Západočeská univerzita v Plzni
Fakulta aplikovaných věd
Katedra informatiky a výpočetní techniky

Bakalářská práce

Útoky na bezdrátové senzorické sítě

Plzeň, 2014 Milan Široký

Prohlášení

Prohlašuji, že jsem bakalářskou práci vypracoval samostatně a výhradně s použitím citovaných pramenů.

V Plzni dne …… ……

The Abstract

The goal of this thesis is to divide attacks on sensor networks to groups by vulnerabilities which are being exploited. Those attacks are then sorted into layers based on OSI/ISO model. The second part of this thesis is dedicated to possibilities of stopping such attacks. In this part are therefore described possibilities and methods which allow protection of transferred data and also authentication of nodes in network. At the end of thesis summarizing specific methods for protection against selected attacks are stated.

Abstrakt

Cílem této práce je nejprve rozdělit útoky na bezdrátové senzorické sítě do skupin podle zranitelností, které využívají pro svoji činnost. Tyto útoky jsou poté rozděleny do vrstev podle OSI/ISO modelu. Druhá část práce se věnuje možnostem, jak těmto útokům zabránit. V této části jsou proto popsány možnosti a metody, které umožňují ochranu přenášených dat a také ověření pravosti uzlů v síti. Na konci práce jsou uvedeny přehledově konkrétní metody na obranu před třemi vybranými útoky.

Obsah

Seznam tabulek

1. Úvod

Bezdrátové senzorické sítě se používají v čím dál více odvětvích, od použití pro monitorování určité oblasti životního prostředí, přes řízení strojů až po použití v armádě pro zjišťování pohybu nepřátelských sil. V těchto prostředích se ale často stává, že dojde k poruše nebo ke zničení některých uzlů. Za tímto účelem je v každé síti velké množství uzlů, což nám zajistí funkčnost celé sítě i přes výpadek některých uzlů. Pokud se rozbije některý uzel na přenosové trase od zdrojového k cílovému uzlu, směrovací protokol si jednoduše vyhledá jinou trasu a přenos těchto dat půjde přes jiný uzel a chod sítě tedy nebude ovlivněn.

Jelikož se ale z jednotlivých uzlů přenáší jimi naměřená data do uzlů, která tato data shromažďují, musíme zabezpečit přenos těchto dat před nechtěnými změnami útočníkem nebo jen před chybou v přenosu. Přenos dat chráníme hlavně proto, že by nám mohl útočník poskytovat falešná data nebo by mohl zabránit přenosu jím vybraných dat.

V první kapitole je vysvětleno, co jsou vlastně zač bezdrátové senzorické sítě a jak fungují a z jakých částí se skládají. Dále jsou uvedeny jejich typologie a architektura těchto sítí. Jsou zde také popsány jejich specifické vlastnosti oproti jiným typům sítí.

V následující kapitole je uveden přehled útoků na bezdrátové senzorické sítě, který je ale rozdělen na jednotlivé vrstvy a v těchto vrstvách jsou útoky dále roztříděny do skupin podle zranitelností, které využívají pro svoji činnost.

Ve třetí kapitole jsou uvedeny možnosti, jak se těmto útokům bránit, případně jak alespoň zmírnit jejich škodlivou činnost v síti. Jako nejčastěji používanou metodou pro ochranu přenášených dat i pro ověřování pravosti jednotlivých uzlů je použití šifrování. Dále je zde uvedeno, že ne všechny šifrovací algoritmy jsou vhodné pro použití v senzorických sítí, zejména z důvodu jejich výpočetní náročnosti a velkých délek klíčů a potřebné režie. Uzly v senzorické síti mají totiž značně omezenou životnost zapříčiněnou malou kapacitou jejich baterie, kterou jsou napájeny, a také malým výpočetním výkonem. Z tohoto důvodu jsou zde uvedeny jak běžné symetrické a asymetrické šifry, tak také nově čím dál více používané eliptické křivky, které se používají zejména kvůli kratším klíčům oproti asymetrickým šifrám.

V poslední kapitole jsou již jen stručně uvedeny různé možnosti a metody, jak se účinně bránit proti třem vybraným útokům.

V závěru je shrnuta celá práce i s poznatky, které jsem získal při zpracování této práce.

2. Vysvětlení senzorických sítích

2.1. Použití

V dnešní době existuje mnoho druhů senzorických sítí vytvořených za účelem monitorování stavu životního prostředí[30]. Bezdrátové senzorické sítě jsou nejčastěji nasazovány pro měření seismické aktivity v oblastech s častým výskytem zemětřesení, pro měření aktuálního počasí a jeho další předpovědi, pro řízení dopravy ve městech, pro vědecká pozorování a předpovídání[31]. Dále se senzorické sítě používají v civilním životě při stavbě inteligentních domů, kde tyto senzory můžou regulovat teplotu v domě, zapínat nebo vypínat osvětlení domu podle toho, že senzory poznají, že je tma nebo světlo. Senzorické sítě se také používají ve firemním prostředí zejména ve výrobních podnicích pro shromažďování informací o jednotlivých strojích pro řízení těchto strojů v reálném čase. V armádách se senzorické sítě používají pro sledování pohybu nepřátel na bitevních polích[6].

Bezdrátové senzorické sítě se skládají z velkého počtu hustě rozmístěných senzorových uzlů, které společně spolupracují na sledování a měření hodnot ve specifických fyzických prostředích. Všechny senzory dohromady poskytují globální pohled na určité prostředí a nabízejí tak daleko větší množství informací než samostatně umístěné uzly, které poskytují jen lokální pohled na měřené prostředí[1, 30]. Senzorické sítě se postupem času vyvinuly z pasivních záznamových (logging) systémů, které vyžadovali ručně provést stažení jimi naměřených hodnot, až do inteligentních senzorových sítí obsahujících rozsáhlou síť automatických senzorových uzlů. Tyto sítě uzlů tvoří komunikační systém, který aktivně sdílí svá data se serverem senzorických sítí (SNS – Sensor Network Server), kde mohou být tato naměřená data dále spojována s dalšími soubory dat naměřených z jiných prostředí[30].

Pro sběr naměřených hodnot z uzlů nacházejících se v bezdrátových senzorických sítích (WSN – Wireless Sensor Networks) se běžně používají základnové stanice (base station) a agregační body (aggregation points). Agregační body se starají o shromažďování naměřených dat z okolních senzorů, spojování těchto dat z jednotlivých senzorů do větších celků a předávání těchto celků dat základnové stanici, která se stará o zpracování přijatých dat od agregačních bodů nebo o jejich přeposlání centru (processing center) pro další zpracování těchto naměřených dat[6].

2.2. Specifické vlastnosti

Bezdrátové senzorické sítě disponují několika specifickými vlastnostmi, které je odlišují od ostatních bezdrátových sítí, jako je třeba bezdrátový Internet. Na tyto charakteristické vlastnosti, uvedené v seznamu níže, musíme dávat pozor a brát je v úvahu při vytváření a navrhování nových protokolů a algoritmů použitelných v senzorických sítích[6].

- Senzory mají značně omezené zdroje energie, paměti a výpočetní kapacity. Z toho důvodu jsou upřednostňovány protokoly a algoritmy, které co nejméně zatěžují jednotlivé uzly (mají malé nároky na energii, na paměť a snižují spotřebu energie při měření a odesílání hodnot z uzlu do sítě), čímž je možné dosáhnout delší životnosti těchto senzorů[6].
- Senzory mají omezenou spolehlivost částečně způsobenou jejich omezenými zdroji[6].
- Bezdrátové senzorické sítě nejčastěji používají dynamicky se měnící topologie sítí. To znamená, že nové uzly mohou být snadno přidány do již existující sítě nebo aktivovány (uzly se již nachází v síti, ale zatím nejsou používány) a připojeny do sítě. Také je zde možnost snadno odebrat senzory, které již dosáhly nebo brzo dosáhnou své hranice spolehlivosti[6].
- Bezdrátové senzorické sítě mohou obsahovat velké množství senzorů[6].
- Bezdrátové senzorické sítě jsou nejčastěji centralizované z hlediska zpracování dat poskytnutých od uzlů a občasného řízení základnovou stanicí. Naměřená data předávají uzly do několika agregačních bodů, které tato přijatá data dále přeposílají do základnových stanic, kterých je v síti menší počet než agregačních bodů. Základnové stanice mohou také někdy vysílat (broadcast) všem uzlům v síti dotazy nebo řídící informace[6].

2.3. Typologie sítí

Při vývoji nových bezdrátových senzorických sítí byly použity již existující topologie sítí, ale musely být upraveny pro specifické potřeby senzorických sítí. Mezi tyto potřeby patří zejména snižování nákladů na provoz těchto uzlů, snižování jejich energetické náročnosti z důvodů omezení jejich napájecích zdrojů. Zároveň s těmito požadavky na senzorické sítě je zde také kladen důraz na zvyšování jejich celkové spolehlivosti.

Mezi základní topologie bezdrátových senzorických sítí patří:[32]

- Síť typu peer-to-peer (uzel k uzlu) umožňuje každému uzlu přímou komunikaci s jiným uzlem bez nutnosti, aby tato komunikace procházela přes centralizovaný komunikační rozbočovač (hub). Každé zařízení v této síti může zastávat obě role, jako klient (kdy uzel odesílá data do jiného uzlu) a jako server (uzel přijímá data odeslaná jiným uzlem)[32].
- V sítích typu hvězda (star) jsou všechny uzly propojeny s centralizovaným komunikačním rozbočovačem (hub). Žádný z uzlů nemůže komunikovat přímo s jiným uzlem, ale veškerá síťová komunikace musí procházet přes centralizovaný rozbočovač. Ten obdrží zprávu od jednoho uzlu a dále ji přesměruje a odešle ke druhému uzlu, kterému chtěl první uzel odeslat zprávu. Každý uzel v síti je označován jako klient. Server, který vyřizuje a přeposílá veškeré zprávy, mířící na jiné uzly, je označován jako server[32].
- Síťová topologie stromového (tree) tvaru se vyznačuje tím, že používá hlavní komunikační směrovač označovaný jako hlavní (kořenový) uzel (root node). O jednu úroveň níže pod hlavním uzlem se nachází centralizovaný rozbočovač, pod kterým se již vyskytuje normální topologie typu hvězda, kde všechny uzly komunikují pouze s centralizovaným rozbočovačem, který tyto obdržené zprávy předává vyššímu patru a to hlavnímu uzlu. Tato typologie typu strom (tree) bývá označována jako hybrid typologií uzel k uzlu (peer-to-peer) a hvězdy (star). Hlavní uzel komunikuje s ostatními centralizovanými rozbočovači přímo a tedy stejně jako v typologii uzel k uzlu (peer to peer). Centralizovaný rozbočovač potom dále komunikuje, stejně jako v topologii hvězda (star), se všemi uzly nacházejících se pod ním[32].
- Sítě typu mesh dovolují uzlům přenášet data přes jednotlivé uzly až k cíli. Tato skutečnost umožňuje, aby síť typu mesh mohla být samo opravná, tedy v případě výpadku jednoho uzlu se budou data přenášet přes jiný uzel až k jejich cíli. V této síti může každý uzel komunikovat s každým uzlem v síti. Data jsou přenášena postupně přes více uzlů do té doby, než dorazí do zamýšleného cílového uzlu. Správně zavést a zprovoznit síť typu mesh může být velmi finančně náročné z důvodu velké složitosti této sítě[32].

Normální fungování sítě se pokoušejí narušit aktivní a pasivní útoky. Aktivní útoky se snaží měnit nebo dokonce ničit data přenášená v rámci sítě. Můžeme je rozdělit do dvou skupin. Jednu skupinu tvoří externí útoky, prováděné z uzlů nacházejících se mimo síť. Druhou skupinou jsou vnitřní útoky kompromitovaných uzlů, umístěných uvnitř sítě. Útoky z vnitřku sítě je těžké rozpoznat. Externí útoky se detekují snadněji. Mezi aktivní útoky můžeme například zařadit útok na předstírání jiné identity (impersonation attack), útok na modifikaci dat v síti (modification attack) a útok na replikaci paketů (replication attack)[9].

Při pasivních útocích útočník nemění žádná data přenášená v síti ani neruší síťový provoz, ale jen odposlouchává přenášená data. Protože nedochází k poškozování přenášených dat, je velmi obtížné pasivní útoky detekovat. Mezi řešení problému s náročnou detekcí přítomnosti útočníka v síti patří používání silných šifrovacích algoritmů pro šifrování veškerého přenášeného obsahu. Zašifrováním přenášených dat docílíme toho, že útočník nezíská žádné užitečné informace z jím zachycených dat[9].

2.4. Architektura senzorických sítí

- Fyzická vrstva je zodpovědná za přenos informací přenosovým kanálem prostřednictvím drátových nebo bezdrátových sítí. Dále provádí detekci signálu, modulaci signálu, dekódování signálu a vybírá frekvenci přenosu. Proto je fyzická vrstva základním kamenem každé sítě. Na této vrstvě mohou útočníci provádět útoky na zničení požadovaného uzlu nebo na odposlouchávání síťového provozu[6].
- Linková vrstva nese odpovědnost za multiplexování datových toků, za detekci datových rámců, kontrolu chyb, zajištění spolehlivost spojení uzlu s uzlem (point-point) nebo uzlu s více uzly (point-multipoint)[33]. MAC (Medium Access Control – podvrstva linkové vrstvy) vrstva je zodpovědná za politiku přístupu ke kanálům (Channel access policies), řízení vyrovnávací paměti a za kontrolu chyb v přenosu. V bezdrátových senzorických sítích musíme vzít v úvahu protokoly MAC vrstvy pro zachování energetické účinnosti, spolehlivosti, nízké přístupové doby a vysoké propustnosti[33].
- Síťová vrstva má za úkol zejména směrování mezi uzly v síti. Mezi největší výzvy této vrstvy patří šetření energie v jednotlivých uzlech, omezená paměť a vyrovnávací paměť (buffer) a také to, že senzory nevlastní žádné globální ID (identifikátor), kvůli čemuž musí být organizovány samostatně. Nepřítomností globálního ID se uzly odlišují od počítačových sítích, kde má každý počítač svoji IP adresu a centrální zařízení pro řízení komunikace mezi nimi[33].
- Transportní vrstva má úkol zajistit spolehlivý přenos síťových paketů, které jsou přenášeny opakovaně, v sítích, kde jsou tato data přenášena z jednoho uzlu na druhý (end-to-end) a dále by měla transportní vrstva snížit nebo zabránit přetížení sítě, pokud bude procházet veliký datový tok skrze směrovače nacházející se v síti[34].
- Aplikační vrstva nese odpovědnost za řízení síťového provozu a také poskytuje příslušný software pro různé aplikace, které provádějí překlad dat do srozumitelné podoby nebo posílají dotazy na zjištění určitých informací[34].

3. Rozdělení útoků na bezdrátové senzorické sítě

V následující kapitole budou rozděleny útoky na senzorické sítě podle toho, na kterou vrstvu se zaměřují při útočení. Dále budou popsána slabá místa směrovacích protokolů a způsoby, jak tyto útoky ovlivňují činnost uzlů. Tyto útoky budou rozděleny do skupin podle toho, jakou využívají zranitelnost pro provedení útoku. U každé skupiny útoků bude uveden návrh, jak zabránit těmto útokům v jejich činnosti.

3.1. Fyzická vrstva

3.1.1. Útoky ničící jednotlivé uzly

Útoky zaměřené na fyzické zničení samotných uzlů nebo klíčových uzlů v síti (základnové stanice a agregační uzly). V dnešní době již nejsou tak účinné kvůli bezdrátovému spojení mezi uzly. I když by útočník zničil několik uzlů, tak by stále nedokázal zničit celou síť.

Do této skupiny patří zejména útok, při kterém dochází k neoprávněnému poškození uzlů, které v průběhu tohoto útoku zničíme nebo modifikujeme podle svých představ. Tímto krokem zastavíme nebo upravíme služby běžící na těchto uzlech. Poškození sítě může být ještě větší, jestliže dojde k napadení základnových stanic nebo bodů, které se starají o rozdělení prostředků v síti. Dopad celého útoku může být velmi rozsáhlý z toho důvodu, že je prováděn na zdrojové (starají se o zpracování přijatých dat od agregačních uzlů) nebo agregační uzly (starají se o sběr dat naměřených uzly v jejich okolí), které mají značnou zodpovědnost za fungování celé sítě, komunikaci v ní a za zpracování dat v síti[6].

Díky tomu, že dnešní senzorické sítě (Wireless Sensor Networks) pracují s velikou mírou redundance, je značně omezena efektivita útoků na samotné senzory. I když je napadeno a kompromitováno značné množství uzlů, tak ale stejně útočník nedokáže zničit nebo zastavit síť[6].

3.1.2. Útoky zaměřené na zachytávání síťového provozu

Do této skupiny patří, kromě níže uvedeného útoku na odposlech (Eavesdropping), také útok na zjištění umístění uzlu v síti (Location disclosure attack), který se nachází v síťové vrstvě, dále sem patří útok manipulování síťového provozu (Traffic Manipulation), který je spolu s útokem na falšování adresy útočníkova uzlu (Identify Spoofing) uveden v linkové vrstvě.

Útočníci při tomto útoku sledují probíhající síťový provoz na komunikačních kanálech a ukládají si získaná data pro pozdější analýzu a získání citlivých údajů. Tento útok, zaměřený na odposlech dat, probíhá bez vědomí odesílatelů a příjemců a je ho proto velmi obtížné detekovat[6].

Dnešní bezdrátové senzorické sítě jsou obzvláště náchylné na tento typ útoků kvůli tomu, že nelze mít mezi uzly natažené kabely z důvodu častého přidávání a odebírání uzlů ze sítě (wirelees transmission). Rádiové signály jsou vysílány (broadcast) volně do prostoru a dají se snadno odposlechnout díky tomu, že jsou všem volně dostupné. Útočník může jednoduše odposlechnout rádiový přenos dat a získat tak surová data, pokud se nachází v dosahu vysílaného signálu odesílatelem dat[6].

3.1.3. Útoky na rušení jednotlivých uzlů

Útočník použije dostatečně výkonný vysílač pro rušení bezdrátového přenosu dat. Nejedná se přímo o útok, při kterém chce útočník získat data nebo se připojit do sítě, ale spíše jen o zahlcení nebo rušení přenosu.

V tomto útoku jde v první řadě o zablokování nebo rušení signálu, což vede k porušeným nebo ztraceným zprávám. Signál generovaný útočníkem může být silnější než signál vysílaný cílovým uzlem a dokáže tento vysílaný signál zahltit a rušit tak celou komunikaci. Pro úspěšné provedení útoku na rušení síťového provozu musí mít útočník dostatečně výkonný vysílač. Mezi nejčastější typy rušení přenášeného signálu patří zejména vysílání šumu nebo pulzů. Zařízení potřebné pro rušení signálů je v dnešní době již snadno dostupné[7].

3.1.4. Obrana proti těmto útokům

Útoky, jejichž cílem je přímo zničit konkrétní uzel nebo třeba základnovou stanic, aby už nebyly schopny fungovat a přenášet data v síti, se použitím šifrování vůbec neovlivní, protože těmto útokům nejde vůbec o to, aby získali nějaké informace z přenášených dat. Těmto útokům se ale dá celkem úspěšně bránit zavedením redundance uzlů v síti, díky čemuž zničení jednoho uzlu neohrozí fungování celé sítě.

Do této skupiny útoků patří zejména útok na neoprávněné poškození uzlů (Device Tampering).

Při útoku, na rušení síťového provozu, nám zavedení šifrování nijak nepomůže zabránit tomuto útoku. Protože při tomto útoku je do sítě útočníkem schválně vysílán šum nebo signálové pulzy. Aby se tento útok útočníkovi podařil, tak musí disponovat výkonným vysílačem, aby dokázal zahltit nebo zablokovat vysílání v síti. Tomuto útoku se dá zabránit použitím přepínání frekvencí podle předem daného schématu. Výstupní signál je tedy modulován řadou rádiových frekvencí a jeví se útočníkovi jen jako šum v přenosovém kanále.

Sem můžeme zařadit útok na rušení síťového provozu (Jamming).

3.2. Linková vrstva

3.2.1. Odposlech přenášených dat a manipulace se síťovým provozem

Manipulování síťového provozu (Traffic Manipulation)

Útočník může velice snadno zmanipulovat bezdrátovou komunikaci ve WSNs (Wireless Sensor Networks) nebo i v ostatních bezdrátových sítích prostřednictvím MAC vrstvy. Ve chvíli, kdy vysílá správný uzel svá data, začne útočník vysílat své pakety za účelem způsobení nadměrných kolizí v síti. Čas, ve kterém by měl útočník začít vysílat, si snadno zjistí poslouchánímprovozu na přenosovém kanálu. Výrazné zvýšení počtů konfliktů v síti může vést ke zhoršení kvality signálů nebo snížení dostupnosti sítě, čímž útočník znatelně sníží celkovou propustnost sítě. Útočník záměrně porušuje pravidla koordinace přenosů a snaží se získat co největší šířku pásma pro sebe. Jelikož tyto pravidla porušuje v MAC schématu, kde je kladen velký důraz na koordinovaný přenos paketů, tak tím, že neuposlechne pravidla koordinace, může pro sebe získat větší část pásma na úkor ostatních uzlů. Použitím těchto způsobů je nakonec snížen výkon celé sítě a je docíleno omezení celého síťového provozu kvůli kolizím a nespravedlivě rozdělenému pásmu[6]

Falšování adresy útočníkova uzlu (Identify Spoofing)

Další útok v MAC (Media Access Control) vrstvě je MAC falšování adresy. Tato adresa se běžně používá jako unikátní identifikátor každého uzlu v síti. Vzhledem k povaze vysílání bezdrátových sítích do širokého okolí (broadcast) je proto pro útočníka velmi snadné sledovat komunikaci a najít v ní MAC adresy ostatních uzlů. Pokud nebude vysílání zabezpečené, může útočník podvrhnout svoji adresu za adresu některého jiného uzlu a vydávat se tak za jiný uzel.

Jedním z nejznámějších útoků na falšování MAC identity (spoofing) je Sybil útok (Sybil attack), při kterém se útočník vydává za více identit najednou[8].

Útočník může pro získání přístupu do sítě zfalšovat nejen normální uzel, ale dokonce i základní stanici (base station) nebo agregační bod (aggregation point). To mu pomůže získat práva nebo prostředky bezdrátové sítě, které by jinak za normálních okolností nezískal. Když tento útok úspěšně dokončí, může ovládnout celou síť a získat nad ní kontrolu. Útoky na zfalšování identity (spoofing) se často používají jako první krok při napadení sítě. Tyto útoky vytvářejí podmínky pro realizaci dalších útoků (cross-layer attacks), které už dokáží značně ohrozit síť nebo získat citlivá data z uzlů. Sybil útok (Sybil attack) například může poskytnout útočníkovi citlivé informace nebo poskytnout síti falešné informace pro směrování mezi uzly a tím zahájit útok na směrování (false routing attack)[8].

3.2.2. Obrana proti těmto útokům

Do následující skupiny patří útoky, které využívají toho, že uzly si předávají informace bezdrátově a díky tomu můžou útočníci odposlechnout tyto přenášená data.

Proti těmto útokům se lze účinně bránit právě zavedením šifrování přenášených dat, čímž tyto útoky přestanou být efektivní.

Je ale také možné použít nahrazení vstupních bitů za sekvenci bitů podle daného schématu (DSSS). Tyto útoky se snaží zejména analyzovat přenášená data, zjišťovat identitu uzlů nebo jejich umístění v síti.

Do této skupiny můžeme zařadit tyto útoky: Útok na odposlouchávání síťového provozu (Eavesdropping) je popsán v části zaměřující se na fyzickou vrstvu. Útok na zjištění umístění uzlu v síti (Location disclosure attack) je uveden v síťové vrstvě. Manipulování síťového provozu (Traffic Manipulation) a falšování adresy útočníkova uzlu (Identify Spoofing) patří do linkové vrstvy.

3.3. Síťová vrstva

3.3.1. Útoky na chyby směrovacích algoritmů

Útok na zjištění umístění uzlu v síti (Location disclosure attack)

Tento útok je pouze jednou z částí dalšího útoku na zjištění potřebných informací (information disclosure). Škodlivý uzel při tomto útoku získává potřebné informace o umístění a struktuře sítě pro svůj další útok. V další části útoku se sbírají informace o umístění jednotlivých uzlů a vytváří se z těchto informací mapy podobné silničním mapám. Z těchto map se zjistí poloha těch uzlů, které se nacházejí na trase zajímající útočníka. V případě analýzy provozu v síti MANET se jedná o dosud nevyřešené bezpečnostní útoky proti těmto sítím[9].

Útočníci se snaží odhalit identitu obou komunikujících stran a analyzovat síťový provoz za účelem naučení se vzorce, podle kterého probíhá síťový provoz, a sledování změn v tomto vzorci provozu. Jestliže dojde k úniku těchto informací, tak může být vážně narušena bezpečnost citlivých scénářů[28]. Zprávy o směrování jsou úmyslně rozesílány s malým počtem přeskoků mezi uzly (hop-limit hodnot), aby zařízení zpátky posílala chybové zprávy ICMP a útočník je mohl zaznamenat. Po skončení tohoto procesu zná již útočník uzly, které se nacházejí na trase k cílovému uzlu. Jestliže útočník již zná umístění mezilehlých uzlů, tak stejně dobře může zjistit polohu jím požadovaného uzlu[13].

Další útoky využívají vlastností aktivních směrovacích algoritmů (vyhledávají periodicky trasy a zjišťují směrovací údaje), které si ukládají veškeré obdržené směrovací informace. Toho využije útočník a začne do sítě vysílat velké množství směrovacích informací o trasách mezi neexistujícími uzly a správnými uzly v síti. Nebo může útočník do sítě vysílat také falešné směrovací informace, které se také díky aktivním algoritmům budou ukládat do tabulek uzlů a způsobí chaos v síti, protože data půjdou po špatných nebo neexistujících trasách.

Útok na přetečení směrovací tabulky (Routing table overflow attack)

Jestliže se uzlu zaplní celá směrovací tabulka falešnými zprávami od útočníka, dojde k tomu, že ji uzel nebude moci normálně používat a ukládat si do ní nově obdržené směrovací informace.

Využitím toho, že uzel po zaplnění jeho tabulky přestane přijímat nové zprávy, začnou vysílat do sítě velké množství prázdných směrovacích informací. Vyslané informace od útočníka do sítě mohou nakonec zaplnit celou směrovací tabulku v normálním uzlu až do takové míry, že dojde k přetečení[6].

Tento typ útoků se nejčastěji vyskytuje v síti, která používá aktivní směrovací algoritmy, které zjišťují periodicky směrovací údaje[8]. Pro spuštění tohoto útoku se útočník snaží vytvořit dostatečné množství cest od neexistujících uzlů ke správným uzlům v síti. Útočníkovým záměrem je poslat velké množství falešných zpráv, aby docílil zaplnění směrovací tabulky vybraného uzlu.

Pokud se v tabulce bude nacházet velké množství tras, tak systém zabrání dalšímu vytváření nových cest, aby zabránil ochromení směrovacího protokolu[9].

Rozesílání falešných směrovacích aktualizací do směrovací tabulky (Routing table poisoning)

V tomto útoku škodlivý uzel rozesílá falešné směrovací aktualizace do sítě nebo provádí modifikaci správných směrovacích aktualizací odesílaných ostatním uzlům v síti[9]. Tyto změny prováděné útočníkem můžou vést ke špatným údajům ve směrovacích tabulkách všech uzlů v síti. Otrávením směrovací tabulky bude ovlivněn přímo provoz v síti, protože některá data půjdou po špatných cestách. Tato otrava může dále způsobit přetížení sítě nebo dokonce její kolaps. Útok se záměrem otrávení směrovací tabulky může vést také k dalším útokům díky tomu, že útočník je schopen se neoprávněně přidat do jím požadované trasy[6].

3.3.2. Útoky na aktualizaci údajů ve směrovací tabulce

Útočník využívá zranitelnosti promiskuitního režimu aktualizace směrovací tabulky. Zranitelnost se projeví, pokud dojde ke smazání nebo vložení falešných informací do směrovací tabulky. Uzel si totiž přidá do své tabulky informace obsažené v hlavičkách zaslechnutých paketů i přesto, že se uzel nenachází na trase, která je uvedena v hlavičce přijatého paketu.

Tento útok se zabývá vkládáním falešných údajů do směrovací vyrovnávací paměti uzlů v síti. Při tomto útoku využívá útočník zranitelnost promiskuitního režimu aktualizace směrovací tabulky. Tato zranitelnost se projeví ve chvíli smazání, změnění nebo vložení falešných informací do směrovací tabulky. Jestliže uzel zaslechne libovolné pakety, tak si přidá informace, které obsahuje hlavička těchto paketů, do své směrovací paměti, i přesto, že se nenachází na trase udávané v přijaté hlavičce paketu. Škodlivý uzel O bude chtít otrávit cestu k uzlu B. Uzel O, proto začne sám vysílat do celé sítě (broadcast) falešné pakety s cestou od uzlu B k uzlu O. Sousední uzly samozřejmě zaslechnou toto vysílání a je možné, že si přidají tyto směrovací údaje do svých směrovacích pamětí[8][9].

3.3.3. Útoky na aktivní směrovací algoritmy

Útočník využívá vlastnosti směrovacích algoritmů, které hledají trasu až na vyžádání, tím, že provádí přeposílání, tunelovým spojením, každého RREQ paketu přímo k požadovanému cílovému uzlu. Mezi využívané vlastnosti DSR a AODV algoritmů patří zejména přeposílání RREQ paketů sousedním uzlem, který zaslechne procházející RREQ pakety a nachází se vedle cílového uzlu. Po tomto přeposlání kopií obdržených RREQ paketů začne bez nějakého zpracování zahazovat všechny ostatní pakety pocházející z toho samého procesu hledání trasy.

Aktivní směrovací algoritmy OLSR a DSDV spoléhají na to, že přijmou vysílané pakety a detekují podle toho své sousední uzly. Tyto protokoly používají pro zjištění sousedů HELLO zprávy, které může útočník snadno přesměrovat přes tunelové spojení všechny HELLO pakety odeslané zdrojovým uzlem cílovému uzlu a stejným způsobem odešle veškeré HELLO zprávy od cílového uzlu zpátky ke zdrojovému uzlu. Tím docílí toho, že si tyto dva uzly budou myslet, že jsou sousedi.

Většina směrovacích protokolů provádí vysílání HELLO zpráv za účelem ohlášení své přítomnosti sousedním uzlům. Uzel může předpokládat na základě přijatých HELLO zpráv, že se nachází v blízkosti uzlu, který mu poslal tyto zprávy. Této vlastnosti právě zneužívá útočník a vysíláním směrovacích zpráv s dostatečným výkonem může přesvědčit každý uzel, že je jeho soused. Díky tomu, že nabídne všem uzlům kvalitní trasu k základnové stanici, se jí budou všechny uzly v síti snažit využít.

Červí díra (Wormhole attack)

Útočník provádí útok ukládáním zpráv v jedné části sítě a následným přesměrováním těchto dat do jiné části sítě. Směrování v síti může být narušeno tím, že jsou přesměrovávány i řídící směrovací zprávy. Toto tunelové spojení dvou částí sítě se nazývá červí díra (wormhole). Tento útok je považován za vážnou hrozbu pro směrovací protokoly v síti MANET (Mobile Ad-Hoc Networks). Jestliže je tento útok použit proti směrovacím protokolům typu DSR (Dynamic Source Routing) a AODV (Ad-hoc On-demand Distance Vector Routing), tak je útočník schopen zabránit nalezení jakýchkoli tras vedoucích mimo tento tunel[8, 9].

Záplava sítě HELLO zprávami (Hello flood attack)

V normální komunikaci je vyžadováno mnoho protokoly zasílání HELLO zpráv mezi uzly za účelem oznámení přítomnosti sousedním uzlům v síti. Uzel si může myslet, že se nachází ve vysílacím okruhu odesílajícího uzlu, pokud obdrží jeho HELLO zprávy. Tento předpoklad uzlu ale může být také špatný a to z důvodu, že útočník je schopen vysíláním směrovacích nebo jiných informací s dostatečně velkým vysílacím výkonem přesvědčit každý uzel nacházející se v síti, že je jeho soused[13].

Útočník také může dosáhnout toho, že se většina uzlů bude snažit využít jím nabízenou trasu z důvodu, že bude nabízet falešné informace o tom, že vlastní kvalitní trasu k základnové stanici. Uzly nacházející se ve větší vzdálenosti od útočníka ale budou tyto pakety posílat do ztracena. To způsobí značný chaos v celé síti[13].

3.3.4. Útoky na reaktivní směrovací algoritmy

Reaktivní algoritmy, které hledají trasu, až když je o ní zájem, používají duplicitní potlačení paketů v průběhu hledání trasy. Jakmile škodlivý uzel obdrží RREQ (Route Request) od normálního uzlu, začne s ním zaplavovat celou síť z důvodu, aby ostatní uzly, které obdrželi stejný paket jako škodlivý uzel, na něj nemohli reagovat.

Z toho důvodu ostatní uzly po obdržení správných RREQ paketů je zahodí, protože je považují za duplicitní s těmi, které obdrželi od útočníka.

Na tento spěchající útok (Rushing attack) jsou zejména citlivé On-demand protokoly používající v průběhu hledání trasy duplicitního potlačení. Škodlivý uzel, který obdrží RREQ (Route Request) paket od zdrojového uzlu jím začne rychle zaplavovat celou síť. Tohle dělá kvůli tomu, aby ostatní uzly, které také obdrželi tento paket, na něj nemohli reagovat. Ostatní uzly, po obdržení správných RREQ paketů, je považují za duplicitní s již získanými pakety od útočníka a proto je odmítnou a zahodí je. Škodlivý uzel bude přítomen ve všech cestách nalezených zdrojovým uzlem jako jeden z mezilehlých uzlů. Ať se zdrojový uzel pokusí nalézt jakoukoli trasu, vždy se v ní bude nacházet škodlivý uzel jako jeden z mezilehlých uzlů. Z toho vyplývá, že zdrojový uzel není schopen nalézt libovolnou trasu bez přítomnosti škodlivého uzlu. Proto je těžké odhalit tento útok v senzorových sítích[9].

3.3.5. Útoky rozesílající falešné nebo upravené Route Reply pakety

Pokud normální uzel spustí proces hledání trasy, tak mu škodlivý uzel zpátky pošle falešný RREP paket s vyšším sekvenčním číslem, než které má správný cílový uzel a na základě porovnání sekvenčních čísel si zdrojový uzel nastaví trasu k cílovému uzlu, která byla obsažena v přijatém RREP paketu od útočníka. V případě Sinkhole útoku útočník využije vlastnost AODV protokolu a pošle zprávu s malou hodnotou přeskoků za účelem sdělení, že vlastní nejkratší cestu. Od té doby jde veškerý datový provoz od zdrojového uzlu do škodlivého uzlu. Útočník také může falšováním směrovacích informací tvrdit, že má kvalitní trasu k cílovému uzlu, že nefunkční uzel funguje normálně a že také vlastní nejkratší cestu k cíli.

Černá díra (Blackhole attack)

V průběhu tohoto útoku škodlivý uzel posílá padělané RREP (Route Reply) pakety zdrojovému uzlu, který provádí hledání tras mezi uzly. Díky těmto falešným paketům si uzel, který prováděl zjišťování trasy k cílovému uzlu, nastaví místo správného koncového uzlu škodlivý uzel útočníka. Zdrojový uzel po obdržení všech RREP paketů rozhodne na základě sekvenčního čísla cílového uzlu, které je obsaženo v RREP paketu. Podle nejvyššího sekvenčního čísla přijatého RREP paketu se zdrojový uzel rozhodne nastavit trasu obsaženou v příslušném RREP paketu.

Nejvyšší sekvenční číslo zdrojový uzel považuje za nosiče nejnovějších směrovacích informací a nastaví trasu směrování podle té trasy, která se nachází v příslušném RREP paketu[10].

Pokud budou obě sekvenční čísla stejná, tak se o trase zdrojový uzel rozhodne na základě množství přeskoků mezi jednotlivými uzly. Útočník může dosáhnout toho, že veškerý síťový provoz půjde přes něho, že se bude vydávat za cílový uzel. Odešle tedy zdrojovému uzlu RREP paket s vyšším sekvenčním číslem než jaké má správný cílový uzel a také pošle údaj o menším počtu přeskoků, než jaký je ve skutečnosti.

Tímto způsobem se nastaví trasa ke škodlivému uzlu a veškerý síťový provoz půjde přes tento uzel. A už bude záležet jen na škodlivém uzlu, jak bude nakládat s pakety, které k němu budou přicházet, zda je bude přeposílat, zahazovat nebo odesílat se zpožděním[10].

Šedá díra (Greyhole attack)

Tento útok se vyznačuje svým charakteristickým chováním. V průběhu útoku dochází také k zahazování datových paketů, ale tyto aktivity škodlivého uzlu jsou omezeny určitými podmínkami nebo spouštěcími mechanismy[11].

Dva nejběžnější typy chování:

- Útok zaměřený na uzly (Node dependent attack)

  V tomto režimu zahazuje data pocházející od určitého uzlu nebo data určená pro konkrétní uzel. Naopak u ostatních uzlů se chová normálně při směrování jejich datových paketů k cílovým uzlům[11].

- Útok závisící na čase (Time dependent attack)

  Při tomto útoku škodlivý uzel zahazuje datové pakety na základě předem daného času nebo při nastání nějaké akce. V jiných případech se však chová zcela korektně[11].

Medúza útok (Jllyfish attack)

Tento útok se trochu liší od útoků černá díra (Blackhole) a šedá díra (Greyhole). Místo toho, aby byly data zahazována, jsou jen zpožděna a poté teprve doručena. Útočník může dokonce změnit pořadí obdržených paketů a poté je může znovu odeslat v náhodném pořadí. Toto počínání narušuje funkčnost řídících mechanismů používaných uzly pro bezproblémový přenos dat. Tento útok může skončit značným zpožďováním komunikace od zdrojového uzlu až ke koncovému uzlu (end-to-end) a způsobí téměř nefunkčnost QoS (Quality of Service – jedná se o službu, která umí zabránit zahlcení přenosového kanálu tím, že umožňuje nastavit priority pro jednotlivé přenosy dat, které tak dostanou větší šířku přenosového pásma na úkor jiných síťových služeb)[11].

Výběr metod používaných útočníkem při provádění medúza (Jellyfish) útoku:

- Jedna z metod spočívá v promíchání pořadí paketů těsně před jejich doručením místo toho, aby zanechal útočník správné FIFO (První dovnitř – První ven (First In - First Out)) pořadí obdržených paketů. Mechanismus řízení toku bude proto generovat duplicitní ACK (potvrzovací) pakety a způsobí tím nadměrné spotřebovávání šířky pásma a také dojde ke snižování životnosti baterie[41].
- Další metodou útoku může být spuštění útoku výběrové černé díry (*selective* Black hole), který bude zahazovat všechny pakety v každém RTO (recovery time objective). Toto chování způsobí vypršení časového limitu po dobu trvání tohoto útoku v každém RTO.Škodlivý uzel si může ukládat všechny přijaté pakety do své paměti (bufferu), ale tyto pakety odešle až po delší době se značným zpožděním. Zachová, ale u těchto paketů pořadí v jakém je obdržel. V této fázi dojde také ke zmatení mechanismu řízení provozu v síti. Někdy také nastane situace, že zdrojový uzel vybere pro přenos dat delší trasu místo té nejkratší cesty. Toho dosáhne útočník tím, že uzlu zašle falešné RREP pakety ve kterých uvede, že vlastní nejkratší cestu[41].

Sinkhole útok

Snahou útočníka je přesměrovat veškerý síťový provoz do jeho škodlivého uzlu. Tento útok je také možné použít jako základ pro spuštění dalších útoků na síť díky tomu, že se škodlivé uzly nacházejí buďto přímo na přenosové trase nebo v její blízkosti a mají tedy možnost manipulovat s přenášenými daty. Útok funguje nejčastěji tak, že škodlivý uzel předstírá vlastnictví kvalitní přenosové trasy.

Útočník může tyto falešné informace o své přenosové trase rozeslat ostatním uzlům nebo je přímo nahrát do základnové stanice. Ve snaze odhalit tento útok se některé protokoly snaží ověřit kvalitu zmíněné trasy zjištěním informací o spolehlivosti a odezvě této trasy zkontrolováním konektivity od zdrojového uzlu k cílovému uzlu (end-to-end)[12].

Falšování potvrzujících zpráv (Acknowledgement spoofing)

Některé směrovací algoritmy v senzorových sítích se spoléhají na implicitní nebo explicitní potvrzení od linkové vrstvy. Díky tomu, že uzly vysílají své pakety do širokého okolí, tak není pro útočníka problém zfalšovat potvrzení od linkové vrstvy pro ty pakety, které zachytil. Cílem tohoto útoku je přesvědčit odesílající uzel o tom, že slabá linka je ve skutečnosti silná. Dále útočník vydává nefunkční nebo poškozený uzel za normálně fungující. Směrovací protokol má možnost si zvolit následující uzel na přenosové trase za použití odkazu spolehlivosti (link reliability)[14].

Jednoduchý způsob, jak zmanipulovat tento systém, je umělé navýšení slabého nebo mrtvého odkazu. Jestliže jsou pakety odeslány přes slabé nebo zničené linky, může útočník jednoduše spustit útok přeposílání jen vybraných paketů (selective forwarding útok). Tento útok spustí díky zfalšování potvrzení od linkové vrstvy, čímž přesvědčí cílový uzel, aby použil pro veškerý jeho přenos dat tyto spoje doporučené od útočníka[14].

Vydírání uzlu (Blackmailing and Co-operative Blackmailing attack)

Při tomto útoku obviní škodlivý uzel nevinný uzel, že je také škodlivý. Tento útok je zejména efektivní proti těm distribuovaným protokolům, které si zakládají a vedou seznam správných a škodlivých uzlů na základě zjištění ostatních uzlů v síti. Jen malá část protokolů si dává záležet na bezpečnosti a zavádí princip většinového hlasování. Jestliže se ale dostane škodlivý uzel do této sítě, dokáže toto zabezpečení také obejít. Mezi další způsoby provádění tohoto útoku ještě patří zasílání neplatných RREP (Route Reply paket) zpráv, ve kterých útočník uvádí falešné náklady na určité uzly[17].

Zanedbávání a chamtivost (Neglect and Greed)

V průběhu tohoto útoku se škodlivé uzly často podílejí na výměně některých dat nebo informacích mezi sousedními uzly. Podílejí se na předávání tím, že místo směrování obdržených dat do jejich cílových destinací, tyto důležitá data zahazují. Škodlivý uzel může ale také vysílat sám sebe do celé sítě jako nejkratší cestu do cílového uzlu a tím přesměrovat celý síťový provoz přes sebe. Tohle celé způsobí vznik zahlcení přenosové linky v okolí tohoto uzlu. Mezi opatření proti tomuto útoku se nejčastěji zavádí směrování přes více tras nebo záměrná redundance u přenášených zpráv[18].

Nesprávné zaměření (Misdirection)

Tento útok je spíše řazen mezi aktivní útoky, při kterých škodlivý uzel za pomoci propagování špatných tras může celý síťový provoz přesměrovat na špatné cesty. Efektivní opatření proti tomuto útoku může být realizováno směrováním informací pouze oprávněnými uzly[18].

3.3.6. Útok na protokoly závislé na lokální výměně informací

Častým cílem útoku jsou také protokoly, které jsou závislé na lokální výměně informací mezi sousedními uzly za účelem vytváření topologie sítě nebo řízení datového toku. Útočník nutně nepotřebuje vytvořit legitimní provoz za účelem pozdějšího použití zaplavení sítě HELLO zprávami. Stačí mu jen znovu vysílat do sítě režijní pakety s dostatečným vysílacím výkonem. Tyto pakety poté obdrží všechny uzly přítomné v dané síti.

Záplavové vysílání HELLO zpráv může být také chápáno jako jednosměrné vysílání červí díry (wormhole)[14].

Jedna ze zranitelností směrovacích protokolů, které využívá pro svoji funkčnost Sybil útok, je směrování přes více uzlů, kde na první pohled různé trasy můžou ve skutečnosti procházet jen jedním škodlivým uzlem, který se vydává za více identit, které ukradl jiným uzlům v síti.

Další zranitelnost spočívá v geografickém směrování, kde se může Sybil uzel vyskytovat na více místech najednou místo toho, aby měl jen jednu sadu souřadnic.

V tomto útoku vytváří útočník Sybil uzel, který používá řadu identit, pro zničení mapování mezi subjektem a identitou. Tento typ útoku se nejprve vyskytoval pouze v P2P (peer-to-peer) sítích, ale v dnešní době došlo k jeho rychlému rozšíření i do bezdrátových senzorických sítích. Následky po tomto útoku mohou být velmi rozsáhlé.

V průběhu tohoto útoku není možné zajistit a garantovat výkon a rychlost pro velké množství síťových mechanismů. Mezi ně patří zejména směrovací protokoly, hlasování, schéma pobídek (incentive scheme) a také bezpečnostních mechanismů. Detekce a obrana proti Sybil útokům je v bezdrátových sítích velmi náročná zejména proto, že uzly mají omezenou energii na svůj provoz, schopnosti výpočtů a komunikaci[15].

3.3.7. Útok na vyčerpání energie uzlu

Uzly, které jsou napájeny bateriemi, vysílají jen v opravdu nezbytných případech, aby spotřebovali co nejméně svojí energie. Ale musejí se také podílet na směrovacím procesu, jinak by se mohl stát nedostupný v síti. Této vlastnosti zneužije útočník a začne posílat na daný uzel velké množství požadavků na zjištění rozsáhlé trasy nebo zbytečných paketů. Tímto chováním zaměstná uzel a tomu se samozřejmě zvýší spotřeba energie.

Pro všechny uzly v síti MANET, které jsou závislé na napájení bateriemi, je životně důležité udržet si energii. Proto tyto uzly vysílají jen tehdy, je-li to opravdu nezbytné. Hlavním cílem tohoto útoku je spotřebovat co nejvíce energie z každého napadenému uzlu. Toho dosáhne útočník zasíláním mnoha požadavků na zjištění rozsáhlé trasy nebo zasíláním naprosto zbytečných paketů. Tímto počínáním může dojít k narušení normálních funkcí v síti MANET (Mobile Ad-hoc Network). Tento útok se také někdy nazývá útok na odepření spánku uzlu (sleep deprivation attack)[8, 9].

Zaměření tohoto útoku je zejména proti zařízením, které neposkytují v síti žádné služby nebo služby poskytují, ale jen na základě předloženého oprávnění. Uzel je povinen se podílet na směrovacím procesu bez ohledu na vlastnosti služeb. Jestliže by se nepodílel na směrování, tak by riskoval možnost, že se stane nedostupným v síti[13].

3.3.8. Obrana proti těmto útokům

Útoky, které vysílají do sítě velké množství SYN paketů se žádostí o nové spojení, chtějí docílit zaplnění směrovací tabulky v napadeném uzlu. V tomto případě nám šifrování přenosu nijak nepomůže a musíme vybrat jiné způsoby či metody, jak se proti těmto útokům bránit. Napadený uzel by si mohl počítat množství žádosti o spojení a při překročení určitého počtu žádostí od jednoho uzlu by ho mohl na nějakou dobu ignorovat a tudíž od něj nepřijímat žádné další žádosti o spojení. Nebo by si mohl nejprve ověřit pravost tohoto uzlu, který mu pořád posílá další a další žádosti o spojení a pokud by zjistil, že tento uzel je škodlivý, tak by ho mohl odstřihnout od sítě nebo ho nahlásit ostatním uzlům ve svém okolí, že se jedná o škodlivý uzel.

Pokud by tento škodlivý uzel posílal normálnímu uzlu falešné směrovací informace, mohl by být normální uzel ovlivněn při hledání nové trasy, kdy by mohl použít směrovací informace přijaté od útočníka a posílat tak svá veškerá data ke škodlivému uzlu. Proti tomuto útoku by se mohl bránit tím, že by si ověřil pravost škodlivého uzlu, a když by zjistil, že je falešný, tak by nepoužil směrovací údaje, které od něho obdržel.

Nebo by při hledání nové trasy slepě nedůvěřoval jen informacím od útočníka, ale vyhledal by si více tras a ty by potom porovnal, čímž by zjistil, že má třeba tři podobně dlouhé trasy a jednu podezřele krátkou. Díky tomu by dokázal zjistit, že se jedná o falešnou trasu a nepoužil by ji při směrování.

Patří sem zejména následující útoky: útok na přetečení směrovací tabulky (Routing table overflow attack), rozesílání falešných směrovacích aktualizací do směrovací tabulky (Routing table poisoning).

Zavedením šifrování se nám nepodaří zabránit útokům, které se snaží za pomoci rozesílání falešných RREP paketů, při hledání trasy normálním uzlem, přesvědčit tento uzel, aby si jako svoji trasu pro posílání dat zvolil útočníkem doporučovanou trasu. Útočník může normální uzel přesvědčit tím, že mu odešle RREP pakety s nižším sekvenčním číslem než jaké je ve skutečnosti, nebo ho přesvědčí o tom, že vlastní trasu s nejmenším počtem přeskoků k cílovému uzlu.

Proti těmto útokům by se mohl uzel bránit tím, že nebude slepě věřit přijatému RREP paketu s podezřele nižší hodnotou (jak sekvenčního čísla nebo počtem přeskoků), ale spíše by měl porovnat hodnoty těchto čísel ve všech přijatých RREP paketech a až po té se třeba rozhodnout, kterou trasu skutečně využije. Další možností by mohlo být to, že by si nejprve ověřil identitu a pravost všech uzlů, které mu zaslaly RREP pakety. To by mohlo být ale náročné na energii a také celkem zdlouhavé. Také by šlo, že by si uzel pouze ověřil pravost toho uzlu, který mu poslal RREP paket s nejlepšími hodnotami. Ještě by si mohl zjistit více tras, čímž by zjistil, zda není náhodou trasa doporučovaná útočníkem podezřele krátká nebo zda je vůbec funkční.

Do této skupiny patří zejména tyto útoky: černá díra (Blackhole attack), šedá díra (Greyhole attack), medúza útok (Jellyfish attack), sinkhole útok, falšování potvrzujících zpráv (Acknowledgement spoofing), vydírání uzlu (Blackmailing and Co-operative Blackmailing attack), zanedbávání a chamtivost (Neglect and Greed) a nesprávné zaměření (Misdirection).

Útoku, který se snaží o vyčerpání baterie konkrétního uzlu, nedokážeme úspěšně zabránit. Útočník totiž na daný uzel posílá značné množství požadavků na zjištění trasy nebo úplně zbytečné pakety jen za tím účelem, aby uzel musel stále provádět výpočty a čerpal tak svoji drahocennou energii. Uzel by se mohl proti tomuto útoku bránit několika způsoby: pokud by dostával od útočníka velký počet žádostí za sebou, mohl by si chtít ověřit jeho pravost po přijetí určitého počtu žádostí o nalezení trasy; mohl by se také zeptat sousedních uzlů, jestli také dostávají tolik žádostí o nalezení trasy od jednoho uzlu (zřejmě se bude jednat o škodlivý uzel) a pokud ne, tak by ho uzel mohl blokovat a nepřijímat od něj již žádné další pakety; nebo pokud by uzlu docházela energie, tak by mohl přijímat jen určitý počet žádostí a po přijetí třeba 20 žádostí by již po nějaký čas nepřijímal žádné žádosti od tohoto uzlu a začal by zase až po vypršení časového kvanta.

Sem patří útok na vyčerpání baterií jednotlivých uzlů (Resource consumption útok).

Většina směrovacích algoritmů používá pro zjištění svých sousedů rozesílání HELLO paketů. Pokud uzel přijme tyto pakety, tak pozná, že je v blízkosti uzlu, který tyto pakety rozeslal. Tohohle využije útočník a přesměruje obdržené pakety z jedné části sítě do druhé části přes tunelové spojení a docílí toho, že si tyto dva uzly, zdrojový a cílový, budou myslet, že jsou sousedi. Dále může útočník přesvědčit okolní uzly v sítí, že je jejich soused, tím, že použije pro vysílání těchto HELLO zpráv dostatečně výkonný vysílač.

Proti těmto útokům je možné se bránit upravením RREQ paketů o značku (flag), kterou do těchto paketů při hledání nové trasy vloží odesílající uzel. Díky této značce je zajištěno, že na tento paket bude moci odpovědět pouze příjemce a nikoliv škodlivý uzel. Pokud příjemce obdrží tento RREQ paket, tak na něj vzápětí reaguje odesláním RREP paketu, se svou aktuální pozicí, uzlu, od kterého mu přišel tento paket. Příjemce tohoto paketu ho ověří, zda opravdu pochází od cílového uzlu. Odesílající uzel je schopen odhadnou nejkratší cestu k cílovému uzlu a příjemce si také zjistí počet přeskoků z obdrženého RREP paketu, je poté schopen porovnat tyto dvě hodnoty přeskoků. Pokud zjistí, že jím odhadované množství přeskoků mezi uzly značně přesahuje hodnotu obdrženou od cílového uzlu v RREP paketu, pak si uvědomí, že je pod útokem červí díry a označí tuto trasu jako nebezpečnou pro ostatní uzly.(25)

Do této skupiny útoků řadíme zejména tyto útoky: červí díra (Wormhole attack) a záplava sítě HELLO zprávami (Hello flood attack).

3.4. Aplikační vrstva

3.4.1. Útoky způsobující zkreslení dat při jejich agregaci

Aplikační vrstva obsahuje zejména uživatelská data a protokoly HTTP (Hypertext Transfer Protocol), SMTP (Simple Mail Transfer Protocol) a FTP (File Transfer Protocol). Škodlivý kód zahrnující viry a červy je snadno použitelný také na různých operačních systémech a v různých aplikacích. V sítích jsou velmi rozšířené škodlivé programy. Existuje velké množství způsobů, jak může červ objevit v síti nový stroj využitelný pro jeho potřeby. Jedna z možností, kterou používají červi pro nalezení nového stroje, je skenování IP adres. Tento způsob útoku sestává z generování (probe) zjišťujících paketů, které jsou odesílány na zranitelné TCP/UDP porty nacházející se na mnoha IP adresách. Uzly, které obdrží tyto pakety používané červem pro skenování IP adres, získají také kopii tohoto červa a nakazí se jím. Jeden z červů, kteří provádějí skenování, je červ Code Red[6].

Někteří červi využívají pro svůj útok zejména chyb v systémech. Mezi takové patří například Worm.Blaster a také Worm.Sasser. Každý z nich využívá jinou zranitelnost v systému pro svůj útok. Worm.Blaster využívá chybu v RPC DCOM (Remote Procedure Call Distributed Component Object Model – RPC je protokol používaný v systémech Windows pro zajištění bezešvé komunikace mezi programy běžící na lokálním počítači a na vzdáleném serveru. DCOM je protokol umožňující programovým komponentám přímou komunikaci přes vícero protokolů zahrnující i internetové protokoly jako je třeba HTTP) a Worm.Saaer používá pro napadení systém LSASS (local security authentication subsystem service). V sítích MANET může útočník vytvořit červa, který bude využívat jakoukoli chybu v systému používaném v mobilních ad-hoc sítích[8].

Zabránění uzlu v komunikaci (Repudiation attack)

Do síťové vrstvy se instalují firewally za účelem zabránění škodlivým paketům vstoupit do sítě a také aby nedošlo k vysílání škodlivých paketů do sítě. V průchodu síťovou vrstvou může být celé spojení šifrováno od odesílajícího uzlu k cílovému uzlu (end-to-end). Tato řešení ale nepomohla vyřešit problém s ověřováním uzlů a problémy s odmítnutím služby. Odvržení (repudiation) se týká zejména odmítnutí účasti v celé komunikaci nebo jen v její části. V této části uvádějí autoři článku příklad na sobeckém člověku, který by nám blokoval provedení platby kreditní kartou nebo by odmítl každou naši transakci přes on-line bankovnictví. Toto je typický příklad útoku na zabránění (repudiation attack) námi chtěných požadavků v obchodním systému[8].

Rozlaďování hodin každého uzlu v síti (Clock Skewing)

Tento útok se zaměřuje na vybrané uzly, které potřebují mít přesně synchronizovaný čas pro společné operace s ostatními uzly. Cílem těchto útoků je rozladit čas v jednotlivých uzlech pomocí vysílání falešných zpráv obsahujících informace o čase.

Ve standardu IEEE 802.11 (může být aplikován také do bezdrátových senzorických sítí) je potřeba zajistit synchronizaci uzlů s přístupovým bodem (access point). Přístupový bod vysílá tyto řídící pakety periodicky všem uzlům v síti[6].

Tyto pakety v sobě obsahují informaci o čase potřebnou pro nastavení hodin každého uzlu. Útočníci se proto pokoušejí rozesílat řídící pakety s falešnými časovými údaji. V případě, že si uzly nastaví svoje hodiny na čas uvedený ve špatných řídících paketech, dojde k porušení synchronizace s přístupovým bodem. Uzly jde pomocí správných řídících paketů později opět synchronizovat s přístupovým bodem, ale tyto uzly se budou i nadále pohybovat mezi těmito dvěma stavy (budou dostávat v jednu chvíli správné informace o čase a po chvíli zase obdrží falešné informace) a budou nestabilní[6].

Zkreslení agregace dat (Data Aggregation Distortion)

Jakmile jsou data shromážděna, uzel je odesílá zpátky na zpracování do základnové stanice (base station). Útočník se může pokusit pozměnit data před jejich spojením a tím dosáhne zkresleného výpočtu těchto dat základnovou stanicí. V důsledku toho bude mít základnová stanice zkreslený pohled na prostředí, které je monitorováno senzory, a může tak vyvolat nesprávnou akci. Pokud dojde ke spuštění útoku černá díra (Black hole) nebo útoku, který přesměruje veškerý síťový provoz do uzlu útočníka (Sink hole), v síti, může se stát, že bude zcela narušena agregace dat. V této situaci nedojdou žádná data k základnové stanici. Jelikož tyto útoky (Black hole. Sink hole) vyžadují znalost síťové vrstvy pro svou funkčnost a proto se řadí do útoků na síťovou vrstvu[6].

3.4.2. Útoky přeposílající jen vybrané zprávy

Při tomto útoku se musí útočník nacházet na trase mezi zdrojovým uzlem a cílovým uzlem. Je proto odpovědný za přeposílání paketů zdrojovému uzlu podle svého uvážení. Tento útok může být spuštěn přeposíláním vybraných zpráv nebo jen jejich částmi. Tento útok se liší od útoku selective forwarding v síťové vrstvě. Pro spuštění tohoto útoku musí být útočník schopen porozumět sémantice (payload) paketů přenášených na aplikační vrstvě. To znamená pro uzel zacházet s každým paketem jako se smysluplnou zprávou místo jako s monolitickou jednotkou. Dále útočník vybírá pakety, které mají být předány dále, na základě sémantiky. Naproti tomu přeposílání jen vybraných zpráv (selective forwarding) v síťové vrstvě vyžaduje po útočníkovi, aby znal informace síťové vrstvy, jako jsou zdrojové a cílové adresy.

Útočníci se musí rozhodnout, zda budou přeposílat pakety jen podle uvedených informací a proto pracují s velkou mírou roztříštěnosti[6].

4. Zabezpečení komunikace mezi uzly

Jedním z hlavních cílů bezpečnostních služeb v bezdrátových senzorických sítích je chránit přenášené informace a jednotlivé uzly před ovlivňováním útočníkem[3].

Mezi bezpečnostní požadavky patří: [3]

- Dostupnost – zajišťuje dostupnost námi požadovaných síťových služeb i při probíhajícím útoku na odmítnutí služeb (DoS – Denial of Service)
- Povolení – zajišťuje, že pouze povolené uzly můžou poskytovat informace síťovým službám
- Ověření – poskytuje nám jistotu, že uzel komunikuje vždy se správným uzlem a ne ze škodlivým uzlem, který se jen vydává za důvěryhodný uzel
- Důvěrnost – zajišťuje, že danou zprávu nebude moci přečíst a porozumět jí nikdo jiný, než uzel, kterému je adresována
- Integrita – dává nám jistotu, že zpráva, odeslaná z jednoho uzlu na druhý, nebyla nijak upravována škodlivým uzlem
- Nemožnost odmítnutí – uzel nemůže popřít, že odeslal zprávu, která již byla před tím odeslána
- Aktuálnost dat – máme zde zaručeno, že data jsou v aktuální podobě a zajišťuje nám, že útočník nebude schopen znovu poslat starší zprávy
- Časová synchronizace – schopnost domluvit se na čase provedení operace (příjem, vysílání), synchronním sběru dat a na synchronním čase pro ověření platnosti
- Bezpečné vytváření a údržba skupin – schopnost vytvářet a rušit skupiny
- Bezpečná lokalizace – důležitá pro udržení integrity sítě, kdy jednotlivé uzly komunikují pouze s ověřenými sousedy

Z důvodu přidávání nových senzorů do sítě a odebírání již nefunkčních senzorů ze sítě je dobré dodržovat také následující pravidla:

- Utajení předávání – snímač by po opuštění sítě již neměl mít možnost číst jakékoli zprávy posílané v této sítí
- Utajení předchozích zpráv – nově připojený senzor by neměl číst již přenesené zprávy v síti

Dále budou rozebrány možnosti, jak zajistit bezpečnost v senzorické síti spolu se zabezpečením dat před útočníky.

4.1. Obrana proti odposlouchávání přenosu dat – FHSS, DSSS

4.1.1. Přepínání frekvencí v rozprostřeném spektru – FHSS

Pro tuto metodu se používá zkratka FHHS, což značí Frequency Hopping Spread Spectrum. Pro zabránění rušení signálu útočníkem se při přenosu dat používá postupu, při kterém dochází k přepínání frekvencí, na kterých se vysílají data, podle předem známého schématu přepínání frekvencí. Přenášený signál je modulován řadou rádiových frekvencí, které se mohou případným útočníkům jevit jako zdánlivě náhodné frekvence, a dochází k přepínání z jedné frekvence na druhou ve fixních intervalech. Způsob rozprostření a tedy vybírání různých frekvencí musí znát jak přijímací uzel, tak i odesílající uzel. Díky této znalosti dokáže cílový uzel tento rozprostřený signál opět složit do původního signálu[8].

Data můžou být přenášena klidně jen přes jeden kanál, ale v tom případě musí být vysílací uzel i přijímací uzel správně synchronizovány. Díky rozprostření signálu do širšího frekvenčního spektra a znalosti tohoto rozdělení pouze vysílačem a příjemcem, se tyto přenášená data jeví případným útočníkům, kteří tento přenos odposlouchávají, jako nesrozumitelné impulzy šumu[8].

Toto přepínání frekvencí používá v praxi například technologie Bluetooth pro komunikaci mezi dvěma zařízeními a následnému přenosu dat. Bluetooth pracuje v bezlicenčním pásmu 2,4 GHz, ve kterém pro přenos dat využívá právě metodu přepínání frekvencí FHSS. Při přenosu dat je každou sekundu provedeno 1600 přeskoků mezi používanými 79 frekvencemi. Těchto 79 frekvencí je od sebe vzdáleno právě 1 MHz.

4.1.2. Nahrazení vstupních bitů sekvencí bitů v rozprostřeném spektru – DSSS

Každý přenášený datový bit v původním signálu je reprezentován více bity ve vysílaném signálu za použití kódu rozprostření. Tímto postupem se rozprostře signál přes více frekvenčních pásem, jejichž počet je přímo úměrný počtu použitých bitů. Příjemce takového vysílání z něj opět dokáže sestavit původní signál za použití stejných kódů používaných také odesílatelem[8].

V tomto schématu je každý původní bit z přenášených dat reprezentován 4 bity v přenášeném signálu. První bit je zde 0 a je přenesen jako 0110, což odpovídá prvním 4 bitům z kódu rozprostření. Druhým bitem v pořadí je 1 a přenáší se jako 0110. Tato sekvence je bitovým doplňkem s druhými 4 bity z kódu rozprostření. V průběhu tohoto procesu je u každého vstupního bitu provedena exkluzivní operace XOR se 4 bity z kódu rozprostření[8].

Použití technologií FHSS a DSSS (Direct Sequence Spread Spectrum) způsobí útočníkům značné problémy při snaze odposlechnout tyto rádiové přenosy. Útočník (eavesdropper) musí znát pro úspěšné čtení přenášených signálů jejich frekvenční pásmo, kód šíření a také použitou modulaci signálu. Technologie rozprostření signálu dále také zmenšuje možnost rušení od ostatních rádiových a elektromagnetických zařízeních. Tato technologie může být bezpečná jen v tom případě, pokud útočník nezná vzorek přepínání z jedné frekvence na druhou a také způsob rozprostření signálu[8].

Technologii DSSS využívá pro svoji funkčnost např. protokol 802.15.4. Tento standard se zabývá komunikací mezi dvěma zařízeními na druhé vrstvě OSI/ISO modelu. Používá právě DSSS za účelem snížení ovlivňování přenášených dat šumem na přenosovém kanálu. Při tomto přenosu je každý informační bit zakódován do čtyř různých signálů, které jsou poté přeneseny přes síť. Přenáší se tedy větší množství dat, které obsadí větší šířku pásma, ale na druhou stranu je zapotřebí pro přenos těchto signálů menší množství energie. Pro zjištění, zda nebylo manipulováno s přenášenými pakety, se používají protokoly, které přidávají za konec paketu malou informaci, která v sobě obsahuje sekvenční číslo nebo maximální počet přeskoků mezi uzly. Tato informace je zde uložena v podobě hash řetězce.

Bezpečnostní protokol SEAD (Secure Efficient Distance Vector Routing for Ad-hoc Networks) je příklad z více protokolů, které se používají na obranu před útoky na modifikaci dat. Protokol SEAD používá jednosměrné hash řetězce podobně jako obecný mechanismus *packet leash* (jedná se o mechanismus používaný pro detekci a obranu před červím útokem (wormhole) útokem. Leash (vodítko) je jakákoliv informace přidaná k paketu za účelem omezení maximální povolené přenosové vzdálenosti). Tento mechanismus je používán pro zabránění škodlivým uzlům ve zvyšování sekvenčního čísla nebo ve snižování počtu přeskoků mezi uzly ve směrovacích paketech, které slouží pro zjišťování trasy k cíli. V protokolu SEAD je nutné, aby uzly dokázali ověřit své sousední uzly pomocí vysílání autentifikačního protokolu TESLA (Timed Efficient Stream Loss-tolerant Authentication) nebo za pomoci mechanizmů symetrického šifrování. V protokolu SEAD každý uzel generuje hash řetězec a uspořádává ho do segmentů o m prvcích v podobě

$$(h_0, h_1, \dots h_{m-1}), \dots., (h_{km}, h_{km+1}, \dots, h_{km+m-1}), \dots, h_n$$

V tomto řetězci znamená $\frac{\mathrm{n}}{\mathrm{m}} - \mathrm{i} = \mathrm{k}$, m je maximální průměr sítě a i je sekvenční číslo[8].

	j=0	1	2	3	4
i=1	h_{15}	h_{16}	h_{17}	h_{18}	h_{19}
2	h_{10}	h_{11}	h_{12}	h_{13}	h_{14}
3	h_5	h_6	h_7	h_8	h_9
4	h_0	h_1	h_2	h_3	h_4

Tabulka 1 – Příklad protokolu SEAD [28]

V tabulce je zobrazena hash funkce pro autentifikaci zpráv, i představuje sekvenční číslo, j je metrika udávající počet přeskoků přes uzly do cílové destinace, průměr sítě (m) je 5 a délka hash řetězce (n) je 20. Díky tomu, že dostaneme $h_i = H(h_{i-1})$, je tak velmi jednoduché ověřit správnost h_j pokud platí $j < i$. Není možné odvodit h_j z $j < i$, ale je možné odvodit h_j z $j > i$. Jelikož se používá rozdílná hash funkce pro různé i a j a používá se podle pořadí, které je uvedeno v tabulce výše, je pro útočníka takřka nemožné, aby zfalšoval a nastavil nižší hodnotu metriky nebo zvýšil sekvenční číslo. Ve směrovacím protokolu DSDV (Destination Sequenced Distance Vector routing) uzel po obdržení směrovacích aktualizacích si je zapíše do své směrovací tabulky jen v tom případě, pokud je sekvenční číslo větší nebo je sekvenční číslo stejné, ale zmenšila se hodnota metriky. Protokol SEAD zamezuje škodlivým uzlům ve snížení hodnoty udávající počet skoků (hop count) nebo ve zvýšení sekvenčního čísla na základě návrhu protokolu DSDV[8].

4.2. Zabezpečení šifrovacími algoritmy

4.2.1. Standardní algoritmy

Mezi používané symetrické šifry patří zejména tyto: RC4, RC5, IDEA, DES, 3DES a zejména nejvíce používané AES. Tyto šifry se používají zejména pro šifrování přenášených dat, kde je důležitá rychlost šifrování/dešifrování, ale již nejsou vhodné pro navazování zabezpečeného spojení z důvodu problému, jak distribuovat klíče mezi jednotlivé uzly.

Z důvodu problému s rozdělením klíčů do uzlů se pro navazování šifrovaného spojení používají asymetrické šifry, které tímto problémem netrpí. Jsou ale výpočetně náročné pro samotné uzly zejména kvůli dlouhým klíčům a proto se používají nejčastěji jen pro navázání spojení přes nezabezpečený kanál a pro přenos relačního klíče pro následné šifrování přenášených dat symetrickou šifrou. Mezi tyto asymetrické šifry řadíme RSA a ECC (Elliptic Curve Cryptography) a mezi protokoly na bezpečnou výměnu relačního klíče se pak používá např. Diffie–Hellman.

Hash funkce se používají buďto pro zjištění, zda nedošlo k úpravě přenášené zprávy. V tomto případě se jedná o kódy pro detekci modifikace (Modification Detection Codes). Další použití mají po přidání tajného klíče ke zprávě, ze které počítáme otisk. Tyto kódy nám umožňují ověřit pravost a integritu doručené zprávy. Toho docílíme tím, že můžeme přidat před a za konec zprávy nám známý tajný klíč, který ale ovšem musí znát i příjemce naší zprávy. Díky tomu není schopen útočník dopočítat otisk k zprávě, kterou zachytil. Mezi používané hash funkce patří MD5, které se již ale nedoporučuje používat z důvodu nalezení kolizních řetězců. Jinak tato hash funkce používá výstupní bloky o velikosti 128 bitů. Dále sem patří hash funkce SHA, která ale byla stažena těsně před schválením na pokyn NSA. Musela být drobně upravena a až poté byla schválena jako SHA-1.V této hash funkci je výstupní blok veliký 160 bitů. U této šifry již není po roce 2010 garantována bezpečnost a je také proto doporučeno ji přestat používat. Místo ní je právě doporučována SHA-2, která je zatím bezpečná, s délkou výstupu 224 – 512 bitů Mezi další hash funkce patří také RIPEMD s délkou výstupu 128 – 320 bitů. Tato funkce je považována za bezpečnou, pokud je délka výstupu alespoň 160 bitů.

Z důvodu velké náročnosti při šifrování a dešifrování u asymetrické kryptografie, a zejména u RSA, se začínají používat eliptické křivky pro navazování šifrovaného spojení. Pro šifrování přenášených dat se opět používá symetrická kryptografie. V následující tabulce je uvedena velikost klíče u RSA a u eliptických křivek a náročnost na prolomení těchto klíčů. Jak je vidět, eliptické křivky používají o hodně kratší klíče než RSA a jsou tak daleko méně náročné pro samotné výpočty na uzlech, které díky tomu ušetří svoji energii a budou tak moci fungovat déle než při používání RSA.

Čas na prolomení (MIPS za rok)	Velikost klíče RSA (v bitech)	Velikost klíče ECC (v bitech)
10^4	512	106
10^8	768	132
10^{11}	1024	160
10^{20}	2048	210
10^{78}	21000	600

Tabulka 2 – Porovnání síly klíčů RSA a ECC [21]

V této tabulce jsou uvedeny délky klíčů nejpoužívanějších šifer. Je zde dobře vidět, že si eliptické křivky vystačí s kratšími klíči při zachování stejné úrovně bezpečnosti jako má šifra RSA, která ale musí použít delší klíče.

Délka klíče RSA	Délka klíče DSA	Délka klíče ECC	Poměr velikosti klíčů
1024	512	160	1:6
2048	1024	224	7:64
3072	2048	256	1:12
7680	3072	384	1:20

Tabulka 3 – Porovnání délky klíčů (v bitech) u různých šifrovacích algoritmů [21]

4.2.2. Eliptické křivky

Hodně rozšířená šifra RSA využívá pro svoje fungování problému s rozkladem velkých čísel. Toto už ale začíná být problém v dnešní době, kde se neustále vyvíjí rychlejší hardware a stoupá výpočetní výkon počítačů. Tento problém se snaží řešit eliptické křivky založené na složitosti výpočtů diskrétních logaritmů. Eliptické křivky používají kratší klíče než běžné asymetrické šifry, zaberou při přenosu menší šířku pásma a jsou také velmi odolné vůči útokům. V současné době se již příliš nedoporučuje používat pro zabezpečenou komunikaci, pomocí šifry RSA, klíč o délce 1024 bitů, ale raději o délce 2048 bitů nebo i více bitů. Větší délky klíče nám sice zvýší bezpečnost šifry, ale také mají vliv na větší využití přenosového pásma a na snížení efektivnosti celého šifrování z důvodu značné náročnosti na výpočet klíčů a následného šifrování a dešifrování[20].

Stačí nám použít pro upřesnění konkrétní eliptické křivky označení F_q, ve kterém je $q = p^m$. Malé q je v tomto případě počtem prvků, písmeno p je prvočíslo a písmeno m je přirozené číslo[22].

Mezi zřejmé výhody eliptických křivek oproti šifrovacímu algoritmu RSA patří:

- Algoritmy

Jako jedna z hlavních výhod eliptických křivek oproti algoritmu RSA je uváděna základní operace, při které se sčítají body P a Q za účelem získání bodu R, která je ale také známá svojí výpočetní náročností. Tato náročnost je také jedním z více důvodů, proč se nemusíme obávat v blízké budoucnosti existence nějakých obecných dílčích exponenciálních útoků na eliptické křivky. V dnešní době sice již existuje několik útoků na konkrétní třídy eliptických křivek, ale tyto rizikové křivky lze snadno objevit a vyhnout se jejich použití při šifrování. Na algoritmus RSA jsou již známé dílčí exponenciální útoky. Z toho důvodu je nutné, aby generované klíče v RSA měly mnohem více bitů než klíče generované v ECC kvůli zachování stejné úrovně bezpečnosti. Větší délka klíče v RSA je způsobena rostoucím výpočetním výkonem a bude se nadále zvětšovat rychleji než délka klíčů v ECC[21].

- Požadavky na paměť

Kvůli neustále se zvyšujícímu výpočetnímu výkonu je potřeba používat delší klíče a je tedy nutné zvyšovat počty tranzistorů v procesorech jednotlivých uzlů za účelem zvládnutí těchto čím dál náročnějších výpočtů. Pokud použijeme místo šifrování RSA eliptické křivky, které jsou mnohem méně náročné na výpočty, dokážeme tím snížit počty tranzistorů v procesorech při zachování stejné úrovně bezpečnosti jako při použití dlouhých klíčů v RSA. Pokud se posílají dlouhé zprávy, které mají být podepsány, tak je zabraná šířka pásma u RSA a ECC úplně stejná. Rozdíl je však tehdy, jsou-li zprávy krátké, kde je ECC už znatelně rychlejší při přenosu těchto zpráv oproti RSA[21]

- Efektivita

Algoritmus RSA může být také rychlý, ale jen za předpokladu, že použijeme krátký klíč, se kterým ovšem podstupujeme značné bezpečnostní riziko. U eliptických křivek můžeme dosáhnout také velké rychlosti, pokud budeme ukládat některé výpočty ještě před tím, než je doopravdy použijeme[21].

Eliptická křivka nám umožňuje definovat si pravidlo pro sečtení dvou bodů, nalézajících se na dané křivce, za účelem zjištění třetího bodu, který se také nachází na křivce. Pro následné sčítání, do kterého můžeme vybrat libovolné dva body, musíme také zahrnout bod nula, který sice nesplňuje rovnici elipsy, ale je brán jako bod nacházející se na křivce.

4.2.3. Použití eliptických křivek

- Výměnu klíčů

Výměnu klíčů mezi uzly A a B můžeme popsat následujícím seznamem činností, které musí uzly provést:[28].

- Uzel A si zvolí číslo $n_A < n$ jako svůj soukromý klíč.
- Dále uzel A vygeneruje svůj veřejný klíč P_A tím, že vynásobí svoje zvolené číslo n_A s bodem P, který náleží eliptické křivce.
- Uzel B si také zvolí své číslo $n_B < n$ jako svůj vlastní soukromý klíč.
- Uzel B si také vytvoří veřejný klíč P_B vynásobením svého zvoleného čísla s bodem P, který opět náleží eliptické křivce.
- Po dokončení těchto činností si uzly navzájem vymění své soukromé klíče. Teď si už může uzel A vytvořit tajný klíč pro komunikaci vynásobením svého zvoleného čísla $n_A$spolu s veřejným klíčem P_B uzlu B. Uzel B provede stejnou činnost jako uzel A. Vynásobí si svoje zvolené číslo n_B s veřejným klíčem P_A uzlu A.

$$n_A < n;\ P_A = n_A * P \qquad n_B < n;\ P_B = n_B * P \qquad K = n_A * P_B = n_B * P_A$$

- Šifrování a dešifrování

Jako první krok je nutné převést otevřená data, která chceme šifrovat, do bloků čísel, která poté převedeme na odpovídající body na eliptické křivce. Pro následné šifrování potřebujeme veřejný klíč, tedy bod Q, dále čistý text a ještě bod P, který také náleží eliptické křivce. V dalším kroku je nutné převést všechny znaky z otevřeného textu do ASCII kódů a spojit je všechny dohromady. Dále si zvolíme soukromý (tajný) klíč d, který si vybereme z intervalu $(1, n-1)$, a následně si vypočítáme bod A tím, že vynásobíme náš tajný klíč s veřejným klíčem $(d * Q)$. Poté je ještě nutné spočítat bod B, který získáme vynásobením našeho tajného klíče s námi zvoleným bodem P, který se nachází na dané křivce $(d * P)$. Přenášenou zprávu (čistý text) si můžeme označit jako T. Zašifrovanou zprávu označíme Cm[23].

Výsledná rovnice, podle které se zašifruje námi zvolený blok dat, má tento tvar:

$$C_m = d * P, T + d * Q \qquad [28]$$

V tomto případě musí uzel A použít při šifrování zprávy, kterou chce odeslat uzlu B, veřejný klíč Q uzlu B.

Při následném dešifrování přijatých bloků dat se tyto data dešifrují následujícím způsobem: přijímací uzel B vynásobí první bod v páru (n_B) s jeho soukromým klíčem a následně odečte ještě výsledek násobení s druhým bodem. Celý postup je zobrazen v následující rovnici:(28)

$$T + d * Q - n_B(d * P) = T + d(n_B * P) - n_B(d * P) = T \qquad [28]$$

- Elektronický podpis

Pro zajištění bezpečného předání relačních klíčů je velmi důležité ověřit identitu dvou komunikujících uzlů a tím zabránit útočníkovi v získání šifrovacích klíčů. Tento proces ověřování pravosti uzlu je klíčový pro zajištění bezpečnosti v senzorické síti.

Z důvodu omezených zdrojů jednotlivých senzorů je nevhodné používat běžné autentizační protokoly, které se běžně používají v internetu a bezdrátových sítí, ale bylo nutné vytvořit nový protokol, který bude brát v potaz odlišnosti senzorických sítích od normálních sítí.

Proto byl navržen protokol, na ověřování identity uzlů, který je založen na algoritmu digitálního podpisu s využitím eliptických křivek (ECDSA – Elliptic Curve Digital Signature Algorithm)[26].

Použitím tohoto protokolu můžeme zabránit škodlivému uzlu, který se vydává za správný uzel v síti, v útoku na síť nebo dokážeme ověřit pravost centra pro přidělování klíčů (KAC – Key Assign Center) a tím účinně zabránit útoku, při kterém se škodlivý uzel maskuje za jiný uzel[26].

4.3. Důvěrnost, pravost a integrita dat

4.3.1. Ověření integrity dat

Pro zajištění bezpečného přenosu dat v síti je zapotřebí zavést šifrování těchto dat a také autentizaci, díky které si uzly mohou ověřit, zda nekomunikují se škodlivým uzlem, ale se správným uzlem. Rozhodnutí, při kterém si vybereme, jaké použijeme šifrovací metody, závisí především na výpočetních a komunikačních schopnostech uzlů v naší síti. Z důvodu omezených zdrojů, které má uzel k dispozici, je použití asymetrické kryptografie často nevhodné kvůli náročnosti a spotřebě energie při šifrování přenášených dat. Proto se také může použít symetrická kryptografie, u které je rychlejší samotné šifrování, ale jsou zde také nevýhody v podobě problému s distribucí a ukládání klíčů[2].

Pokud v síti zavedeme šifrovací mechanismy, které samozřejmě potřebují pro svoje fungování přenos pomocných bitů, tak musíme počítat s tím, že nám zpracování zašifrovaného přenosu zvýší spotřebu paměti i energie. Tyto zdroje (paměť, velikost baterie) nám udávají dobu životnosti jednotlivých uzlů. Zavedení bezpečnostních mechanismů nám také může zvýšit zpoždění při přenosu dat nebo zvýšit počet ztracených paketů[2].

Zvolení asymetrické nebo symetrické kryptografie pro ochranu dat v naší síti závisí především na následujících kritériích: [2]

- Energie – kolik energie bude spotřebováno při šifrování a dešifrování dat
- Programová paměť – kolik paměti nám zabere program pro šifrování a dešifrování
- Dočasná paměť – kolik dočasné paměti (RAM) nebo kolik registrů je potřeba mít při provádění šifrování a dešifrování
- Čas potřebný pro zašifrování a dešifrování zakódovaných dat
- Požadovaná velikost paměti pro uložení potřebného množství klíčů při šifrování a dešifrování

Jedním ze způsobů pro zajištění integrity dat, původu dat a zjištění, zda nedošlo k neoprávněné modifikaci přenášené zprávy, se používají krátké řetězce přidané ke zprávě. Tento řetězec se vytváří pomocí hash funkce, do které je navíc ke zprávě přidán ještě tajný klíč, který je sdílen oběma komunikujícími stranami. Tento klíč, často nazývaný jako předem sdílený klíč (Pre-Shared Key), je tedy nutné před začátkem komunikace rozeslat do všech komunikujících uzlů. Ověření pravosti zpráv je možné díky tomu, že pouze ty uzly, vlastnící předem známý klíč, mohou vygenerovat správný řetězec hash a přiložit ho k přenášené zprávě. Příjemce, který obdrží jak zprávu tak i tento hash řetězec (MAC), si vygeneruje svůj hash řetězec z přijaté zprávy a svého tajného klíče[29].

Pokud zjistí, že se mu nerovnají tyto dva hash řetězce, obdržený a jím vytvořený, dojde k závěru, že v průběhu přenosu došlo k poškození dat nebo k jejich úpravě útočníkem[29].

Do jedné dnes hodně využívané skupiny ověřovacích kódů (MAC) patří také ověřovací kódy HMAC (Hash-based Message Authentication Codes), které používají pro svou funkčnost hash metody schopné využít jakoukoli hash funkci. Mezi tyto hash funkce můžeme zařadit zejména SHA1 nebo MD5, které umožňují vytvoření MAC algoritmů. Další možností pro vytváření MAC algoritmů představují ověřovací kódy postavené na šifrování CMAC (Cipher-based Message Authentication Codes). V případě těchto kódů se pro vytvoření bezpečného MAC algoritmu využívá symetrická bloková šifra. Nejčastěji se pro tyto případy používá šifra AES[29].

Kódy pro ověření integrity se můžou používat třemi způsoby a to, že se před a za zprávu vloží tajný klíč, který zná jen odesílající a přijímací uzel. Z toho se poté vytvoří hash a ten se spolu se zprávou pošle druhému uzlu. Příjemce si doplní ke zprávě stejný tajný kód a vytvoří si hash, který porovná s tím, co přijal. Druhý způsob je ten, že se vytvoří hash zprávy a ten se zašifruje sdíleným klíčem mezi uzly a pošle se také spolu se zprávou druhému uzlu. Z toho vyplývá, že v obou těchto způsobech je nutné, aby uzly, které si chtějí posílat zprávy, vlastnili sdílený tajný klíč. Ale je zde opět problém s jeho bezpečnou distribucí. Třetí způsob už využívá asymetrickou kryptografii a to tak, že vytvořený hash zprávy zašifruje odesílající uzel veřejným klíčem příjímacího uzlu. Díky tomu si tento hash dokáže dešifrovat pouze příjemce za použití svého soukromého klíče.

Požadavky na předem sdílený klíč (Pre-shared Key): [29]

- Společný předem sdílený klíč

  Tento sdílený klíč neumožňuje rozlišit jednotlivé uzly z toho důvodu, že stejný klíč může použít jak odesílatel, tak příjemce. Je ale velmi jednoduché přidat nový uzel do skupiny již existujících uzlů. Hrozí zde ale nebezpečí, že dojde k prozrazení sdíleného klíče, které může útočník využít pro zpětné dešifrování dosud proběhlé komunikace.

- Unikátní předem sdílený klíč

  Je zde použit symetrický klíč mezi každou dvojicí uzlů v síti. Při přidávání nového uzlu do sítě je ale nutné vložit sdílený klíč nového uzlu do všech ostatních uzlů, aby mohl nový uzel komunikovat s ostatními uzly. Toto posléze vede ke špatné udržovatelnosti aktuálních klíčů v systému.

4.3.2. Ověření pravosti pomocí veřejných klíčů

Pro zajištění pravosti dat, tedy zda jsou přijatá data opravdu od správného uzlu, se používá jak symetrická tak i asymetrická kryptografie. V případě symetrického šifrování je zde ale problém s distribucí tajného klíče před samotným šifrováním. Z toho důvodu se pro ověřování používá asymetrická kryptografie s veřejným a tajným klíčem. Příkladem tohoto způsobu ověřování je například elektronický podpis. Tento podpis slouží pro podepisování odesílaných zpráv a zaručuje nám, že odesílatelem těchto dat je opravdu správný uzel a ne škodlivý uzel. Odesílající uzel nejprve vytvoří ze zprávy hash řetězec, který poté zašifruje svým soukromý klíčem. Příjemce těchto dat si z nich také vytvoří hash řetězec a pomocí veřejného klíče rozšifruje přijatý podpis, ze kterého také získá hash řetězec vytvořený odesílajícím uzlem, a tyto dva hash řetězce porovná a zjistí, zda jsou stejné nebo nikoliv. Pokud by nebyly stejné, je možné, že došlo při přenosu k modifikaci přenášených dat škodlivým uzlem. Jestliže ale příjemce zjistí, že se tyto hash řetězce rovnají a zpráva je tedy v pořádku, stále si ale nemůže být jistý, jestli veřejný klíč odesílajícího uzlu opravdu patří tomuto uzlu. Za účelem zjištění důvěryhodnosti podpisu se musí příjemce obrátit na certifikační autoritu, která vydala odesílajícímu uzlu digitální certifikát, který je v podstatě elektronicky podepsaný veřejný klíč a obsahuje potřebné údaje pro identifikaci odesílajícího uzlu. Této autoritě musíme poté důvěřovat, že tento certifikát vydala příslušnému uzlu až po ověření jeho identity.

V senzorických sítích můžou uzly používat předem distribuované klíče nebo dynamicky generovat párové nebo skupinové klíče na základě informací o tvorbě klíčů. Hlavním úkolem je objevit efektivní a bezpečnou cestu pro distribuci klíčů a informací o jejich tvorbě do všech uzlů v síti. Pro vyřešení problému s distribucí klíčů je možné použít některý z těchto tří přístupů: pravděpodobnostní, deterministický anebo hybridní. V pravděpodobnostním přístupu jsou řetězce klíčů náhodně vybírány z fronty již vygenerovaných klíčů (key-pool) a následně distribuovány mezi jednotlivé senzory. V deterministickém řešení jsou používány deterministické procesy pro navrhování řetězců klíčů a jejich následné uložení ve frontách za účelem zlepšení přenosu klíče. Hybridní řešení kombinují používání pravděpodobnostních přístupů s deterministickými řešeními pro zlepšení rozšiřitelnosti a odolnosti[5].

Nachází se zde jedna nebo více výpočetně silných základnových stanic, které mohou fungovat v těchto sítích jako důvěryhodná autorita pro sdílení klíčů (key distribution center). Tato základnová stanice může s každým uzlem v síti sdílet odlišné klíče (každý uzel má svůj vlastní klíč, který zná pouze on a základnová stanice). Tyto klíče můžou být použity pro navázání zabezpečeného spojení mezi dvěma uzly, které spolu chtějí komunikovat a navazují spojení právě přes základnovou stanici (důvěryhodnou autoritu)[5].

Distribuční schéma párových klíčů

Tyto schémata jsou rozdělena podle navrhovaných druhů klíčů, mezi které patří zejména párové klíče, náhodné řetězce klíčů a primární klíče. Obecně se navrhované schémata skládají ze tří fází: nastavení klíče před nasazením uzlu do sítě, sdílení klíče až po zavedení uzlu do sítě, zřízení nového klíče (path-key) v případě, že spolu dva uzly nesdílejí žádný klíč[5].

V bezdrátových senzorických sítích hierarchického typu je potřeba pro jednosměrnou komunikaci mezi základnovou stanicí a uzlem nebo uzlem a základnovou stanicí párový klíč. Základnová stanice proto sdílí s každým uzlem v síti odlišný párový klíč. Díky tomu, že základnová stanice sdílí párový klíč s každým uzlem v síti, může vytvořit nové párové klíče mezi libovolnou dvojicí uzlů. Protokol LEAP (Localized encryption and authentication protocol) navrhuje, aby si každý uzel v síti vytvořil párové klíče se svými sousedními uzly[5].

Distribuční schéma skupinových klíčů

V hierarchických sítích vyžadují uzly pro zajištění bezpečnosti více směrového vysílání skupinové klíče. Jedna z možností jak zabezpečit přenos je použít bezpečnou, ale nákladnou asymetrickou kryptografii. Je ale také možné použít stávající párové klíče pro vytvoření nových skupinových klíčů. Základnová stanice sdílí také se všemi uzly v síti párové klíče a proto je pro ni snadné vytvořit nové skupinové klíče[5].

Všechny certifikáty ve všech uzlech

První možností je rozeslat všechny certifikáty do všech uzlů v síti. Kvůli tomu ale potřebujeme mít ve všech uzlech rozsáhlý prostor pro uložení všech certifikátů ostatních uzlů v síti. S tímto ale přichází problém s obtížným udržováním změn těchto certifikátů, když se třeba do sítě přidá nový uzel, jehož certifikát se musí rozeslat do všech uzlů v síti, nebo když dojde ke změně skupiny u nějakého uzlu a následném přesunu uzlu do jiné skupiny. Pokud by při vstupu nového uzlu nedošlo k rozeslání jeho certifikátu všem uzlům, ostatní uzly by nebyly schopné identifikovat tento nový uzel[29].

Ve všech uzlech je přítomný certifikát certifikační autority

Další možností, jak vyřešit problém s distribucí klíčů, je rozeslání certifikátu důvěryhodné autority do všech uzlů v síti. Následně se veřejné klíče posílají spolu s certifikátem přímo se zprávou. Certifikát, který podepsala certifikační autorita, je možné ověřit za použití statického veřejného klíče této autority. Poté po ověření pravosti certifikátu si uzel může ověřit pravost podpisu zprávy veřejným klíčem certifikátu. Lze také použít zjednodušené SSL (Secure Sockets Layer) pro vytvoření vzájemně ověřeného sdíleného klíče mezi uzly[29].

V hierarchických senzorických sítích je síťový provoz směrem od základnové stanice k uzlům zajištěn a zabezpečen síťovým klíčem (network-wise key). Distribuce jednoho síťového klíče všem uzlům v síti není příliš doporučovaná z důvodu zajištění bezpečnosti[5].

Existují také ale řešení založená na protokolu TESLA (Timed Efficient Stream Loss-tolerant Authentication), který je ověřovací protokol pro více směrové vysílání[5].

Centrální správa klíčů (KDC)

Základní síťová struktura hraje významnou roli ve fungování protokolů na správu klíčů. Na základě síťové struktury můžeme rozdělit protokoly do dvou skupin: centralizované klíčové systémy a distribuované klíčové systémy[3].

- Systém centrální správy klíčů

V centrální správě klíčů se nachází jen jedno centrum pro distribuci, vytváření a pro obnovu klíčů, které je často označováno jako centrum pro distribuci klíčů (key distribution center KDC). Jako jediné centrální schéma pro distribuci klíčů je v současné literatuře uváděné LKHW schéma (Logical Key Hierarchy). Ve schématu LKHW se zachází se základnovou stanicí jako s centrem pro distribuci klíčů (KDC) a všechny klíče, které obsahuje, jsou logicky distribuovány do kořene stromu na základnové stanici[3].

- Systém distribuované správy klíčů

V tomto schématu se používají pro vytváření, obnovu a distribuci klíčů různé řídící jednotky. Rozdělení jednotlivých služeb nám umožňuje snížit riziko poruchy a zlepšit škálovatelnost. V tomto přístupu může selhat více senzorů, než dojde k ovlivnění celé sítě[3].

4.3.3. Zajištění důvěrnosti

Pro zajištění důvěrnosti přenášených dat, a tedy jejich utajení před ostatními uzly v síti, se používá symetrická a asymetrická kryptografie. Symetrická kryptografie se může použít z důvodu rychlejšího šifrování, ale stále je zde velký problém, který nastane v případě, že dojde k vyzrazení šifrovacího klíče. Útočník poté může zpětně dešifrovat celou proběhlou komunikaci v síti. Použitím asymetrické kryptografie tomuto zabráníme, ale stále je nejprve nutné vůbec si ověřit identitu uzlu, se kterým zamýšlíme komunikovat a použít tedy jeho veřejný klíč. Pro ověření, zda jeho veřejný klíč patří opravdu jemu, můžeme použít jeho certifikát a ten si ověřit u důvěryhodné certifikační autority.

4.3.4. Použití certifikátů

Mezi již dobře známé a používané implementace tohoto konceptu můžeme zařadit zejména RSA, které se používá pro zajištění důvěrnosti a pro ověřování, a také DSA (Digital Signature Algotithm), které se používá výhradně jen pro ověřování. Každý uzel při použití těchto kryptografických systémů používá soukromý (tajný) klíč a veřejný klíč, který dává k dispozici všem okolním uzlům. Soukromý klíč je použit pro podepisování zpráv a pro dokázání, že tento uzel vlastní tento určitý klíč. Příjemce za použití veřejného klíče odesílatele ověří pravost jeho podpisu jím odeslané zprávy[29].

Pro úspěšnou identifikaci jednotlivých uzlů v síti za pomoci jejich veřejných klíčů je nutné zajistit provázání jejich klíčů na jejich identitu. Z toho důvodu se používají certifikáty pro ověřování pravosti veřejných klíčů jednotlivých uzlů v síti. Certifikační autorita (CA – Certificate Authority), jako vyšší entita v síti, svým podpisem potvrdí, že veřejný klíč a identita uzlu jsou správné a patří k sobě. Všechny uzly v síti ale musejí důvěřovat této certifikační autoritě[29].

Také existují schémata PKC, která jsou založená na kryptografii eliptických křivek (Eliptic Curve Cryptography), mezi které patří zejména ECDSA (Elliptic Curve Digital Signature Algorithm). Toto schéma založené na eliptických křivkách využívá kratší veřejné klíče než obvykle používané RSA z toho důvodu, že vypočítat diskrétní logaritmus je mnohem obtížnější právě v eliptických křivkách[29].

4.3.5. Kryptograficky generované adresy

Tento systém nám zaručuje, že veřejný klíč patří tomu jistému partnerovi, se kterým komunikujeme. Toto ověření se provádí díky přítomnosti otisku (hash) veřejného klíče v síťové IPv6 adrese. V této adrese je hash řetězec umístěn na nižších 62 bitech této adresy. Tento mechanismus CGA byl vytvořen primárně pro ověřování v průběhu objevování sousedních uzlů. Princip fungování spočívá v tom, že veřejný klíč zaslaný spolu se zprávou může být ověřen porovnáním poslaného veřejného klíče s hash řetězcem v síťové adrese odesílajícího uzlu, která obsahuje také jeho veřejný klíč[29].

Jestliže zavedeme v síti ověřování pomocí CGA, tak již není nutné mít nadále v síti certifikační autority. Nachází se zde ale také několik problémů spočívající v tom, že CGA není schopné se certifikovat samo a že každý si může vytvořit nové validní CGA pro svoji podsíť i když bude vy výsledku jiná adresa. Dále se CGA dokáže celkem účinně bránit před útoky hrubou silou (brute-force attacks) na získání páru veřejného a soukromého klíče. Pokud tyto klíče útočník získá, tak si může pro tyto klíče vytvořit stejné CGA, jaké již používají jiné uzly v síti. Tato obrana je realizována pomocí parametru *sec*, který stanovuje požadavek na to, aby určité množství bitů v hash řetězci obsahovalo jen náhodná čísla a nulové hodnoty v místě, kde je umístěn veřejný klíč[29].

4.3.6. Podpisy založené na identitě uzlu

Myšlenka vychází z předpokladu, že veřejným klíčem může být téměř cokoliv, tedy i text identifikující uživatele, např. jeho email, IP/Ethernet adresa, MAC adresa a tak. Odpovídající tajný klíč je posléze generován společnou důvěryhodnou autoritou, která bývá nejčastěji označovaná jako TA (Trusted Authority). Následně jsou všechny soukromé (tajné) klíče rozeslány do každého ověřeného uzlu v síti. Pro ověření pravosti podpisu potřebujeme jen veřejně známé informace a to tyto: veřejný klíč odesílajícího uzlu, zprávu a podpis[29].

4.3.7. Schémata ověřování a ukládání klíčů

Použitím asymetrické kryptografie, která se stala díky své bezpečnosti velmi používaná v senzorových sítích pro ověřování vysílané všem uzlům, dokáže uzel ověřit přijatou zprávu ještě před jejím odesláním na jiný uzel a odhalit tak třeba falešnou zprávu v prvním přeskoku mezi odesílajícím uzlem a jím. Ale je zde také nevýhoda v náročnosti výpočtů prováděných při šifrování a dešifrování, které je náročné pro senzory mající k dispozici baterii s relativně omezenou kapacitou. Právě z důvodu slabé baterie a slabého výpočetního výkonu se může stát, že bude docházet k prodlevám a zpožděním při doručování zpráv ostatním uzlům[27].

Je zde také možnost, že uzel předá obdrženou zprávu ostatním uzlům, aniž by jí vůbec ověřoval. Uzel tímto značně snižuje zpožďování jím přeposílaných zpráv na úkor ověřování jejich odesílatele. Samozřejmě se musí uzel správně rozhodnout, zda bude kontrolovat odesílatele jím přijatých paketů nebo tyto pakety rovnou předá bez ověřování toho, kdo je poslal. Tímto počínáním může uzel dosáhnout rychlého a efektivního přeposílání zpráv v rámci sítě. Proto byly vytvořeny dvě nová schémata pro možnost ověřování více uzly (broadcast). První z nich je prostor, ve kterém jsou uloženy příslušné klíče (key pool) a v případě druhého se jedná o vkládání jednosměrných hash řetězců kvůli zabránění předvídání těchto klíčů útočníkem[27].

- Výběr šifrovacích klíčů z úložiště těchto klíčů (Key pool scheme)

V tomto schématu vlastní každý uzel v síti jen podmnožinu ze všech uložených klíčů v daném úložišti. Přístupový bod má naopak k dispozici veškeré tajné klíče nacházející se v síti. Díky tomuto rozdělení neobsahuje každý uzel všechny klíče, díky čemu se minimalizuje riziko získání klíčů útočníkem, který by tento uzel obsadil a získal z něj uložená data. Toto schéma obsahuje základní předpoklad, že pouze přístupový bod bude mít umožněn přístup ke globálnímu úložišti klíčů. Dále se vychází z toho, že každý uzel bude mít dostatek paměti pro uložení určitého počtu klíčů přímo u sebe. Toto schéma se skládá ze tří částí a to z fáze přerozdělení, fáze generování podpisů a fáze ověřování a přeposílání zpráv[27].

V první fázi si každý uzel před svým nasazením do sítě nahraje do své paměti své lokální klíče, které tvoří podmnožinu klíčů vlastněných přístupovým bodem. V další fázi se určí způsob, jakým můžou být generovány zprávy, posílané do všech uzlů (broadcast), a podpisy. Třetí a poslední fáze se zabývá ověřením a rozhodnutím o přeposílání zpráv, pokud dorazí zprávy vysílané do všech uzlů (broadcast message)[27].

- Zřetězení klíčů (Key chain scheme)

Úkolem tohoto schématu je co nejvíce snížit komunikační režii v úložišti klíčů (Key pool scheme) používáním jednosměrných hash řetězců. Díky tomu, že každý hash řetězec je jiný, dokážeme tím účinně bránit možnému předpovídání klíčů útočníkem. Dále díky hash řetězcům nemusíme vkládat indexy klíčů do každé zprávy vysílané do sítě. Globální úložiště klíčů, které má velikost N klíčů, je použito pro uložení N nezávislých nedělených řetězců klíčů. V síti je používána hash funkce H a výchozí klíč K_0, což nám umožní najít klíč v řetězci K_i na pozici i za použití této rovnice $K_i = H(K_{i-1})$ [27].

Běžně se provádí generování jednosměrného hash řetězce před tím, než je vůbec zahájen proces ověřování, ve kterém se použije poslední klíč, tedy klíč na nejvyšší pozici v řetězci vytvořených klíčů. Z tohoto řetězce klíčů se používají klíče až do té doby, dokud nedosáhnou posledního zbývajícího klíče K_0 a až poté jsou teprve vygenerovány nové řetězce klíčů a rozeslány do sítě. Systém řetězců klíčů (Key chain scheme) používá ale na rozdíl od běžného přístupu vybírání klíčů od začátku řetězce a ne od jeho konce. To znamená, že každý uzel začíná od klíče s indexem 0 a poté pokračuje postupně ke klíčům s vyššími indexy. Díky tomu není nutné, aby se tento protokol zabýval rozesíláním klíčů jednotlivým uzlům. Také není vyžadována přítomnost indexů klíčů ve zprávách rozesílaných všem uzlům. Nachází se zde ale také jedna velmi podstatná nevýhoda spočívající v tom, že pokud nastane situace, že bude jeden klíč získán útočníkem, tak dojde k ohrožení celého řetězce těchto klíčů[27].

4.3.8. Bezpečná agregace dat

V bezdrátových senzorických sítích je nutné zabezpečit přenos naměřených informací jednotlivými uzly až k agregačnímu uzlu. Tím, že zavedeme pro zajištění tohoto přenosu agregaci dat, snížíme tím množství přenášených dat od uzlů k agregační stanici. Díky tomu zvýšíme energetickou účinnost a také prodloužíme životnost bezdrátové sítě. Z důvodu častého nasazení senzorických sítích do nepřátelských oblastí je velice pravděpodobné, že bude docházet ke vkládání falešných zpráv či falšování agregačních hodnot. Proto je kladen veliký důraz na výzkum a následné řešení možných bezpečnostních rizik v těchto sítích[16].

- Zaváděcí fáze

V této fázi se provádí nastavování celé sítě a distribuce klíčů mezi jednotlivé uzly. Toto rozesílání klíčů je důležité pro správné fungování dalších fází, mezi které patří agregace dat a proces ověřování posílaných dat. Samozřejmě s rozesíláním klíčů také souvisí zřejmý problém, kdy se může útočník k těmto klíčům dostat. Běžně se v senzorických sítích používají klíče, které sestávají z fixních a generovaných klíčů. Stálé (fixní) klíče nacházejí uplatnění v sítích, ve kterých je kladen důraz na rychlé šifrování a dešifrování a proto jsou tyto klíče v každém uzlu již od jeho nasazení nebo jsou vytvořeny postupy, které ale znají jen legitimní uzly. Náhodně vytvářené klíče se naopak používají tam, kde není potřeba, aby šifrování a dešifrování bylo rychlé. Distribuce těchto klíčů je možná třemi způsoby: párovými, skupinovými a síťovými klíči[16].

- Fáze agregace dat

Tato fáze má za úkol zajistit zvolení správného způsobu směrování a zabezpečeného sběru naměřených dat. V prvním schématu, kdy k přenosu dat dochází předáváním těchto dat mezi sousedními uzly (hop by hop), agregační uzel tyto data sbírá v otevřené podobě (dešifrovaná data) a následně šifruje jím sesbíraná data. Z toho je zřejmé, že tento agregační uzel musí sdílet šifrovací klíč s těmi uzly, od kterých přijímá data. V dalším schématu, kde jsou data přenášena od zdrojového až k cílovému uzlu, agregační uzel přímo shromažďuje naměřená data z jednotlivých uzlů v síti v zašifrované podobě, aniž by k těmto datům vlastnil dešifrovací klíč[16].

Volba směrování ve schématu přeskoku mezi sousedními uzly (hop by hop) nenabízí tolik možností kudy vést přenos dat z důvodu, že námi sesbíraná data můžeme poslat pouze agregačnímu uzlu nebo sousednímu uzlu, které ale musejí mít platné klíče a těch v síti není mnoho.

Na rozdíl od tohoto schématu je možné v schématu end to end volit směrovací trasy daleko volněji, protože nejsou v agregační fázi potřeba žádné klíče a k dešifrování dochází až v koncovém uzlu. Kvůli prodloužení životnosti uzlů a potažmo celé sítě je možné, aby se vícero uzlů stalo agregačním uzlem a ušetřilo tak energii uzlům, které by nemusely zatěžovat jen jeden agregační uzel. Je zde také pamatováno na případ, kdy dojde k narušení jednoho agregačního uzlu útočníkem, při kterém se prostě tento uzel přestane používat a přejde se na jiné agregační uzly, které normálně fungují[16].

- Ověřovací fáze

Po dokončení této fáze musí být ověřena pravost jak samotných uzlů, tak i jejich dat odesílaných do sítě. Jednou z možností, jak detekovat falešná data posílaná do sítě, je sdílení statistické funkce mezi jednotlivými uzly. Nebo je dále možné se rozhodnout, zda tyto data spadají do určitého rozsahu a podle toho rozpoznat falešná data, která nepatří do tohoto rozsahu. Je také nutné, aby byly správné data také aktuální a zobrazovaly by tedy co nejaktuálnější stav celé sítě. Pro zajištění před případným útokem, který by znovu odesílal již poslaná data (replay attack), se často používá časová značka nacházející se v přenášených datech za účelem zajištění aktuálnosti těchto dat. Proto je důležité, aby se síť ujistila, že zdroj dat je pravý. Pro toto ověření používá většina protokolů pro rozpoznání správného uzlu jeho MAC nebo jeho ID[16].

- Fáze nápravy a obnovení

V senzorických sítích se většina agregačních protokolů zaměřuje zejména na prevenci před útokem a jeho odhalení než na důležitou, ale často opomíjenou část, a to na jeho nápravu. Agregační protokol by měl být schopný zajistit to, aby základnová stanice přijímala, i přes probíhající útok na senzorovou síť, stále správné výsledky agregace dat. Z toho důvodu je nutné spustit nápravné mechanismy poté, co dojde v sítí k objevení útočníka. Můžeme tedy označit tuto nápravnou fázi jako další inicializační fázi z toho důvodu, že se i při této fázi provádí obnovení struktury celé sítě, znovu se rozesílají šifrovací klíče a také se vymažou všechny informace pocházející od škodlivého uzlu[16].

4.4. Obrana proti vybraným útokům

4.4.1. Černá díra (Black hole)

Vícerozměrný příznakový vektor je definován v každém uzlu pro vyjádření stavu sítě. Každá ze všech dimenzí se počítá na každém časovém slotu. V důsledku odhalení útoku Black hole je každému cílovému sekvenčnímu číslu přikládána velká váha a důležitost. Při běžném provozu se sekvenční číslo každého uzlu mění v závislosti na síťových podmínkách. Pokud se začne počet spojení zvyšovat a začne také stoupat cílové sekvenční číslo, které se ale bude zvyšovat postupně v případě malého počtu spojení. Jestliže však k útoku dojde, tak se bez ohledu na prostředí, sekvenční číslo značně zvětší. Většinou se počet odeslaných RREQ paketů rovná počtu přijatých RREP paketů. Z těchto důvodů se používají tyto funkce k vyjádření stavu sítě[10].

- Počet zaslaných RREQ (Route Reply Question)zpráv
- Počet obdržených RREP zpráv
- Průměrný počet Dst- sekvencí v každém časovém slotu mezi pořadovým číslem RREP zprávy a jednou zprávou, která zbyla v seznamu.

Každý uzel si zapíše cílovou IP adresu a Dst_Seq (hodnota je používána k určení aktuálnosti směrovacích informací, které jsou obsažené ve zprávě od odesílajícího uzlu) do svého seznamu při odesílání a přijímání RREQ zpráv. Uzel se po obdržení RREP zprávy podívá do svého seznamu, jestli se tam nenachází stejné cílové IP adresy. Pokud tam existuje, tak se vypočte rozdíl od Dst_Seq. Tuto operaci provádí uzel pro každou obdrženou RREP zprávu. Průměrná hodnota tohoto rozdílu se nakonec vypočte funkcí pro každý časový slot[10].

Vědci vytvořily různé techniky pro zabránění blackhole útoku v mobilních ad-hoc sítích. H. Weerasinghe and H. Fu [1] představili použití DRI (Data Routing Information) pro sledování historie směrování v síti mezi mobilními uzly a křížovou kontrolu RREP (route reply paket) zpráv z mezilehlých uzlů, kterou provádějí zdrojové uzly. Mezi hlavní nevýhodu této techniky patří skutečnost, že každý mobilní uzel si musí udržovat databázi historie směrování společně se svojí směrovací tabulkou. Z toho je patrné, že udržování směrovací historie je náročné na paměť, stejně jako je časově náročné samotné zpracovávání těchto hodnot a dochází potom ke zpomalování komunikace.

Druhou nevýhodou tohoto řešení je přílišná spotřeba omezené šířky pásma. Křížová kontrola platnosti cest, které obsahuje RREP zpráva z mezilehlého uzlu, je prováděna zasíláním FREQ (Further Request) zpráv do dalšího směrování konkrétního mezilehlého uzlu. Odesílání dalších zpráv spotřebovává značnou část šířky pásma z již tak omezených a drahých zdrojů[1].

Výzkum je podobný Weerasingheově technice kromě dodatečné slabiny, neschopnosti zabránit útoku z několika uzlů černých děr. P.Raj a P. Swadas[3] navrhli vhodné řešení na kontrolu RREP paketů z průběžných uzlů pro objevení možného narušení aktivity. Tato technologie je založena na předpokladu spolupráce mezi jednotlivými uzly. V případě, že mobilní uzel zjistí možný útok od narušitele, tak odešle všem uzlům zprávu, kterou je varuje před tímto útokem. Proces varování je velmi pomalý a vyžaduje velké množství času pro varování všech uzlů v rozsáhlé síti a mimo jiné tento proces ještě způsobí značné zatížení sítě rozesíláním varující zprávy[1].

4.4.2. Červí díra (Wormhole)

Mezi různé způsoby obrany a detekce proti Wormhole útoku patří zejména packet leash (paketové vodítko). Jedná se o obecný postup pro detekování a následnému bránění proti Wormhole útoku. Leash je sám o sobě doplňková informace obsažená v každém paketu kvůli omezení maximální povolené přenosové vzdálenosti paketu. Leashes jsou navrženy pro ochranu před Wormhole útoky přes jediný bezdrátový přenos. Pokud jsou pakty zaslány přes více uzlů najednou, tak je vyžadováno pro každé spojení použití nové doplňkové informace (leash).

Nacházejí se zde dva druhy doplňkových informací:

- geografické informace
  - zajišťují určitou vzdálenost příjemce paketu od jeho odesílatele
- časové omezení
  - udává maximální dobu životnosti paketu, což omezuje celkovou vzdálenost trasy, kterou paket může urazit.

Oba tyto typy informací (razítek) umožňují se bránit proti Wormhole útoku díky tomu, že příjemce paketů si z těchto razítek zjistí, jestli konkrétní paket neputoval dále, než mu povolovalo časové razítko[35].

Další způsob detekce červí díry je pomocí informací o spojení. Síťové spojení ovlivňuje zejména umístění této červí díry, která vytváří dlouhá spojení mezi dvěma skupinami od sebe vzdálených uzlů. Konečný graf spojení se proto liší od skutečného grafu spojení. Vědci z Stony Brook University vyvinuli svůj vlastní algoritmus, který se snaží najít zakázané substruktury v grafu spojení, které by se neměli nacházet ve správném grafu spojení.

Jejich algoritmu, na detekci Wormhole, hodně pomáhá znalost bezdrátových sítích. Díky tomu může komunikační model pomoci stanovit substruktury, které mohou být ve sledovaném grafu spojení zakázány. Nicméně jejich přístup zůstává platný, pokud je neznámý komunikační model[36].

Nejprve museli vědci vyvinout jejich vlastní detekční algoritmus, který začíná grafovým modelem diskové jednotky, dále pokračuje obecnou komunikací (známou nebo neznámou) a nakonec končí projednáním, jak odstraňovat automaticky vytvořené odkazy od Wormhole uzlu jakmile je tento škodlivý uzel detekován[36].

Dále se pro detekci Wormhole útoku používá End-to-end detekce (EDWA – End-to-end Detection of Wormhole Attack). V této části se zaměřím na popsání detekčních a preventivních mechanismů EDWA. Jako první přijde na řadu popis hlavního návrhu EDWA detekčních metod. Mezi tři kroky patří následující postupy: zdrojový uzel detekuje wormhole útok na základě odhadu nejkratší cesty v každé části sítě. Jakmile uzel zjistí wormhole útok, tak spustí sledovací fázi pro objevení dvou koncových bodů wormhole útoku. Následně zdrojový uzel zvolí nejkratší cestu ze správných tras, které jsou určeny pro datovou komunikaci[37].

4.4.3. Přeposílání jen vybraných zpráv (Selective Forwarding)

Autoři z National Institute of Technology Rourkela uvádějí na základě jejich předchozích výzkumů tyto způsoby pro zmírnění rizika do těchto způsobů:

1) Schéma, které je schopné samostatně detekovat škodlivý uzel a vyloučit ho ze síťového provozu.

- Detekce založená na potvrzování
- Detekce škodlivého uzlu pomocí informací poskytnutých sousedními uzly

2) Schéma, které používá ke zmírnění následků útoku multi-data flow.

Dále se autoři zabývají hlavně možností, jak se vyhnout ztrátě paketů pomocí multi-data flow, místo zaměřování se na detekci útoku a škodlivého uzlu. V každém detekčním schématu se samozřejmě nacházejí různé přednostní požadavky. Předtím než se podrobněji podíváme na schémata, která nám pomohou zmírnit případný útok, si nejprve musíme uvést několik předpokladů a požadavků, které jsou potřeba v každém detekčním schématu:

- V síti se musí nacházet zabezpečená komunikace mezi jednotlivými uzly.
- Uzly nesmí být kompromitovány v průběhu zaváděcí fáze.
- Vzhledem k nespolehlivému přenosu dat v bezdrátových sítích musí být pokles hodnoty zahozených paketů větší než je obvykle, za účelem odlišení od útoku, který by jen zahazoval pakety[38].

Mezi další možnosti jak detekovat a zabránit Selective forwarding patří detekční schéma založené na potvrzení události okolními uzly. Vědci z univerzit v Hongkongu a Číně přišli se svým detekčním schématem, ve kterém každý mezilehlý uzel nacházející se podél přenosové cesty má za úkol detekovat škodlivý uzel. Jestliže mezilehlý uzel rozpozná špatné chování předchozího nebo následujícího uzlu, tak vytvoří poplašný paket a zašle ho zdrojovému uzlu přes několik mezilehlých uzlů nacházejících se mezi mezi nimi. Autoři označují v jejich práci *downstream* jako přenos k základní stanici a *upstream* směrem ke zdrojovému uzlu. Pro učinění rozhodnutí a vykonání patřičné odpovědi na útok můžou základnová stanice a zdrojový uzel použít více složitých IDS (Intrsion Detection Systems) algoritmů[39].

5. Závěr

Cílem mojí práce bylo nalezení útoků na bezdrátové senzorické sítě a jejich zařazení do skupin podle zranitelností, které tyto útoky využívají pro svoji činnost. Ale nejprve jsem na začátku uvedl informace a základní popis senzorických sítích a jak vlastně vůbec fungují. Poté jsem se snažil dát dohromady co nejvíce útoků, které existují na senzorické sítě. Dále jsem u těchto nalezených útoků hledal zranitelnosti nebo vlastnosti směrovacích protokolů, díky kterým mohou útočit na síť. Tyto útoky jsem dále roztřídil podle vrstev používaných v senzorických sítích (jsou velmi podobné vrstvám z OSI/ISO modelu) a v každé této vrstvě jsou příslušné útoky dány do skupin podle jejich společné slabiny, kterou využívají. Útoků jsem našel opravdu hodně, ale nakonec jsem zjistil, že těchto hodně útoků využívá jen několik chyb pro svoji činnost. Většina těchto útoků si je hodně podobných a liší se jen v detailech.

Ve druhé části práce jsem se věnoval možnostem, jak zabezpečit přenos dat mezi jednotlivými uzly senzorické sítě. Jako první možnost je zde samozřejmě zavedení šifrování, které je účinné, ale musí se zvolit takové algoritmy, aby nespotřebovávali zbytečně mnoho energie a výpočetní výkon uzlu. Z tohoto hlediska jsem se proto více zaměřil na šifrování pomocí eliptických křivek, které se začínají používat v čím dál větší míře zejména kvůli jejich menší energetické náročnosti a kratší délce šifrovacích klíčů, které snižují výpočetní náročnost. Dále uvádím přehled možností, jak pomocí šifrování zajistit integritu dat, ověření pravosti uzlu a také důvěrnost těchto dat. Poté jsem se věnoval možnostem pro bezpečné sdílení klíčů v těchto sítích a také bezpečné agregaci dat z jednotlivých uzlů. V případě agregace se jedná o sběr dat naměřených samotnými uzly do sběrných uzlů, které jsou označovány jako agregační uzly.

V poslední podkapitole jsem uvedl několik konkrétních metod pro obranu před třemi vybranými útoky.

Literatura

[1] Alem, Y.F., Zhao Cheng Xuan; „Preventing black hole attack in mobile ad-hoc networks using Anomaly Detection“; Future Computer and Communication (ICFCC), 2010 2nd International Conference on (Volume:3); Wuhan, 21-24 May 2010, ISBN 978-1-4244-5821-9

[2] T.Kavitha, D.Sridharan; „Security Vulnerabilities In Wireless Sensor Networks: A Survey“; Journal of Information Assurance and Security 5 (2010) 031-044; 1554-1010 $ 03.50 Dynamic Publishers, Inc.

[3] Yong Wang, Garhan Attebury, Byrav Ramamurthy; „A Survey Of Security Issues In Wireless Sensor Networks“; Ieee Communications Surveys & Tutorials • 2nd Quarter 2006; 1553-877X

[4] Hani Alzaid, Ernest Foo, Juan Gonzalez Nieto; "Secure data aggregation in wireless sensor network: a survey“; AISC '08 Proceedings of the sixth Australasian conference on Information security - Volume 81; ISBN 978-1-920682-62-0

[5] Seyit A. C¸Amtepe and Bulent Yener; „Key Distribution Mechanisms for Wireless Sensor Networks: a Survey“; TR-05-07, Department of Computer Science, Rensselaer Polytechnic Institute.

[6] Kai Xing, Shyaam Sundhar Rajamadam Srinivasan, Manny Rivera, Jiang Li, Xiuzhen Cheng; „Attacks and Countermeasures in Sensor Networks: A Survey“;

[7] Bing Wu, Jianmin Chen, Jie Wu, Mihaela Cardei; „A Survey of Attacks and Countermeasures in Mobile Ad Hoc Networks“; ISBN 978-0-387-33112-6

[8] Bing Wu, Jianmin Chen, Jie Wu, Mihaela Cardei; „A Survey on Attacks and Countermeasures in Mobile Ad Hoc Networks“; Springer 2006

[9] K.P.Manikandan, Dr.R.Satyaprasad, Dr.K.Rajasekhararao; „A Survey on Attacks and Defense Metrics of Routing Mechanism in Mobile Ad hoc Networks“; International Journal of Advanced Computer Science and Applications, Vol. 2, No.3, March 2011;

[10] Satoshi Kurosawa, Hidehisa Nakayama, Nei Kato, Abbas Jamalipour, Yoshiaki Nemoto; „Detecting Blackhole Attack on AODV-based Mobile Ad Hoc Networks by Dynamic Learning Method“; International Journal of Network Security, Vol.5, No.3, PP.338–346, Nov. 2007

[11] Aniruddha Bhattacharyya, Arnab Banerjee, Dipayan Bose, Himadri Nath Saha, Debika Bhattacharjee; „Different types of attacks in Mobile ADHOC Network: Prevention and mitigation techniques“; Department of Computer Science & Engineering, Institute Of Engineering & Management, Saltlake

[12] Sheela, D., Naveen, K.C. ; Mahadevan, G.; „A non cryptographic method of sink hole attack detection in wireless sensor networks“; Recent Trends in Information Technology (ICRTIT), 2011 International Conference; 3-5 June 2011; ISBN 978-1-4577-0588-5

[13] Janne Lundberg; „Routing Security in Ad Hoc Networks“; Helsinki University of Technology 2000, Tik-110.501 Seminar on Network Security

[14] Chris Karlof David Wagner; „Secure Routing in Wireless Sensor Networks: Attacks and Countermeasures“; University of California at Berkeley

[15] Shaohe Lv, Xiaodong Wang ; Xin Zhao ; Xingming Zhou; „Detecting the Sybil Attack Cooperatively in Wireless Sensor Networks“; Computational Intelligence and Security, 2008. CIS '08. International Conference on (Volume:1), 13-17 Dec. 2008; ISBN 978-0-7695-3508-1

[16] Jia Guo, Jian'an Fang, Xuemin Chen; „Survey on secure data aggregation for wireless sensor networks“; Service Operations, Logistics, and Informatics (SOLI), 2011 IEEE International Conference, 10-12 July 2011, ISBN 978-1-4577-0573-1

[17] Aniruddha Bhattacharyya, Arnab Banerjee, Dipayan Bose, Himadri Nath Saha Debika Bhattacharjee; „Different types of attacks in Mobile ADHOC Network: Prevention and mitigation techniques“; Department of Computer Science & Engineering, Institute Of Engineering & Management, Saltlake

[18] Prachi Bansal, Beenu Yadav, Sonika Gill, Harsh Verma; „Security Attacks in Wireless Sensor Network“; International Journal of Scientific & Engineering Research, Volume 3, Issue 4, April-2012 1, ISSN 2229-5518

[19] Ahmed, M.H., Alam, S.W., Qureshi, N., Baig, I.; „Security for WSN based on elliptic curve cryptography“; Computer Networks and Information Technology (ICCNIT), 2011 International Conference, 11-13 July 2011, ISBN 978-1-61284-940-9

[20] Bai Qing-hai, Zhang Wen-bo ; Jiang Peng ; Lu Xu; „Research on Design Principles of Elliptic Curve Public Key Cryptography and Its Implementation“; Computer Science & Service System (CSSS), 2012 International Conference, 11-13 Aug. 2012, ISBN 978-1-4673-0721-5

[21] Vivek Kapoor, Vivek Sonny Abraham, Ramesh Singh; „Elliptic Curve Cryptography“; ACM Ubiquity, Volume 9, Issue 20 May 20 – 26, 2008

[22] Eliška Ochodková; „Přínos teorie eliptických křivek k řešení moderních kryptografických systémů“; Katedra informatiky, FEI, VŠB - Technická Univerzita Ostrava

[23] Lukáš Geyer; „Eliptické křivky v kryptografii“; Vysoké učení technické v Brně, Bakalářská práce

[24] Ing. Tomáš Vaněk, Ph.D., „Kryptosystémy na bázi eliptických křivek“; Fakulta elektrotechnická ČVUT Praha

[25] Xia Wang, Johnny Wong; „An End-to-end Detection of Wormhole Attack in Wireless Ad-hoc Networks“; 31st Annual International Computer Software and Applications Conference(C0MPSAC 2007), 0-7695-2870-8107 $25.00

[26] Wang Wei-hong, Cui Yi-ling ; Chen Tie-ming; „Design and implementation of an ECDSA-based identity authentication protocol on WSN“; Microwave, Antenna, Propagation and EMC Technologies for Wireless Communications, 2009 3rd IEEE International Symposium, 27-29 Oct. 2009, ISBN 978-1-4244-4076-4

[27] Chuchaisri, P., Newman, R., „Fast response PKC-based broadcast authentication in wireless sensor networks“; Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom), 2010 6th International Conference, 9-12 Oct. 2010, ISBN 978-963-9995-24-6

[28] Asha Rani Mishra, Mahesh Singh; „Elliptic Curve Cryptography (ECC) for Security in wireless Sensor Network“; International Journal of Engineering Research & Technology (IJERT), Vol. 1 Issue 3, May – 2012, ISSN 2278-0181

[29] Tobias Markmann, „Authentication Schemes for Wireless Sensor Nodes at a Glance“; July 23, 2013

[30] Shi Lan, Miao Qilong, Jinglin Du; „Architecture of Wireless Sensor Networks for Environmental Monitoring“; Education Technology and Training, 2008. and 2008 International Workshop on Geoscience and Remote Sensing. ETT and GRS 2008. International Workshop on (Volume:1), 21-22 Dec. 2008, ISBN 978-0-7695-3563-0

[31] Mo Li, Baijian Yang; „A Survey on Topology issues in Wireless Sensor Network“

[32] Steven Kosmerchock; „Wireless Sensor Network Topologies“;

[33] Ahmad Abed Alhameed Alkhatib, Gurvinder Singh Baicher; „Wireless Sensor Network Architecture“; 2012 International Conference on Computer Networks and Communication Systems (CNCS 2012)

[34] Fei Hu, Xiaojun Cao; „Wireless Sensor Networks: Principles and Practice“; Chapter 5: Transport Layer in Wireless Sensor Networks

[35] Yih-Chun Hu, Perrig, A., Johnson, D.B.; „Packet leashes: a defense against wormhole attacks in wireless networks“; INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies (Volume:3), 30 March-3 April 2003, ISBN 0-7803-7752-4

[36] Maheshwari, R., Jie Gao, Das, S.R.; „Detecting Wormhole Attacks in Wireless Networks Using Connectivity Information“; INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE, 6-12 May 2007, ISBN 1-4244-1047-9

[37] Xia Wang, Johnny Wong; „An End-to-end Detection of Wormhole Attack in Wireless Ad-hoc Networks (2007)“; 37 Computer Software and Applications Conference, 2007. COMPSAC 2007. 31st Annual International, 24-27 July 2007, ISBN 0-7695-2870-8

[38] Bysani, L.K., Turuk, A.K.; „A Survey on Selective Forwarding Attack in Wireless Sensor Networks“; Devices and Communications (ICDeCom), 2011 International Conference, 24–25 Feb. 2011, ISBN 978-1-4244-9189-6

[39] Bo Yu, Bin Xiao; „Detecting selective forwarding attacks in wireless sensor networks“; Parallel and Distributed Processing Symposium, 2006. IPDPS 2006. 20th International, 25-29 April 2006, ISBN 1-4244-0054-6