnému nasazení.

Seznam použitých zdrojů

[1] Pužmanová R.: *Bezpečnost bezdrátové komunikace*. Brno: Computer Press, 2005. ISBN 80-251-0791-4

[2] Koton J., Čika P., Křivánek V.: *Standard nízkorychlostní bezdrátové komunikace ZigBee* http://access.feld.cvut.cz/view.php?cisloclanku=2006032001 (květen 2007)

[3] Wikipedia. *Wireless sensor network* http://en.wikipedia.org/wiki/Wireless_sensor_network (květen 2007)

[4] Wikipedia. *ZigBee* http://cs.wikipedia.org/wiki/ZigBee (květen 2007)

[5] Galeev M.: *Home networking with Zigbee* http://www.embedded.com/showArticle.jhtml?articleID=18902431 (květen 2007)

[6] Römer, Kay, Friedemann Mattern (December 2004). *The Design Space of Wireless Sensor Networks*. IEEE Wireless Communications 11, str. 54-61. http://www.vs.inf.ethz.ch/publ/papers/wsn-designspace.pdf (květen 2007)

[7] ZigBee Alliance. *Zigbee Specification v1.0*, 2004 http://www.zigbee.org (květen 2007)

[8] Pužmanová R.: *Quo vadis, bezdrátová komunikace?* http://www.lupa.cz/clanky/quo-vadis-bezdratova-komunikace-3-3/ (květen 2007)

[9] Pužmanová R.: *ZigBee versus 802.15.4 - Osobní a domácí sítě*, část 10 http://telnet.cz/content_print.php?con_id=316 (květen 2007)

[10] Wireless Sensor Network Wiki http://wsn.oversigma.com/wiki/index.php/Main_Page (květen 2007)

[11] Hackmann G.: *802.15 Personal Area Networks* http://www.cs.wustl.edu/~jain/cse574-06/ftp/wpans/index.html (květen 2007)

[12] ZigBee Alliance. *ZigBee Security Specification Overview* 2005 http://www.zigbee.org/en/events/documents/December2005_Open_House_Presentations/ZigBee_Security_Layer_Technical_Overview.pdf (květen 2007)

[13] ZigBee Alliance. *ZigBee Security* http://www.zigbee.org/imwp/idms/popups/pop_download.asp?contentID=9436 (květen 2007)

[14] Geer D.: *Users Make a Beeline for ZigBee Sensor Technology* http://ieeexplore.ieee.org/iel5/2/33102/01556477.pdf?arnumber=1556477 (květen 2007)

[15] Zheng J., Lee M. J., Anshel M.: *Towards Secure Low Rate Wireless Personal Area Network* http://cs.kaist.ac.kr/~sjhong/Papers2006/K6.wireless%20personal%20LAN.Lee.pdf (květen 2007)

[16] Bundesamt für Sicherheit in der Informationstechnik. *Drahtlose Kommunikationssysteme und ihre Sicherheitsaspekte*
http://www.bsi.bund.de/literat/doc/drahtkom/drahtkom.pdf (květen 2007)

[17] Louderback J.: *ZigBee Ushers in Age of Connected Devices*
http://www.extremetech.com/article2/0,1558,1771993,00.asp (květen 2007)

[18] Bradáč, Z., Fiedler, P., Hynčica, O., Bradáč, F.: *Bezdrátový komunikační standard ZigBee*, Automatizace, 4, 2005, s.261-263
http://www.automatizace.cz/article.php?a=638 (květen 2007)

[19] ZigBee Alliance. *ZigBee Architecture Overview* 2006
http://www.zigbee.org/en/events/documents/Mar2006_Open_House_Presentations/ZigBee%20Architecture2.pdf (květen 2007)

[20] Yuxiang Y.: *ZigBee. IEEE 802.15.4*
http://www.sasase.ics.keio.ac.jp/jugyo/2005/zigbee.pdf (květen 2007)

[21] Vojáček A.: *ZigBee - novinka na poli bezdrátové komunikace*
http://www.hw.cz/Rozhrani/ART1299-ZigBee---novinka-na-poli-bezdratove-komunikace.html (květen 2007)

[22] Microchip Stack for the ZigBee Protocol AN965
http://ww1.microchip.com/downloads/en/AppNotes/00965c.pdf (květen 2007)

[23] Sastry N., Wagner D.: Security Considerations for IEEE 802.15.4 Networks
http://portal.acm.org/citation.cfm?id=1023654&coll=portal&dl=ACM&CFID=14103579&CFTOKEN=23482867 (květen 2007)

[24] Texas Instruments - Chipcon Products: CC2420, 2.4 GHz IEEE 802.15.4/ZigBee Ready RF Transceiver
http://www.chipcon.com/files/CC2420_Data_Sheet_1_4.pdf (květen 2007)

[25] Certicom – *Securing Sensor Networks*, March 2006
http://www.certicom.com/download/aid-575/WP-sensor-networks.pdf (květen 2007)

Seznam zkratek

ACK	Acknowledgment
ALC	Access Control List
AES	Advanced Encryption Standard
APL	Application Layer
APS	Application Sub-layer
DES	Data Encryption Standard
DOS	Denial of Service
DSSS	Direct Sequence Spread Spectrum
ENC	Encryption
FFD	Full Function Device
GHZ	Gigahertz
ICD	In-Circuit Debugger
ID	Identification, Identifier
IV	Initialization Vector
IEEE	Institute of Electrical and Electronics Engineers
ISM	Industry, Scientific, Medical
HDR	Header
MAC	Media Access Control, Message Authentication Code
MCLR	Memory Clear
MIC	Message Integrity Code
MHZ	Megahertz
NWK	Network Layer
PAN	Personal Area Network
PHY	Physical Layer
RFD	Reduced Function Device
RXFIFO	Receive FIFO
SABUF	Security Buffer
SAP	Service Access Point
SKKE	Symmetric-key Key Establishment
TXFIFO	Transmit FIFO
WPAN	Wireless Personal Area Network
ZED	ZigBee End Device
ZDO	ZigBee Device Object

Přílohy

A. Diagramy sekvence znázorňující ZigBee komunikaci
B. CD se zdrojovými texty, některými použitými zdroji a textem diplomové práce

A. Diagramy sekvence znázorňující ZigBee komunikaci

Komunikaci (viz obrázek nahoře) začíná vždy nejvyšší vrstva a jednotlivé primitivy se předávají nižším vrstvám. Vrstva MAC zahajuje vysílání a vrstva PHY je zodpovědná za příjem dat. Po přijetí odpovědi se zase zpráva dostává od nejnižší vrstvy k vyšším. Podle hodnoty přijaté primitivy pak aplikace rozhodne o dalším postupu.

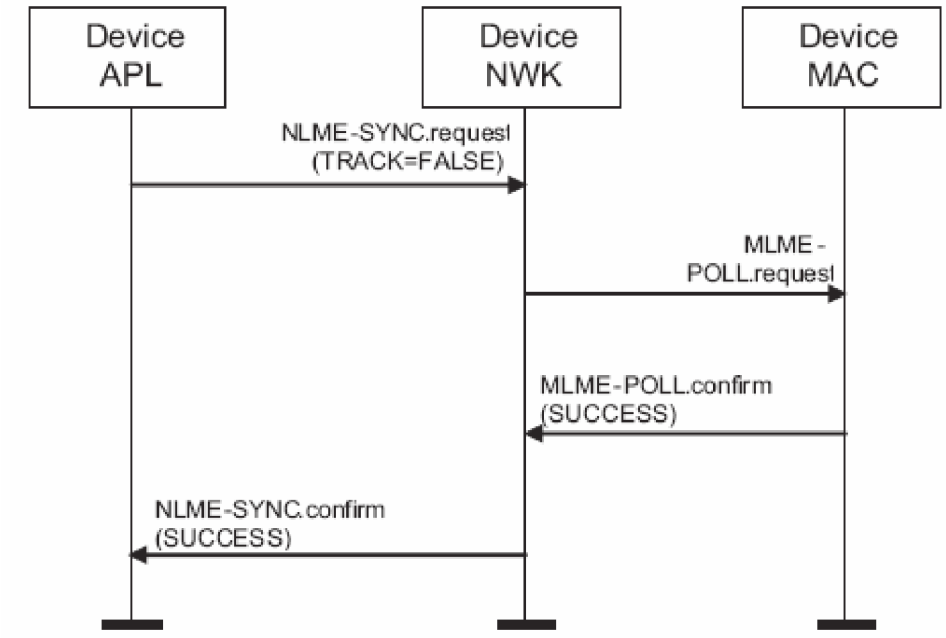
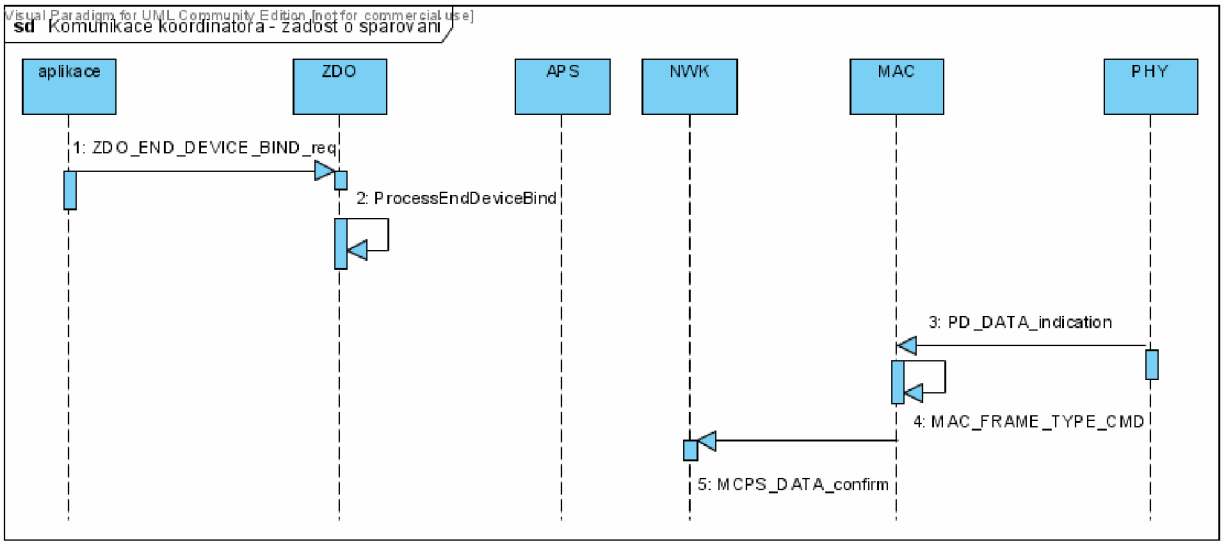
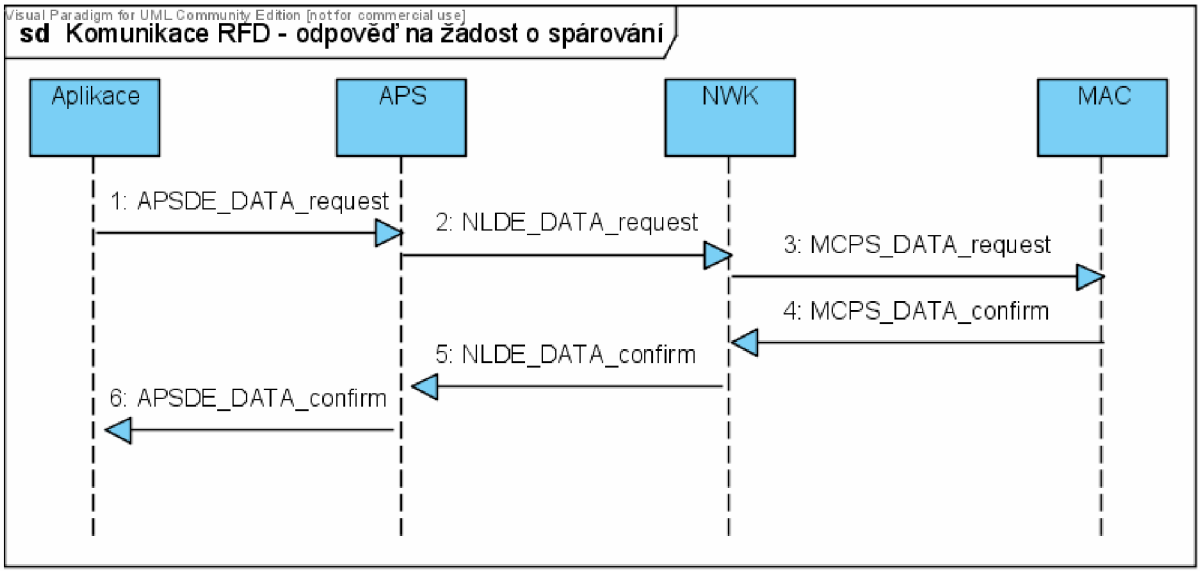


Diagram sekvence při synchronizaci, tedy pravidelnému dotazování koordinátora. (Zdroj [7])



Koordinátor přeposílá RFD žádost o spárování



RFD odpovídá na žádost o spárování