VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV INFORMAČNÍCH SYSTÉMŮ
DEPARTMENT OF INFORMATION SYSTEMS

ANALÝZA BEZPEČNOSTI ZAŘÍZENÍ V CHYTRÉ DOMÁCNOSTI

SECURITY ANALYSIS OF SMART HOME DEVICES

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE **PETER GROFČÍK**
AUTHOR

VEDOUCÍ PRÁCE **Ing. PETR MATOUŠEK, Ph.D., M.A.**
SUPERVISOR

BRNO 2018

Vysoké učení technické v Brně
Fakulta informačních technologií

Ústav informačních systémů (UIFS) Akademický rok 2018/2019

Zadání bakalářské práce

Student: **Grofčík Peter**
Program: Informační technologie
Název: **Analýza bezpečnosti zařízení v chytré domácnosti**
Security Analysis of Smart Home Devices
Kategorie: Počítačové sítě

Zadání:

1. Seznamte se s principy komunikace zařízení v chytré domácnosti.
2. Nainstalujte si dostupné řešení pro vytvoření chytré domácnosti (senzory, kamery, kontrolér). Popište použitá zařízení, způsob přístup k zařízení, typy komunikace a možnost vzdáleného řízení systému.
3. Proveďte analýzu bezpečnosti jednotlivých zařízení chytré domácnosti: popište způsob přístupu, dostupné možnosti zabezpečení, možné zranitelnosti.
4. Podle doporučení vedoucího práce vyzkoušejte různé kybernetické útoky na chytrou domácnost. Popište nalezená zjištění.
5. Navrhněte možné způsoby monitorování chování prvků v chytré domácnosti s cílem detekovat tyto útoky.
6. Zhodnoťte výsledky své práce.

Literatura:

- David Hanes, Gonzalo Salqueiro, Patrick Grossetete, Rob Barton, andJereme Henry. IoT Fundamentals. Networking Technologies, Protocoland Use Cases for the Internet of Things. Cisco Press, 2017.
- Joshua I. James, Petr Matoušek, Ondřej Ryšavý: Data Acquisition in the Internet of Things, WIRE Forensics, 2019.
- Me ert, C., Clark, D., Baggili, I., & Breitinger, F. (2017). Forensic State Acquisition from Internet of Things (FSAIoT): A General Framework and Practical Approach for IoT Forensics Through IoT Device State Acquisition. In Proceedings of the 12th Int. Conference on Availability, Reliability and Security (pp. 56:1{56:11).

Pro udělení zápočtu za první semestr je požadováno:

- Body 1-3.

Podrobné závazné pokyny pro vypracování práce viz http://www.fit.vutbr.cz/info/szz/

Vedoucí práce: **Matoušek Petr, Ing., Ph.D., M.A.**
Vedoucí ústavu: Kolář Dušan, doc. Dr. Ing.
Datum zadání: 1. listopadu 2018
Datum odevzdání: 15. května 2019
Datum schválení: 16. října 2018

Abstrakt

Hlavným cieľom mojej bakalárskej práce je prevedenie analýzy bezpečnosti zariadení inteligentnej domácnosti a následne otestovať a overiť zraniteľnosti odhalené na týchto zariadeniach. Prvá časť popisuje zariadenia zo sady MioSMART použité na analýzu vrátane spôsobov prístupu na ne. V ďalšej časti sa nachádza popis komunikácie zariadení pre ich funkcionalitu v rámci inteligentnej domácnosti, a to vrátane protokolov, ktoré na danú komunikáciu využívajú. V ďalšej kapitole sú použité voľne dostupné nástroje, za účelom odhalenia jednotlivých zraniteľných miest v komunikácii. Na základe týchto informácií boli prevedené útoky na najzávažnejšie odhalené zraniteľnosti, za účelom dokázania možnosti ich zneužitia na poškodenie funkcionality zariadení inteligentnej domácnosti. Časť tejto kapitoly je tiež venovaná popisu spôsobov monitorovania jednotlivých prvkov sady pre odhalenie prebiehajúcich útokov.

Abstract

Main purpose of my bachelor thesis is to analyze security of smart home devices and afterwards to test and confirm exposed vulnerabilities on these devices. The first section describes MioSMART kit devices used for analysis including ways, how to access them. The next section describes communication of smart home devices including protocols, that are used for it. In the next chapter, free tools are introduced to identify vulnerabilities in IoT communications. The next section contains description of network attacks that were carried, based on detected vulnerabilities to damage the functionality of smart home devices. Part of this chapter is dedicated to the description of monitoring options for ongoing attacks on smart home devices.

Kľúčové slová

Inteligentná domácnosť, Internet vecí, Bitdefender, OpenVAS, bezpečnostné testovanie

Keywords

Smart home, Internet of Things, Bitdefender, OpenVAS, penetration testing

Citácia

GROFČÍK, Peter. *Analýza bezpečnosti zařízení v chytré domácnosti.* Brno, 2018. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Petr Matoušek, Ph.D., M.A.

Analýza bezpečnosti zařízení v chytré domácnosti

Prehlásenie

Prehlasujem, že som túto bakalársku prácu vypracoval samostatne pod vedením pána Ing. Petra Matouška. Uviedol som všetky literárne pramene a publikácie, z ktorých som čerpal.

.....
Peter Grofčík
9. mája 2019

Poďakovanie

Rád by som poďakoval Ing. Petrovi Matouškovi za jeho odborné rady, čas a vynaložené úsilie počas konzultácií pri vedení mojej práce.

Obsah

Kapitola 1

Úvod

1.1 Motivácia

Technológia inteligentnej domácnosti využíva zariadenia pripojené k IoT sieti na automatizáciu a monitorovanie domácich systémov. Táto technológia bola pôvodne vyvinutá spoločnosťou IBM a bola označená ako prediktívna analýza porúch. V dnešnej dobe dochádza k vysokému záujmu spotrebiteľov o túto technológiu, a to hlavne za účelom zabezpečenia a monitorovania domácnosti. Aj keď je táto technológia na trhu od posledných rokov minulého tisícročia, stále sa často objavujú lacnejšie sady s nižšou úrovňou zabezpečenia. Na trh sa tak dostáva veľké množstvo cenovo dostupných inteligentných súprav, ktoré pre bežného používateľa dokážu spĺňať úlohy pre riadenie či monitorovanie domácnosti, avšak kvôli nízkej úrovni zabezpečenia zvyšujú riziko zneužitia monitorovaných dát. Táto skutočnosť môže pre bežného spotrebiteľa v konečnom dôsledku znamenať presný opak toho, za čo si zaplatil.

1.2 Postup práce

Hlavným cieľom práce je preskúmať princípy komunikácie zariadení inteligentnej domácnosti spolu s možnými prístupmi k zariadeniam a dostupnými možnosťami zabezpečenia. Následne previesť analýzu bezpečnosti za účelom odhalenia jednotlivých zraniteľností v nezabezpečených častiach komunikácií. Následne previesť útoky so zameraním na odhalené zraniteľnosti a popísať dostupné možnosti monitorovania prebiehajúcich útokov. Výstupom práce je dátová sada súborov vo formáte PCAP, ktorá obsahuje jednak typickú odchytenú komunikáciu zariadení inteligentnej domácnosti, ale aj prevedené útoky na odhalené zraniteľnosti.

1.3 Rozdelenie práce

V druhej kapitole sa nachádza popis zariadeniami poskytovaných možností. V kapitole sú priblížené možnosti prístupov, a zároveň rôzne možnosti využitia jednotlivých zariadení. Nasleduje popis komunikácie vedenej zariadeniami za bežnej prevádzky, čo zahŕňa kľudový stav, stavy zavádzania zariadení do prevádzky či rôzne reakcie na udalosti v rámci inteligentnej domácnosti. Nasledujúca kapitola bližšie priblížuje možné zraniteľnosti týchto zariadení. Z časti popisuje zraniteľnosti odvodené z odsledovanej komunikácie v rámci lokálneho segmentu siete, a to so zameraním na zariadeniami podporované protokoly, ktoré

predstavujú určité riziko v zabezpečení inteligentnej domácnosti. Táto analýza je doplnená o testovanie pomocou voľne dostupných programov a popis týmito programami zistených potencionálnych zraniteľností. V ďalšej kapitole sa nachádza popis útokov vedených na zariadenia inteligentnej domácnosti za účelom zneužitia odhalených zraniteľností. Kapitola tiež obsahuje popis spôsobov pre odhalenie prebiehajúcich útokov.

1.4 Ciele

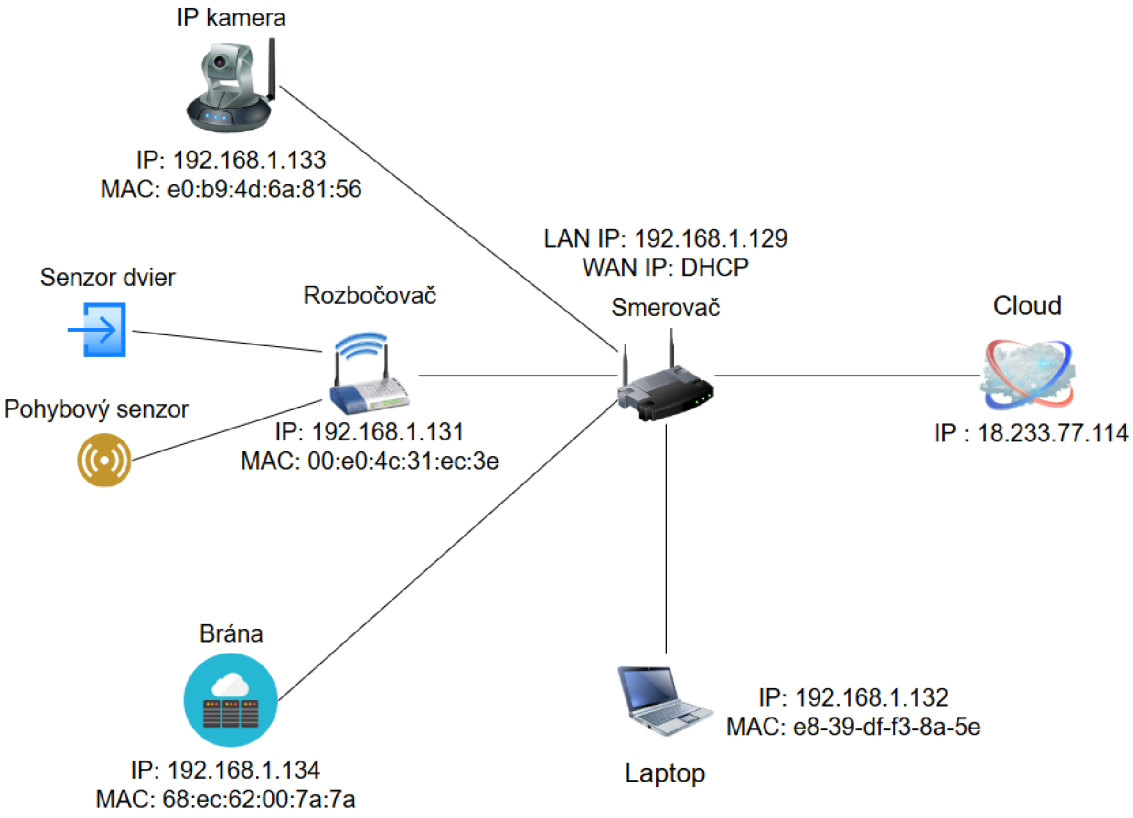
Cieľom tejto bakalárskej práce je oboznámenie sa so spôsobmi komunikácie zariadení inteligentnej domácnosti a následné odhalenie hlavných typov zraniteľností a vytvorenie postupu pre overenie zabezpečenia zariadení inteligentnej domácnosti za použitia voľne dostupných prostriedkov. Na odhalenie a overenie jednotlivých zraniteľností sú použité dáta z odchytenej komunikácie, ale aj voľne dostupné nástroje na detekciu a následne útoky vyplívajúce z odhalených zraniteľností.

Kapitola 2

Popis zariadení pre inteligentnú domácnosť

Pre analýzu bezpečnosti zariadení inteligentnej domácnosti mi poslúžila súprava MioSMART pozostávajúca z troch hlavných zariadení (rozbočovač, IP kamera a brána) a dvoch vedľajších zariadení (pohybový senzor a senzor otvárania dvier). Brána zo sady MioSMART nepredstavuje bránu (gateway) ako výstupný bod v lokálnej sieti, ale slúži len ako lokálny FTP server. Vedľajšie zariadenia (senzory) komunikujú rádiovým signálom o frekvencii 2.4GHz priamo s rozbočovačom sady MioSMART.

Pre riešenie analýzy som použil základnú topológiu lokálnej siete podľa obrázka 2.1.



Obr. 2.1: Obrázok základnej topológie použitej pri analýze a testovaní

Softvér na týchto zariadeniach dostáva aktualizácie zo vzdialeného servera, čo môže v budúcnosti viesť k ošetreniu či znefunkčneniu funkcionality poskytnutej v momente riešenia práce, a aj preto v nasledujúcej tabuľke pridávam krátky popis o zariadeniach, z ktorých pozostáva sada MioSMART.

Tabuľka 2.1: Základné informácie o zariadeniach sady MioSMART

Zariadenie	MAC adresa	Verzia softvéru	Sériové číslo
Rozbočovač	00:e0:4c:31:ec:3e	1.0.3.5016	G1069M02077
IP kamera	e0:b9:4d:6a:81:56	1.0.26.8	C1069E10173
Brána	68:ec:62:00:7a:7a	2.6.36	S1069E01287

2.1 Rozbočovač (Hub)

Úlohou zariadenia je oznamovanie udalostí zaznamenaných na s ním spárovaných senzoroch na vdialený cloud. Zariadenie je možné pripojiť do domácej siete dvomi spôsobmi. Jedným je pripojenie pomocou ethernetového kábla použitím smerovača s prístupom na internet a jeho následné pridanie do aplikácie MioSMART. Druhá možnosť je pripojenie pomocou WiFi siete generovanej smerovačom (príp. iným zariadením) zadaním potrebných údajov (SSID a heslo pre WiFi sieť).

2.1.1 Mobilná aplikácia MioSMART (časť pre rozbočovač)

Nastavenia pomocou aplikácie nie sú rozsiahle. Spočívajú v možnosti pridať (spárovať) či odobrať senzory. Ďalej tiež obsahujú možnosť prezerať si udalosti, ktoré nastali, avšak po vypnutí rozbočovača dôjde k ich premazaniu. WiFi pripojenie je možné zmeniť aj neskôr po pridaní. Rovnako tak je možné zmeniť heslo a názov zariadenia (UID je nemenné) pre prístup cez webovú aplikáciu a tiež aktualizovať softvér.

2.1.2 Webová aplikácia (IPCAM užívateľské rozhranie)

Pomocou webovej aplikácie (po pripojení na základe hesla a mena) získame prístup k rozšíreným možnostiam nastavenia zariadenia. Jedná sa však len o všeobecnú verziu webovej aplikácie, pretože obsahuje väčšie množstvo možností (napr. nastavenia pre kameru či nastavenia pre FTP [14] server atď.), ktoré však nefungujú, nič nezobrazia a dokonca webová aplikácia hlási chýbajúce súbory pre zobrazenie.

Medzi funkčné sieťové nastavenia tejto webovej aplikácie patrí nastavenie IP [13] adresy, ktoré je možné pomocou protokolu DHCP [6] alebo staticky. V prípade zvolenia statickej IP adresy, je možné staticky nastaviť aj IP adresu DNS [10] servera, avšak nie je možné tieto nastavenia separovať zvlášť ako vo väčšine počítačových systémov. Samozrejme aplikácia umožňuje zmenu nastavení WiFi pripojenia, ktorá má však význam len v prípade, že pre pripojenie zariadenia v rámci lokálnej siete použijeme WiFi a nie pripojenie pomocou ethernetového kábla. V rámci možností sieťových nastavení sa tu nachádzajú aj možnosti pre sadu UPnP [4] (univerzálna plug and play sada sieťových protokolov) a protokol PPPoE [5] (protokol point-to-point cez ethernet).

Zo systémových nastavení sú funkčné nastavenia pre správu užívateľov obsahujúce zmenu prihlasovacieho mena a hesla. Ďalej k nim patrí možnosť pre resetovanie zariadenia na predvolené nastavenia a jeho následný reštart. Vedľajšími systémovými funkciami sú možnosti

nastavenia časovej zóny, nastavenie času staticky alebo synchronizácia pomocou protokolu NTP [9]. Funkčné je tiež zobrazenie logov v rámci zariadením poskytovaných funkcií, ktoré boli vykonané pripojeným užívateľom pomocou webovej aplikácie, a teda logy o komunikácii pripojeného užívateľa alebo záznamy jadra operačného systému.

2.2 IP kamera

Pripojenie IP kamery do domácej siete je možné len pomocou WiFi siete, a to opäť zadaním potrebných údajov (SSID a heslo pre WiFi sieť) a následným pridaním zariadenia do aplikácie MioSMART. Zariadenie totiž nepozostáva z ethernetovej prípojky, kvôli čomu nie je možné pripojiť kameru pomocou kábla, ako tomu bolo pri rozbočovači.

2.2.1 Mobilná aplikácia MioSMART (časť pre IP kameru)

Najrozsiahlejšia časť aplikácie MioSMART z oblasti funkcií pre kameru je možnosť sledovania videa v reálnom čase, pričom táto časť obsahuje kontextovú ponuku zloženú z nasledujúcich možností:

- Spustenie a zastavenie nahrávania videa v reálnom čase priamo na SD kartu, pričom pre úspešne uloženie záznamu je nutné zastavenie nahrávania videa, inak môže dôjsť k strate nahrávaných dát,
- zaznamenanie jednorazovej fotografie priamo na SD kartu,
- otočenie videa v reálnom čase horizontálne a zrkadlovo,
- zmena kvality videa, pričom na výber je z dvoch možností, stredná (640 x 368) a vysoká (1280 x 720),
- zmena jasu a kontrastu videa v reálnom čase (päť rôznych úrovní jasu a kontrastu),
- možnosť posielania zvuku z mikrofónu zariadenia do kamery (tzv. komunikácia v reálnom čase s človekom pred kamerou), spojená so zapnutím a vypnutím prenášania zvuku z kamery do zariadenia s aplikáciou MioSMART,
- možnosť zobraziť a prehrať videá či fotografie uložené priamo na SD karte kamery.

Medzi nastavenia kamery patrí zmena názvu a hesla a tiež zmena WiFi siete, na ktorú je zariadenie pripojené. Kameru je tiež možné použiť ako ďalší senzor. Na základe pohybu v zornom poly kamery dôjde k udalosti, na ktorú je možné naviazať nahrávanie záznamu či jednorazovú fotografiu. Na tieto udalosti je možné nastaviť citlivosť zaznamenaného pohybu a tiež interval pre upozornenie o nasledujúcom pohybe. Prípadne je možné na kamere zapnúť nahrávanie dvadsaťštyri hodinového záznamu. Rovnako ako na rozbočovači tak aj na kamere je možnosť nastavenia časového pásma pre korektnú časovú značku vo videu. Menej podstatnými funkciami sú zobrazenie informácií o zariadení, formátovanie SD karty a odobratie kamery z MioSMART aplikácie.

2.2.2 Webová aplikácia

Webová aplikácia pre IP kameru obsahuje podstatne rozsiahlejšie možnosti nastavení štyroch základných kategórií: video a audio nastavenia, sieťové nastavenia, nastavenia činnosti po udalosti a systémové nastavenia.

2.2.3 Video a audio nastavenia

Pre video sú dostupné dva nezávisle prednastavené formáty. Formáty podporujú rovnaké rozlíšenie ako formáty v aplikácii pre mobil (stredná 640 x 368 a vysoká 1280 x 720), pričom v týchto formátoch sa nachádza rozšírená možnosť pre nastavenie bitových a rámcových hodnôt v určených rozmedziach, spolu s nastavením typu bajtovej hodnoty a rámcovej medzery nezávisle pre jeden alebo druhý formát videa.

Nastavenia, ktoré pôsobia na oba formáty zároveň, sú nastavenie normy (PAL alebo NTSC). Ďalej nastavenie OSD (na obrazovke displeja), ktoré sa týkajú nastavení informácií zobrazených na videu v reálnom čase alebo na záznamoch z kamery. Jedná sa o časovú značku, názov kamery a veľkosť fontu písma. V neposlednom rade nastavenia PTZ (snímanie sklonu priblíženia), pri ktorých sa jedná o otáčanie obrazu, nastavenie pozície, rôzne časovania a rýchlosť zaznamenania. Pre snímky z kamery je v nastaveniach jediná možnosť, a to nastavenie kvality WDR (technika pre fotografovanie a reprodukovanie dynamickej vzdialenosti snímku).

Pre audio sa jedná o nastavenia kódovacieho formátu, výšku pre nahrávaný a prehrávaný zvuk a tiež možnosť zapnúť či vypnúť zvuk pre jednotlivý formát streamu.

2.2.4 Sieťové nastavenia

Podobne ako pri rozbočovači, obsahuje aj kamera možnosť nastavenia IP adresy a adresy DNS servera staticky alebo dynamicky, ale taktiež obsahuje nastavenia portov pre protokoly HTTP [7], RTSP [15] (štandardne 80 a 554). Súvisle s tým poskytuje možnosť sieťového testu pre zistenie internetového pripojenia a nastavenie pripojenia WiFi siete. Medzi rozšírené nastavenia patria:

- Nastavenie DDNS (dynamický DNS) s možnosťou nastavenia poskytovateľa (3322.org, Dyndns.org alebo myq-see), užívateľského mena, hesla a vlastnej domény,
- nastavenie prístupu pre email na ľubovoľnom servery (ako odosielateľ) s možnosťou zabezpečeného (SSL: zabezpečená soketová vrstva) pripojenia za účelom odosielania informačných správ (udalostí z kamery) na ľubovoľný email,
- možnosť nastavenia externého servera FTP pre ukladanie záznamov z kamery (primárny účel pre prepojenie s bránou),
- možnosť zapnutia či vypnutia P2P spojenia

2.2.5 Nastavenia činností po udalosti

Činnosťami po udalosti sú reakcie zariadenia na vstupný impulz z neho samotného alebo ostatných zariadení zapojených v rámci inteligentnej domácnosti. Možnosti nastavení týchto reakcií sú nasledovné:

- Nastavenia snímania pohybu pred kamerou a následné odoslanie informačného emailu, s možnosťou zaznamenania fotografie či video záznamu o určitej dĺžke a jeho odoslanie na FTP server (brána),
- nastavenie aktivácie alarmu na určitý časový interval (dôjde k prehrávaniu zvuku alarmu z kamery,

- nastavenie zaznamenania fotografie v určitom časovom intervale a jej následné odoslanie na FTP server,
- nastavenie pravidelného nahrávania videa o určitom čase v určitej hodine či minúte,
- nastavenie masky ochrany osobných údajov pre vyznačenie regiónu video záznamu kamery, ktorý nebude zaznamenaný za účelom ochrany osobného súkromia majiteľa,
- nastavenie reakcie kamery na spustenie (udalosť) alarmu pohybového alebo dverového senzora pripojeného na rozbočovač a následné odoslanie informačného emailu s možnosťou zaznamenania fotografie či video záznamu o určitej dĺžke a jeho odoslanie na FTP server.

2.2.6 Systémové nastavenia

Do systémových nastavení opäť spadajú možnosti nastavenia času a časovej zóny, zobrazenie systémových logov udalostí a informácií o zariadení z obsahu lokálneho ukladacieho priestoru kamery. Zmenou sú nastavenia užívateľského rozhrania. Okrem nastavenia mena a hesla pre prístup jedného užívateľa (ako tomu bolo pri nastaveniach užívateľského rozhrania na rozbočovači), oplývajú tiež možnosťou pridania až siedmich ďalších užívateľov bez administrátorských práv a teda s obmedzeným prístupom (bez prístupu k nastavenaim) k zariadeniu.

2.3 Brána (Gateway)

Bránu je možné pripojiť len pomocou ethernetového kábla. Neposkytuje pripojenie pomocou WiFi a rovnako ako kamera vyžaduje externý ukladací priestor pripojený pomocou USB rozhrania pre využitie funkcie, ktorú poskytuje (FTP server).

2.3.1 Nastavenia brány

Medzi nastavenia brány ako takej pomocou aplikácie MioSMART patria nastavenie názvu, hesla a tiež názov externého ukladacieho priestoru na ňu pripojeného. Pomocou aplikácie je tiež možné k bráne priradiť kameru a prípadne nastaviť zaznamenávanie dvadsaťštyri hodinového záznamu spolu s možnosťou nastavenia miesta uloženia na externom úložnom priestore. Záznam je ukladaný formou minútových záznamov vo formáte MP4. Brána slúži vyslovene na ukladanie dát prenášaných z kamier, pričom neposkytuje rozsiahle možnosti nastavenia, a to hlavne z dôvodu, že samotné kamery obsahujú možnosť odosielania záznamu na FTP server, ktorým pre nich môže byť daná brána (predurčená funkcionalita brány). Za iným účelom by brána v inteligentnej domácnosti nemala význam.

Nastavenia brány je možné robiť len pomocou aplikácie na mobil, pretože nie je možné zobraziť žiadnu webovú aplikáciu. Pomocou počítača je možné pripojiť sa na bránu len ako na FTP server a všetky nastavenia týkajúce sa ukladania dát na bránu je nutné robiť pomocou aplikácie na mobil alebo webového rozhrania pre IP kameru, kde je bránu možné nastaviť ako FTP server pre ukladanie kamerových záznamov v intervaloch po udalosti či nepretržite.

2.4 Zhrnutie

Jednotlivé aplikácie na zariadeniach obsahujú aj nadbytočné a nefunkčné nastavenia, ktoré očividne nie sú určené priamo pre ne. Z toho vyplýva, že na ne boli nainštalované akési všeobecné aplikácie pre prácu s daným typom zariadenia, avšak táto skutočnosť nijak neovplyvňuje činnosti, ktoré by mali vykonávať. IP kamera oplýva veľkým množstvom nastavení zaznamenávaného obrazu, ktoré nijak nezasahujú do komunikácie ako takej, ale mení sa len spôsob využitia prenášaných dát v rámci webovej či mobilnej aplikácie. Brána naopak neoplýva žiadnymi špeciálnymi nastaveniami, keďže funguje len ako FTP server pre danú inteligentnú domácnosť. Na jednotlivé senzory nie je možné nadviazať akékoľvek pripojenie. Sú spojené rádiovým signálom priamo s rozbočovačom a okrem informácií, ktoré poskytujú rozbočovaču, nie je možné z nich čerpať údaje počas prevádzky.

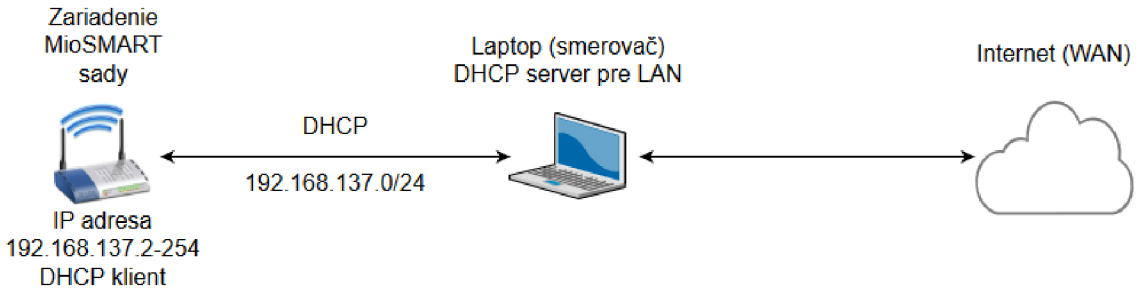
V nasledujúcej kapitole sa teda zameriam na analýzu komunikácie jednotlivých zariadení pomocou sondy (počítač a zároveň poskytovateľ pripojenia na internet), vďaka ktorej budem schopný odsledovať dáta posielané jednotlivými zariadeniami spolu s použitými sieťovými protokolmi. Takto odchytená komunikácia nám tak poskytne prvotný prehľad o úrovni zabezpečenia na jednotlivých zariadeniach spolu s mapovaním potrebných úkonov na splnenie funkcionalít zariadení.

Kapitola 3

Analýza komunikácie zariadení

Pre účely zisťovania a následného testovania zraniteľností zariadení inteligentnej domácnosti je v prvom rade nutné skúmať komunikáciu zariadení pomocou odchytu paketov za účelom zistenia typov komunikácií, ktoré tieto zariadenia vykonávajú, pre splnenie ich funkcionality. Pred nadviazaním spojenia zariadenia s cloudom dochádza k DNS rezolúcií, a teda k procesu, kedy sa na základe doménového mena zariadenie odkazuje na server s požiadavkou na IP adresu, na ktorú sa chce následne pripojiť. Vzhľadom na rezolúciu sa môže výsledná adresa servera, na ktorý sa jednotlivé zariadenie v prípade užitia odkáže, líšiť. Vo všetkých sledovaných prípadoch sa však jedná o Amazon server, keďže odkazované IP adresy odpovedajú adresnému priestoru serverov patriacich Amazonu.

Pre účely odchytu paketov som nepoužil základnú topológiu z kapitoli 2. Pre jednoduchší odchyt paketov som umiestnil sondu (počítač s bežiacim programom Wireshark[1]) priamo pred jednotlivé zariadenia sady MioSMART (obrázok 3.1). Jednotlivé zariadenia majú v odchytených súboroch dynamicky pridelené IP adresy, avšak v rámci programu Wireshark je možné si ich vyfiltrovať pomocou MAC adries zmapovaných v tabuľke 2.1. Pre jednoduchšie dohľadanie nižšie popisovaných prípadov užitia v priložených PCAP súboroch som v závere kapitoly vytvoril tabuľku 3.2 s konkrétnymi číslami paketov.

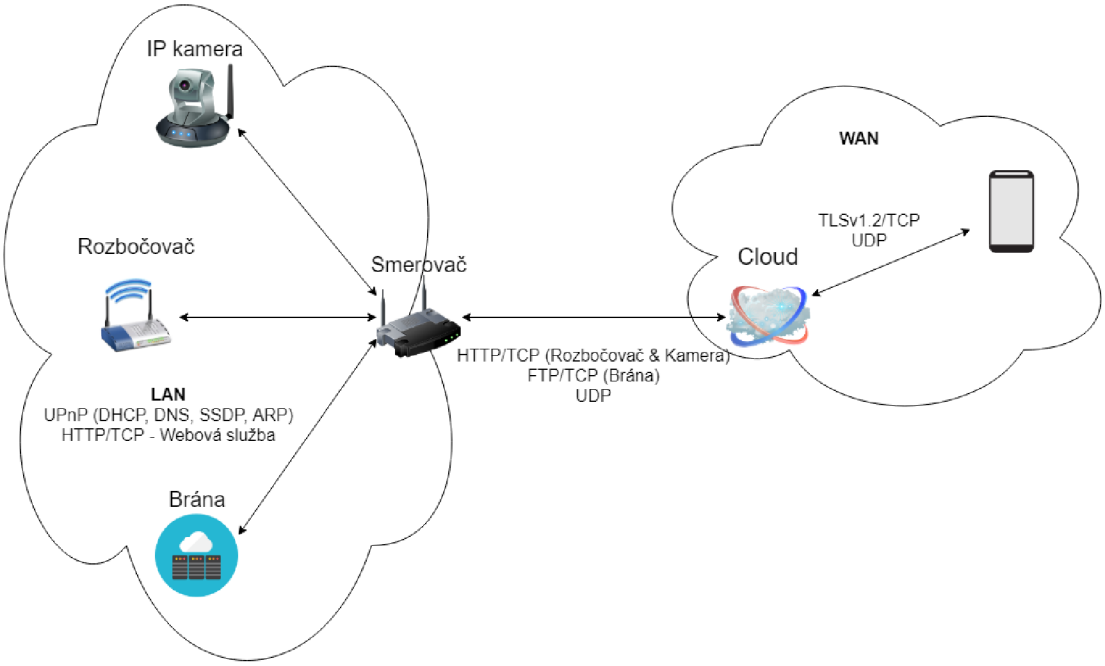


Obr. 3.1: Zapojenie pre odchyt komunikácie

Na obrázku 3.2 sú zobrazené jednotlivé komunikačné protokoly, ktoré zariadenia či samotná aplikácia MioSMART používajú pri komunikácií s cloudom. Na obrázku je možné vidieť, že šifrovaný spôsob komunikácie je využitý len pri komunikácii mobilnej aplikácie s cloudom, a že všetky dáta prenášané medzi zariadeniami sady MioSMART a cloudom sú posielané nešifrovanou formou. Senzory ako také sú spojené priamo bezdrôtovou formou s rozbočovačom sady MioSMART. Bez priameho spárovania v rámci inteligentnej domácnosti

[1] https://www.wireshark.org/

nepracujú a nie sme tak schopný sledovať komunikáciu medzi rozbočovačom a jednotlivými senzormi.



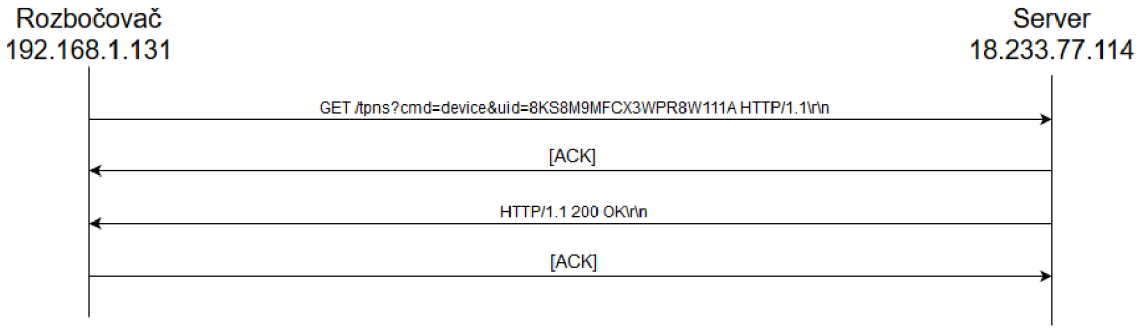
Obr. 3.2: Zariadeniami využívané protokoly

3.1 Prípady užitia rozbočovača

Zariadenie udržiava neustálu komunikáciu pomocou protokolu UDP s Amazon serverom. Podľa odchytenej komunikácie sa jedná o dáta, ktoré sa časom nemenia. Jedná sa len o udržiavanie spojenia, z ktorého zrejme aplikácia MioSMART čerpá informácie ohľadom dostupnosti zariadenia. K zmene týchto dát dochádza len v prípade udalosti zaznamenanej na rozbočovači.

3.1.1 Nadviazanie spojenia so serverom

Nadviazanie spojenia s Amazon serverom nastáva formou trojfázovej synchronizácie TCP. Nasleduje komunikácia pomocou protokolu HTTP formou prostého textu, a to pomocou príkazu GET spolu s UID zariadenia následované ukončením TCP spojenia (obrázok 3.3). Počas tejto komunikácie môže zariadenie obdržať zápornú odpoveď zo strany servera. Pravdepodobne sa jedná o situáciu, kedy je daný server zaneprázdnený inými požiadavkami. V takejto situácií sa pomocou DNS rezolúcie zariadenie dotáže ďalšiemu možnému serveru a opakuje pokus o spojenie.



Obr. 3.3: Nadviazanie komunikácie rozbočovača s cloudom

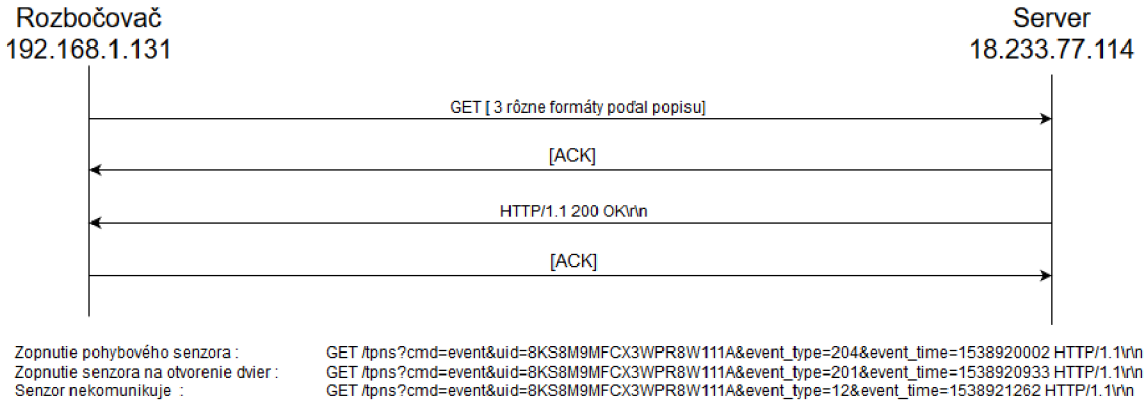
3.1.2 Komunikácia viazaná na spárované senzory

Správy týkajúce sa senzorov sú v rovnakom tvare ako nadviazanie spojenia. Jediná zmena je v prostom texte po príkaze GET. Príkaz nie je zakončený identifikačným číslom zariadenia, ale pokračuje označením typu udalosti a jej časom. Na obrázku 3.4 je možné vidieť použité typy komunikácie podľa údajov z nasledujúcej tabuľky 3.1.

Tabuľka 3.1: Typy udalostí komunikácie na základe senzorov

Popis udalosti	Typ udalosti (číslo)
Zopnutie pohybového senzora	204
Zopnutie senzora na otvorenie dvier	201
Senzor nekomunikuje (hlásenie chyby)	12

Z odsledovanej komunikácie usudzujem, že hlásenie chyby senzora je všeobecný typ udalosti bez možnosti rozoznania, o ktorý senzor sa jedná. Dochádza však k zmene dát posielaných pomocou protokolu UDP, z ktorých cloud následne zistí, o ktorý konkrétny senzor sa jedná.

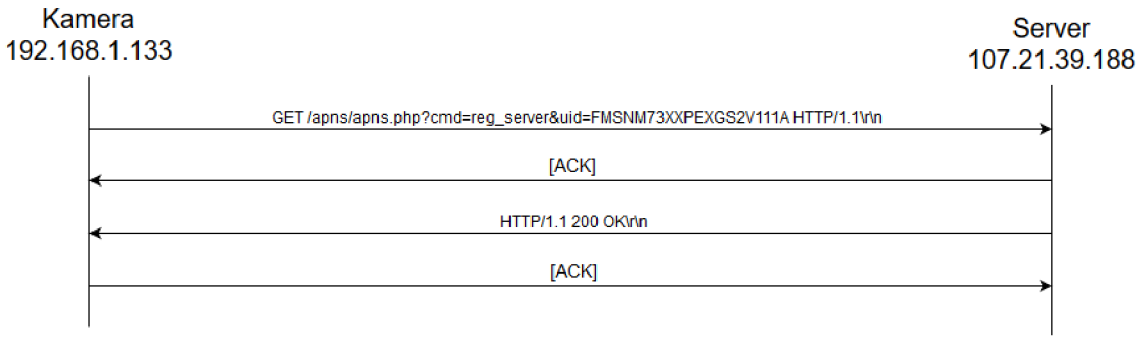


Obr. 3.4: Komunikácia na podnet senzorov pripojených k rozbočovaču

3.2 Prípady užitia kamery

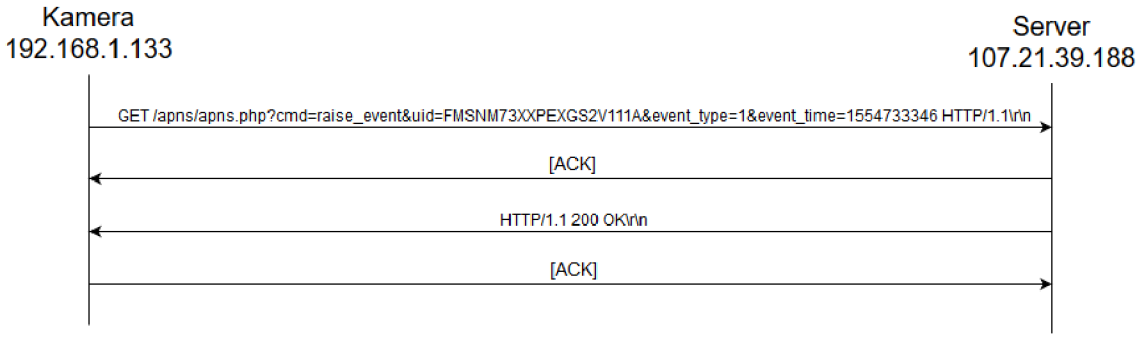
Kamera rovnako ako rozbočovač nadväzuje spojenie s Amazon serverom HTTP spojením, avšak celkový formát príkazu má mierne pozmenený tvar (obrázok 3.5). Rovnako však

posiela svoje UID nezašifrovanou formou v rámci správy. Hlavný rozdiel oproti rozbočovaču je vo forme príkazu odosielaného za spúšťacím príkazom GET. Zatiaľ čo pri rozbočovači sa jednalo o príkaz *device* tak pri naväzovaní spojenia kamery je to príkaz *reg_server*. Podľa cesty, na ktorú sa zariadenie na could odkazuje, zrejme dochádza k spusteniu skriptu s funkciami na servery za účelom práce s videom prenášaným z kamery, čo je však ďalej riešené UDP protokolom po nadviazaní spojenia.



Obr. 3.5: Nadviazanie komunikácie kamery s cloudom

Druhým prípadom užitia je notifikácia o zaznamenanej udalosti na kamere (obrázok 3.6). Podobne ako pri senzoroch na rozbočovači dôjde k odoslaniu správy obsahujúcej typ udalosti. V tomto prípade sa však jedná len o jeden typ, a to zaznamenanie pohybu v zornom poly s číslom jedna. Na rozdiel od rozbočovača, kde sa môže jednať o veľké množstvo s ním spárovaných typov senzorov, je na kamere možné nastaviť len jeden typ udalosti, a teda generuje pri spojení len jeden typ udalosti.



Obr. 3.6: Zaznamenanie pohybu v zornom poly kamery

3.3 Všeobecné závery z analýzy užitia zariadení

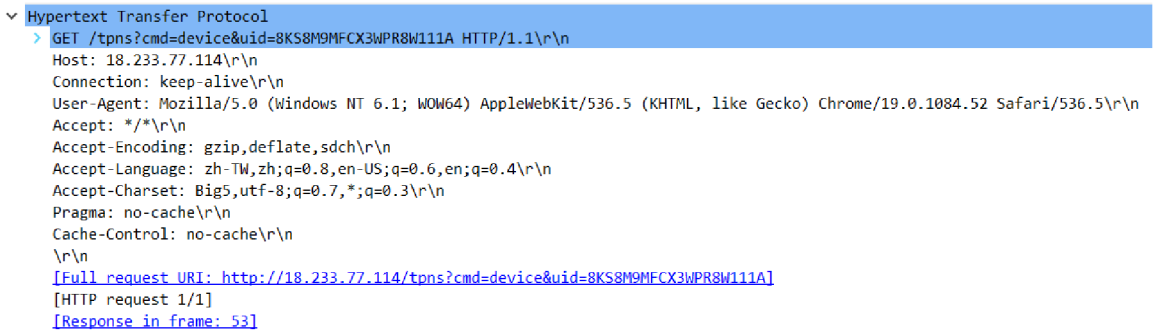
Brána sady MioSMART sama o sebe nenadväzuje žiadne špeciálne spojenie. Po zapojení sa odkáže na geografické informácie o mieste, kde sa nachádza ale, žiadnym spôsobom neoznamuje cloudu svoju dostupnosť, čo je zrejme aj jedným z dôvodov, prečo ako jediná neposkytuje bezdrôtové pripojenie a je potrebné ju s aplikáciou naviazať na priamo, a tiež prečo iba pri nej nie je v aplikácii MioSMART zobrazená informácia o dostupnosti, ale iba možnosť pokusu o pripojenie na ňu.

Pri spojení zariadení so serverom nedochádza k žiadnemu šifrovaniu. Jediné zabezpečenie nešifrovanej HTTP komunikácie je vo forme UID zariadenia. Toto identifikačné číslo

zariadenia je nemenné. Za predpokladu, že útočník už toto UID pozná, uľahčuje mu to akýkoľvek pokus o nepozorovaný (z hľadiska servera) útok na zariadenie. Samozrejme je tiež možná generácia UID náhodne alebo sporadicky. Vzhľadom na to, že komunikácia medzi aplikáciou MioSMART a serverom prebieha pomocou protokolu TLSv1.2 s SHA-256 kryptovaním tak si myslím, že má väčší význam zamerať sa na zabezpečenie strany komunikácie zariadenia so serverom a zariadenia s užívateľom na priamo pomocou webovej aplikácie na jednotlivých zariadeniach. Jednotlivé prípady užitia komunikácie zariadenia so serverom (cloudom) je možné dohľadať v odchytených dátach, ktoré sú zhrnuté v tabuľke 3.2.

Tabuľka 3.2: Odchytená komunikácia

Zariadenie	Názov súboru	Číslo paketu	Poznámka
Rozbočovač	pohybovy_senzor.pcap	49	Nadviazanie komunikácie hubu s Amazon serverom pomocou TCP komunikácie.
		160	
		247	
		2440	Zopnutie pohybového senzora.
		2779	
		4837	Pohybový senzor nekomunikuje. Hlásenie chyby pohybového senzora.
	dverovy_senzor.pcap	560	Zopnutie dverového senzora
		1936	Dverový senzor nekomunikuje. Hlásenie chyby dverového senzora.
Kamera	kamera.pcap	466	Nadviazanie komunikácie kamery s Amazon serverom.
		963	Zaznamenanie pohybu na kamere.
Brána	brana.pcap	N/A	N/A



Obr. 3.7: Príklad odchyteného HTTP paketu zobrazeného vo Wiresharku

Kapitola 4

Analýza a testovanie bezpečnosti

Táto kapitola slúži na odhalenie možných zraniteľností zariadení inteligentnej domácnosti. Na analýzu som opäť využil topológiu z kapitoly 2. Analýzu popísanú v tejto kapitole je rozdelená do dvoch nasledovných kategórií.

1. Manuálna analýza na základe odchytených PCAP súborov z kapitoly 3.
2. Analýza pomocou voľne dostupných programov Bitdefender, OpenVas a Nmap.

4.1 Analýza na základe odchytenej komunikácie zariadení

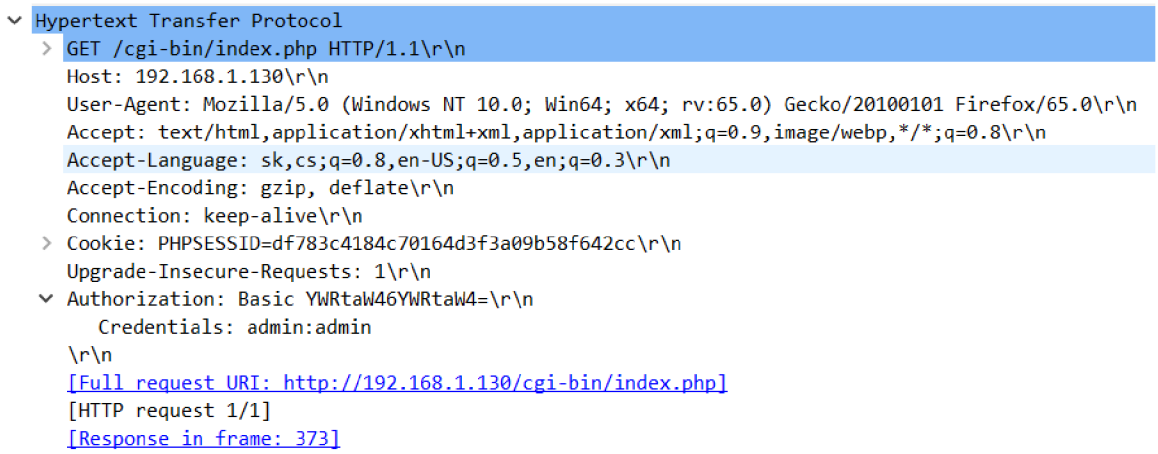
4.1.1 Komunikácia ARP

Z odchytenej komunikácie je možné zistiť, že každé zo zariadení sady MioSMART podporuje ARP [12] protokol, ktorý tvorí základnú súčasť pre zisťovanie zariadení v sieti, pričom jeho úlohou je mapovanie fyzickej (MAC) adresy na IP adresu priradenú zariadeniu. Zariadenia zo súpravy MioSMART tak odpovedajú na ARP dotazy, ktoré tečú po lokálnom segmente siete. Súčasťou ARP odpovede je MAC adresa zariadenia, ktorú je tak pri bežnej prevádzke smerovač či iné zariadenie schopné zmapovať na IP adresu za účelom následného smerovania paketov v rámci lokálnej siete. Zariadenia na jednotlivé dotazy odpovedajú bez akýchkoľvek výhrad, čo umožňuje ich zneužitie zo strany útočníka. Útočník je tak schopný vysielať vlastné ARP dotazy na jednotlivé IP adresy, na ktoré mu zariadenia v sieti odpovedia, vďaka čomu je tak schopný zmapovať bežiace zariadenia na lokálnom segmente siete.

Opačné využitie pre útok vychádza z odpovedí na ARP dotazy, kedy útočník vysiela podvrhnuté odpovede za účelom presvedčenia dopytujúceho sa zariadenia, že práve jeho MAC adresa sa nachádza na dopytovanej IP adrese. Tento útok sa nazýva *ARP spoofing* a slúži na presmerovanie komunikácie určenej obeti na zariadenie útočníka. V prípade nešifrovanej komunikácie je tak schopný zistiť určité citlivé informácie preposielané medzi zariadeniami. Následne môže tieto správy upraviť a preposlať ďalej, alebo dokonca odstaviť komunikáciu smerujúcu na dané zariadenie v sieti. Najčastejšie sa však tento útok používa ako otvárací pred ďalšími útokmi (odmietnutie služby, MITM či zneužitie relácie) a je možné ho uskutočniť len v sieťach, na ktorých pracuje ARP protokol a za predpokladu, že má útočník prístup do lokálneho segmentu siete. Prístup do lokálnej siete môže dôjsť útokom na WiFi komunikáciu, ktorá je nevyhnutná pri väčšinu zariadení inteligentnej domácnosti.

4.1.2 Komunikácia HTTP

Ďalšou možnou zraniteľnosťou vyplývajúcou z analýzy odchytenej komunikácie je využitie nešifrovanej formy protokolu HTTP. Zariadenia totiž posielajú citlivé informácie pomocou HTTP protokolu vo forme prostého textu. Pri komunikácii zariadenia s cloudom sa jedná o identifikačné číslo, ktorým sa zariadenie identifikuje voči serveru, s ktorým chce následne komunikovať pre umožnenie funkcionality v rámci poskytovanej služby. V prípade komunikácie lokálneho užívateľa, ktorý využíva webovú formu prístupu na zariadenie, sa jedná o užívateľské meno a heslo (obrázok 4.1) pre umožnenie prístupu k nastaveniam zariadenia vo forme webovej aplikácie.



Obr. 4.1: Príklad odchyteného HTTP paketu s prihlasovacími údajmi zobrazeného vo Wiresharku

V prípade, že sa útočníkovi podarilo odchytiť danú komunikáciu, je tak schopný vyčítať citlivé informácie nachádzajúce sa v daných správach. Tieto informácie môže následne použiť pri ďalších útokoch, či na ich základe získať plný prístup k zariadeniu, čo by takmer určite znamenalo fatálne následky na funkcionalitu zariadenia.

4.1.3 Komunikácia UPnP

V neposlednom rade je možnou zraniteľnosťou podpora sady protokolov UPnP, ktorá je implementovaná na všetkých zariadeniach MioSMART súpravy. Jedná sa o univerzálnu sadu protokolov, ktorých účelom je zjednodušiť zavádzanie nových zariadení v rámci lokálneho segmentu siete. Táto sada je založená na komunikačných štandardoch pre zdieľanie dát a informácií o zariadeniach. Zohráva tak rolu v rámci analýzy bezpečnosti za účelom zisťovania informácií o jednotlivých zariadeniach.

Každé zariadenie, ktoré podporujúce sadu protokolov UPnP, disponuje klientom DHCP, vďaka ktorému po pripojení do siete vyhľadá dostupný server DHCP pre pridelenie IP adresy v rámci lokálneho segmentu siete. Po úspešnom obdržaní IP adresy, umožňuje UPnP následne oznámiť služby poskytované zariadením kontrolnému bodu v sieti. Hlavným prvkom pri takejto komunikácii je správa zisťovanie (discovery 4.2), ktorá obsahuje niekoľko základných špecifikácií o zariadení (podporované služby, typ či identifikátor).

```
Simple Service Discovery Protocol
  NOTIFY * HTTP/1.1\r\n
  HOST: 239.255.255.250:1900\r\n
  CACHE-CONTROL: max-age=120\r\n
  lOCATION: http://192.168.137.84:59986/rootDesc.xml\r\n
  SERVER: Ubuntu/lucid UPnP/1.1 MiniUPnPd/1.6\r\n
  NT: upnp:rootdevice\r\n
  USN: uuid:2c24dccc-8bd1-4fe0-b520-146e44baa8f2::upnp:rootdevice\r\n
  NTS: ssdp:alive\r\n
  OPT: "http://schemas.upnp.org/upnp/1/0/";\r\n
  01-NLS: 1\r\n
  BOOTID.UPNP.ORG: 1\r\n
  CONFIGID.UPNP.ORG: 1337\r\n
  \r\n
  [Full request URI: http://239.255.255.250:1900*]
```

Obr. 4.2: Príklad odchyteného SSDP paketu zobrazeného vo Wiresharku

Po získaní základných špecifikácií zo zisťovacej správy dochádza k využitiu poskytnutej URL informácie, na ktorej je dotazujúci schopný získať bližší popis zariadenia vo forme XML. Tento popis zahrňuje napríklad meno či označenie zariadenia, jeho sériové číslo a názov výrobcu (obrázok 4.3). Nasledujúcimi ktorkmi sú ovládanie, upozornenie a prezentácia. Tieto kroky vychádzajú z URL poskytnutej v predchádzajúcej správe. Po obdržaní tejto informácie je kontrolný bod schopný komunikovať so službami, ktoré dané zariadenie podporuje. V súprave MioSMART sa jedná o webové rozhrania rozbočovača a IP kamery, ktoré som bližšie popísal pri popise zariadení 2.

```
−<root>
 −<specVersion>
    <major>1</major>
    <minor>0</minor>
  </specVersion>
 −<device>
  −<deviceType>
      urn:schemas-upnp-org:device:DigitalSecurityCamera:1
    </deviceType>
    <friendlyName>MioSMART-3WPR8W111A </friendlyName>
    <manufacturer>Mio</manufacturer>
    <manufacturerURL>http://www.smartbridge-tech.com/</manufacturerURL>
    <modelDescription>Network Camera</modelDescription>
    <modelName>SBT-IPC-01</modelName>
    <modelNumber>IPC-01</modelNumber>
    <modelURL>http://www.smartbridge-tech.com/IP_CAM.html</modelURL>
    <serialNumber>0000001</serialNumber>
    <UDN>uuid:2f01369c-6c79-4ddc-b487-ecf34d0b782e</UDN>
   −<serviceList>
    −<service>
      <serviceType>urn:schemas-upnp-org:service:Dummy:1</serviceType>
      <serviceId>urn:upnp-org:serviceId:dummy1</serviceId>
      <controlURL>/dummy</controlURL>
      <eventSubURL>/dummy</eventSubURL>
      <SCPDURL>/dummy.xml</SCPDURL>
     </service>
    </serviceList>
    <presentationURL>http://192.168.1.130/</presentationURL>
    <ipcamModelName>smartbridge-tech-ipcam</ipcamModelName>
  </device>
</root>
```

Obr. 4.3: Popis rozbočovača vo formáte XML

Táto sada však nevyužíva žiadny spôsob autentifikácie. Existuje aj neštandardné rozšírenie UPnP-UP, ktoré poskytuje autorizačné mechanizmy pre zariadenia podporujúce sadu

UPnP, avšak toto rozšírenie nie je implementované na zariadeniach sady MioSMART. Sada UPnP sa vo všeobecnosti využíva na zariadeniach určených pre lokálnu sieť, kde výrobcovia predpokladajú, že lokálne systémy a lokálny používatelia sú dôveryhodný, čo však nie vždy musí byť pravda.

4.2 Testovanie bezpečnosti pomocou programu Bitdefender

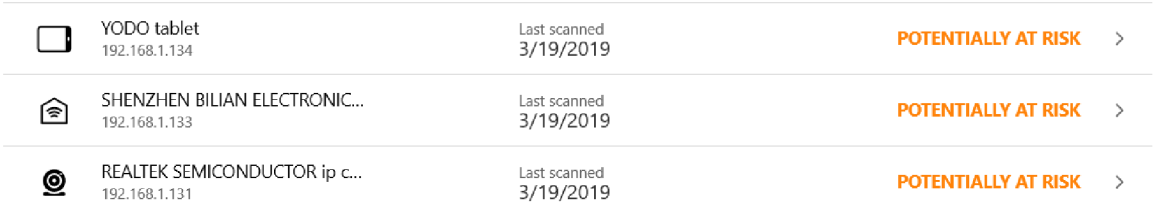
Bitdefender[1] je nástroj určený na zabezpečenie domácich IoT zariadení vytvorený pre platformu Windows. Jeho platená verzia poskytuje určitú formu ochrany voči útokom z vonka (internetu) ako aj anti-vírusovú ochranu v reálnom čase. Vo voľne dostupnej verzií poskytuje plne automatizovaný skener zraniteľností pre zariadenia pripojené v lokálnej sieti. Po nájdení zraniteľností poskytuje možnosť zabezpečenia vytvorením akéhosi druhu firewallu, ktorý má zabrániť útokom z internetu. Toto je však možné len v platenej verzií. Pre účely analýzy bezpečnosti sady MioSMART mi poslúžil ako skener možných zraniteľností, avšak neposkytuje žiadne bližšie informácie o tom, ako na dané zraniteľnosti prišiel. Bitdefender bol tiež schopný odhaliť niekoľko zraniteľností, na ktoré som narazil už v prvotnej analýze na základe odchytenej komunikácie zariadení. A rovnako tak takmer všetky zraniteľnosti, ktoré som neskôr odhalil a bližšie popíšem pri programe OpenVas. Jedná sa o:

- Prenos citlivých informácii cez nezašifrovaný HTTP protokol,
- krížové skriptovanie (cross-site skripting),
- spúšťanie ľubovoľného kódu cez protokol HTTP,
- chyba zabezpečenia prístupu na protokole HTTP,
- poškodenie (korupcia) pamäte,
- odmietnutie služby.

Tieto zraniteľnosti od pohľadu naznačujú iné vyhodnotenie ako zraniteľnosti neskôr nájdené programom OpenVas, avšak v konečnom dôsledku popisujú rovnaké chyby zariadení len menej presnejším pomenovaním a bez konkretizovaného spôsobu ich detekcie. Krížové skriptovanie spolu so spúšťaním podvrhnutého kódu a chyby zabezpečenia prístupu cez HTTP protokol sú totiž zraniteľnosti detekované na základe chýbajúcich bezpečnostných hlavičiek v paketoch pri komunikácií cez HTTP protokol, pričom OpenVas tieto skutočnosti popísal ako jednu zraniteľnosť spolu s oveľa presnejším popisom jej detekcie. Skutočnosť, že Bitdefender mi nebol schopný poskytnúť informácie o spôsobe detekcie jednotlivých zraniteľností, čo bolo hlavným dôvodom prečo som siahol aj po ďalšom nástroji.

Druhým dôvodom boli však aj nepresné informácie o zariadeniach ako takých. Už pri popise zariadení 2 som spomínal, že webové aplikácie bežiace na jednotlivých zariadeniach sady MioSMART sú očividne určené pre použitie na viacerých zariadeniach rovnakého typu. Podľa informácií, ktoré mi poskytol program Bitdefender, sa však podľa neho jedná o úplne iné typy zariadení, než akými sú v realite.

[1] https://www.bitdefender.com/solutions/home-scanner.html



Obr. 4.4: Informácie o detekovaných zariadeniach programom Bitdefender

Podľa informácií z programu Bitdefender je z obrázka 4.4 možno vidieť, že nebol schopný správne detekovať typy zariadení sady MioSMART. Zariadenia boli počas detekcie zapojené podľa obrázku 2.1 z prvej kapitoly popisu zariadení 2, a teda je možno vidieť, že rozbočovač tejto sady bol programom klasifikovaný ako IP kamera a brána ako YODO tablet. Jedine IP kameru bol program schopný zdetekovať ako zariadenie značky *Shenzen bilian electronic* so všeobecným popisom zariadení inteligentnej domácej automatizácie.

4.3 Testovanie bezpečnosti pomocou programu OpenVas

Otvorený systém hodnotenia zraniteľnosti (OpenVas[2]) je softvér obsahujúci niekoľko nástrojov, ktoré poskytujú skenovanie a analýzu bezpečnosti jednotlivých zariadení pripojených v lokálnej sieti. Všetky OpenVas produkty patria pod licenciu GPL (General Public License). Pre tento nástroj som sa rozhodol po neúspešných pokusoch s nástrojmi, ktorých výstupy môžu byť tiež relevantné, avšak užívateľovi bližšie neprezradia ako prišli k záveru, že dané zariadenie je náchylné na daný typ útoku. OpenVas je softvér určený na prácu na operačnom systéme Linux a neponúka podporu iných operačných systémov. Pred jeho inštaláciou, je nutné sa ubezpečiť, že sú v operačnom systéme nainštalované knižnice pre podporu tvorby a práce so soketmi.

OpenVas pracuje na podobnom princípe ako Bitdefender. Pokúša sa o spojenie s cieľom (zariadením) na dostupné porty a snaží sa tak nadviazať spojenie pomocou základných protokolov využívaných pri sieťovej komunikácií. Počas týchto pokusov dochádza tiež k útokom hrubou silou za účelom eventuálneho prelomenia zabezpečenia odhadnutím prístupových údajov [11]. Na tieto útoky využíva vlastnú databázu ale aj databázu CVE záznamov, čo je veľké pozitívum oproti Bitdefenderu. Na základe CVE[3] databáze je tak schopný testovať väčšie množstvo verejne známych zraniteľností domácich zariadení, vďaka čomu aj poskytne lepšiu a presnejšiu spätnú väzbu spolu s popisom jednotlivých zraniteľností.

Jednou z ďalších možností programu je generácia výsledného dokumentu s kompletným popisom jednotlivých odhalených zraniteľností na testovaných zariadeniach a jeho export vo viacerých formátoch. Dokumenty s informáciami týkajúcimi sa zariadení sady MioSMART prikladám vo formáte PDF spolu s odchytenými PCAP súbormi. Zraniteľnosti odhalené programom OpenVas boli nasledovné:

- Prenos citlivých informácií cez nezašifrovaný HTTP protokol,
- chýbajúce HTTP hlavičky zabezpečenia,
- Lighttpd viacnásobné zraniteľnosti,

[2] https://www.kali.org/penetration-testing/openvas-vulnerability-scanning/
[3] https://cve.mitre.org/

- TCP časová pečiatka (Timestamp),
- Odmietnutie služby (DoS),
- povolený prístup pomocou protokolu Telnet.

Tieto zraniteľnosti pokrývajú všetky nájdene zraniteľnosti programom Bitdefender a tiež sa tu nachádzajú také, ktoré program Bitdefender neodhalil. Keďže OpenVas poskytuje bližšie informácie k nájdeným zraniteľnostiam, som tak schopný ich teraz bližšie popísať a neskôr sa pokúsiť o z nich vyplývajúce typy útokov.

4.3.1 Prenos citlivých informácií cez nezašifrovaný HTTP protokol

Táto zraniteľnosť sa týka všetkých zariadení MioSmart sady, čo som odhalil už v kapitole 3. Z analýzy komunikácie je jasné, že daný nešifrovaný prenos predstavuje zraniteľnosť pri komunikácií zariadenia so serverom. OpenVas však odhalil, že danú zraniteľnosť obsahujú zariadenia aj na strane komunikácie klienta so zariadením. V prípade pripojenia klienta (počítača) na zariadenie sú pri tomto spojení tiež prenášané citlivé údaje (meno a heslo) cez nezašifrované spojenie HTTP protokolom. Útočník by mohol túto zraniteľnosť využiť na skompromitovanie komunikácie medzi klientom a zariadením pomocou napríklad ARP spoofingu, čím by získal prístup k citlivým údajom, či pomocou následného útoku man-in-the-middle (MITM) tieto údaje aj pozmenil a znemožnil tak pripojenie.

Na zistenie tejto zraniteľnosti program použil metódu zozbierania informácií z pokusov o pripojenie na dané zariadenie. Pomocou tejto metódy zistil, že zariadenie nepoužíva šifrovanú verziu protokolu HTTP. Dochádza tak k odosielaniu citlivých údajov na server cez jeho nešifrovanú formu. Rovnako tak zistil, že webová aplikácia na strane zariadenia nevykonáva prenos citlivých informácii pomocou zabezpečeného SSL/TLS pripojenia, a teda po pripojení lokálneho klienta na rozhranie webovej aplikácie rovnako dochádza k nešifrovanému prenosu prihlasovacích údajov vo forme prostého textu.

4.3.2 HTTP hlavičky zabezpečenia

Program pri analýze komunikácie kontroluje všetky známe hlavičky zabezpečenia [2]. Na základe tejto analýzy bol schopný zistiť nasledovné chýbajúce hlavičky zabezpečenia v protokole HTTP [8]:

- Content-Security-Policy
- Referrer-Policy
- X-Frame-Options
- X-Permitted-Cross-Domain-Policies
- X-XSS-Protection

Tieto hlavičky môžu aplikácie na zariadeniach použiť pre zvýšenie zabezpečenia poskytovanej aplikácie. V prípade ich použitia môžu zabrániť internetovým prehliadačom, ktoré sa spájajú s aplikáciou, podľahnúť zraniteľnostiam, ktorým je možné ľahko predchádzať. Absencia týchto hlavičiek tiež indikuje, že dané zabezpečenie nemusí byť implementované ani priamo na strane webovej aplikácie zariadenia.

Hlavička *Content-Security-Policy* pomáha predchádzať krížovému skriptovaniu spolu s ďalšími útokmi typu kódových injekcií. Ich použitím definujeme zdroje obsahu webovej aplikácie, ktoré sú schválené a je možné ich v prehliadači načítať. Týmto ošetrením je tak možné zabrániť načítaniu útočníkom podvrhnutej webovej aplikácie, ktorá sa môže pokúšať vymámiť citlivé informácie od lokálneho používateľa.

Hlavička *X-XSS-Protection* je rozšírením ochrany proti rovnakým typom útokov. Jej použitím dôjde k povoleniu v prehliadači vstavaného filtra krížového skriptovania, na základe ktorého je prehliadač schopný, v prípade detekcie odrazeného útoku, prerušiť načítanie podvrhnutej webovej stránky.

Hlavička *X-Frame-Options* pri vykresľovaní stránky indikuje, či daný objekt (frame) je dôveryhodný. Jedná sa tak o ochranu voči útoku typu *clickjacking*, čo je spôsob podvedenia používateľa. Vykrášlením objektu, ktorý na stránku nepatrí, a jeho následným kliknutím či vyplnením užívateľom, je útočník schopný vylákať od užívateľa citlivé informácie, či prípadne prevziať kontrolu nad zariadením nič netušiaceho používateľa.

Hlavička *Referrer-Policy* určuje, ktoré informácie o sprostredkovateľovi posielané v hlavičke *Referrer*, by mali byť zahrnuté so žiadosťami. Táto hlavička slúži na ochranu proti nežiadúcemu presmerovaniu na škodlivé stránky. V prípade správnej implementácie dôjde pri pokuse o presmerovanie na nie správne označenú časť webovej aplikácie varovanie na strane internetového prehliadača o opustení dôveryhodnej stránky.

Posledná chýbajú hlavička *X-Permitted-Cross-Domain-Policies* bola vytvorená za účelom ochrany voči súborom typu Flash a PDF. Pokiaľ webový klient požiada o súbor na vlastnej doméne, ale tento obsah je smerovaný na inú doménu, môže tak dôjsť k načítaniu škodlivého obsahu, ktorý môže z užívateľa vyvábiť citlivé informácie. Pri použití tejto hlavičky však dochádza k určitému typu autentifikácie medzi vlastnou a odkazovanou doménou, čo pri správnej implementácií zabráni načítaniu škodlivého obsahu.

Za zmienku stojí ešte hlavička *strict-transport-security*. Táto hlavička povoľuje načítanie webovej aplikácie výhradne cez protokol HTTPS, čím nepovoľuje využitie nešifrovanej verzie HTTP. Jej použitím by tak došlo k zníženiu bezpečnostných rizík v rámci komunikácie medzi zariadením inteligentnej domácnosti a lokálnym užívateľom. Pre zariadenia sady MioSMART však táto hlavička nie je použiteľná, pretože ako také neobsahujú implementáciu protokolu HTTPS, a preto nebola jej absencia programom odhalená.

4.3.3 Odmietnutie služby (DoS)

Detekcia zraniteľnosti na DoS útok je prevediteľná odosielaním veľkého počtu neoprávnených požiadaviek na zariadenie. Toto sa však už považuje za penetračné testovanie, čo tento program sprostredkúva, avšak bez konkrétneho pluginu pre útok DoS dochádza len k vyhodnoteniu toku testovania ostatných zraniteľností a útokom hrubou silou na predvolené prihlasovacie údaje. Z analýzy tohto toku, ktorý vôbec nie je taký intenzívny, akým je útok DoS, program zistil, že dochádza k rôznym oneskoreniam v negatívnych odpovediach zo zariadenia či miestami až k ich absencii. Túto skutočnosť je možno vyčítať z odchytenej komunikácie počas behu programu. Z týchto dôvodov vyhodnotil, že zariadenia môžu byť náchylné na DoS/DDoS útok.

4.3.4 Viacnásobné zraniteľnosti Lighttpd

Táto zraniteľnosť bola programom zistená na bráne. Jedná sa o softvér založený na otvorenom kóde (open-source), ktorý je optimalizovaný pre prostredia s kritickou rýchlosťou a pre zabezpečenie veľkého množstva pripojení v reálnom čase. Tento softvér je však náchylný

na viacero zraniteľnosti. Zneužitie tejto zraniteľnosti umožní útočníkovi spúšťať ľubovoľné SQL príkazy a tým následne umožní čítanie ľubovoľných súborov prostredníctvom názvu hostiteľa.

4.3.5 TCP časová pečiatka

Touto zraniteľnosťou opäť oplývajú všetky zariadenia súpravy MioSMART, pričom však patrí k tým menej závažným. Zariadenia používajú pri komunikácií TCP časové pečiatky, z čoho je možné vypočítať časovú dobu, počas ktorej zariadenie poskytuje svoje služby. Jedná sa o rozšírenie komunikácie TCP pre vysoký výkon. Pri ich využití je útočník schopný vypočítať čas, počas ktorého je zariadenie dostupné, či v extrémnych prípadoch odsledovať čas do najbližšieho plánovaného reštartu zariadenia. Na zistenie tejto zraniteľnosti OpenVas odosiela špeciálne IP pakety s malým oneskorením medzi klientom a zariadením. Odpovede sú prehľadávané a pri zistení časovej pečiatky následne referované.

4.3.6 Telnet

Detekcia bežiaceho Telnet servera vyšla pozitívne pre kameru a bránu. Na Telnet pripojení ako takom nedochádza k analýze zabezpečenia, pretože Telnet ako taký predstavuje určitú úroveň zraniteľnosti. Viacerí experti na systémové zabezpečenia (napr. SANS Institute [16]) odporúčajú, že použitie Telnetu na vzdialené prihlasovanie by malo byť štandardne vypnuté. Telnet totiž štandardne nešifruje dáta posielané cez toto pripojenie a to vrátane hesiel. Preto je často praktické odpočúvať dané pripojenie a zistené heslá neskôr použiť na škodlivé účely.

Každý kto má prístup k smerovaču, prepínačom alebo bráne umiestnenej v sieti medzi dvoma hostiteľmi, môže zachytiť prechádzajúce pakety Telnet a získať tak informácie na prihlásenie či čokoľvek iné čo sa v dátach nachádza a to už aj za použitia bežných utilít (Wireshark či tcpdump). Väčšina z implementácií Telnetu nepoužíva žiadne spôsoby, ktoré by zabezpečili, že komunikácia prebieha len medzi dvomi priamo pripojenými zariadeniami, a teda nie je narušená niekde medzi nimi.

4.4 Analýza aktívnych služieb pomocou Nmap

V rámci analýzy zraniteľností zariadení v inteligentnej domácnosti som použil nástroj Nmap[4]. Jedná sa o bezplatný otvorený sieťový skener, používaný na zisťovanie zariadení a služieb na nich bežiacich. Program OpenVas a Bitdefender pri analýze určite zisťovali pred testovaním zraniteľností jednotlivé otvorené porty na zariadeniach, avšak neposkytli žiadnu komplexnú informáciu o službách, ktoré zariadenia celkovo poskytujú. Z toho dôvodu sa ukázalo užitočným vykonať analýzu aj pomocou programu Nmap. Výstupom samotného programu sú základné zistené údaje o skenovaných zariadeniach spolu s otvorenými portami a na nich poskytovanými službami. Detekované otvorené porty a služby poskytované zariadeniami sady MioSMART sú rozpísané v tabuľke 4.2.

[4] https://nmap.org/

Tabuľka 4.1: Otvorené porty

Zariadenie	Port	Služba
Rozbočovač	80	HTTP
	52655	Neznáme
Kamera	23	Telnet
	80	HTTP
	554	RTSP
	1018	Neznáme
	1235	Mosaicsyssvc1
	8840	Neznáme
	34780	Neznáme
Brána	21	FTP
	23	Telnet
	80	HTTP
	139	netbios-ssn
	445	microsoft-ds
	8080	http-proxy
	34759	Neznáme

Všetky zmapované aktívne služby pracujú na protokole TCP. Väčšina aktívnych portov sú služby, ktoré zariadenia poskytujú pre prácu na nich, a ktoré som bol schopný popísať už v kapitole 2. Niektoré však poskytujú neznáme služby, avšak môžu poslúžiť pri testovaní určitého typu útokov ako napríklad odmietnutie služby. Porty s číslom tridsaťtisíc a viac sú zariadeniami používané na spojenie s cloudom a ich konkrétne číslo nie je statické. Zariadenie si ich vyberá náhodne pred nadviazaním spojenia.

4.5 Zhrnutie

Tabuľka 4.2: Porovnanie nástrojov

Nástroje		Bitdefender	OpenVAS
Operačný systém		Windows	Linux
Odhalená zraniteľnosť	Citlivé informácie (HTTP)	áno	áno
	HTTP hlavičky zabezpečenia	áno	áno
	Lighttpd	X	áno
	TCP časová pečiatka	X	áno
	Odmietnutie služby	áno	áno
	Povolený prístup cez Telnet	X	áno

Analýza pomocou programu Bitdefender dopomohla k prvotným zisteniam možných zraniteľností jednotlivých zariadení, avšak nijak nekonkretizovala dôvody, vďaka ktorým by dané zariadenie malo byť na danú zraniteľnosť náchylné. Poukázala však fakt, že v rámci lokálneho segmentu siete existujú aj určité zraniteľnosti na základe bežiacich štandardných protokolov, ktoré som si bol schopný potvrdiť analýzou komunikácie v rámci lokálneho segmentu siete.

Naopak analýza pomocou nástroja OpenVas bola oveľa rozsiahlejšia. Tento nástroj totiž skúmal oveľa väčšie typy komunikácie s jednotlivými zariadeniami, a tiež mi poskytol de-

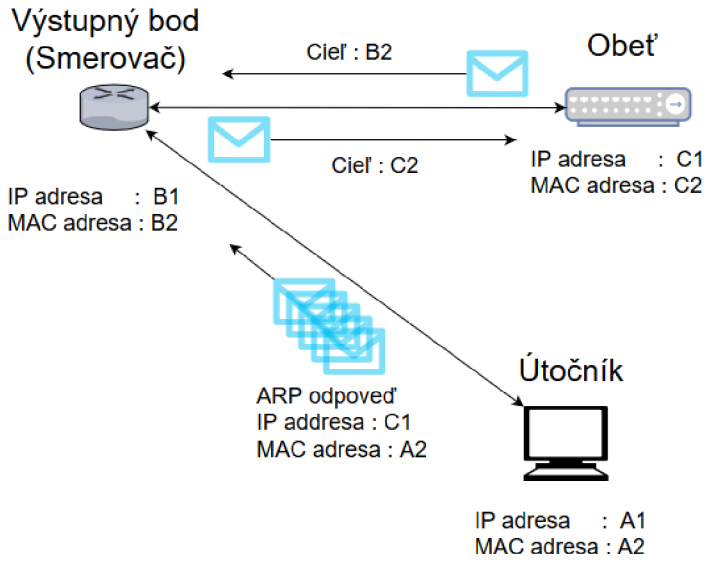
tailnejší popis jednotlivých zistení spolu so spôsobom, akým na dané zistenie prišiel. Táto analýza bola o dosť zdĺhavejšia, no priniesla tak oveľa detailnejšiu predstavu o type útokov, na ktoré by dané zariadenia mohli byť náchylné, čo bude mať veľký dopad na výber jednotlivých útokov pre testovanie. Tento nástroj tiež využíva CVE databázu na základe informácií zistených o zariadeniach za účelom zistenia ich zraniteľností. V prípade sady MioSMART však našiel len jeden záznam, a to týkajúci sa TCP časovej hlavičky, pričom tento záznam nebol pre konkrétne zariadenie z tejto sady, ale jednalo sa o zraniteľnosť nachádzajúcu sa pri ich využívaní v protokole TCP. V nasledujúcej časti sa tak zameriam na testovanie jednotlivých zistených zraniteľností a následne navrhnutie spôsobu detekcie ich prelomenia.

Kapitola 5

Testované útoky a spôsoby ich monitorovania

V tejto kapitole sa zameriam na testovanie jednotlivých útokov na odhalené zraniteľnosti [11] zariadení a následný popis možnosti ich detekcie. Útoky sú zamerané na vážnejšie zraniteľnosti, ktoré môžu obmedziť zariadeniami poskytované služby. Odhalené zraniteľnosti týkajúce sa časových pečiatok či hlavičiek zabezpečenia na protokole HTTP považujem za menej závažné, keďže buď bez útoku miereného na ďalšiu zraniteľnosť nepredstavujú vysokú hrozbu (časové pečiatky) alebo ich využitie spočíva v interakcií legitímneho užívateľa s webovou aplikáciou, ktoré môže nastať, avšak je menej pravdepodobné, keďže interakcia užívateľa so zariadením je primárne vedená pomocou mobilnej aplikácie MioSMART.

5.1 ARP spoofing

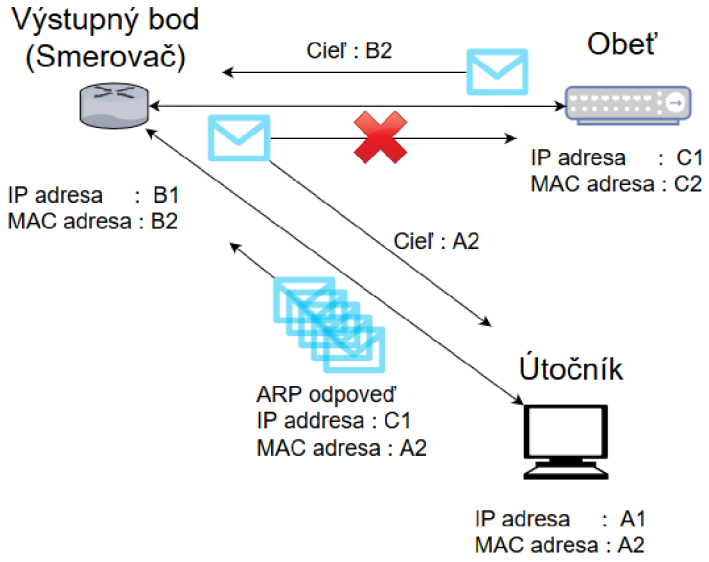


Obr. 5.1: Podvrhnutie ARP odpovede

Tento útok som si vybral na základe dvoch odhalených zraniteľností. Jednak sa jedná o samotný fakt, že zariadenia inteligentnej domácnosti používajú ARP protokol pre smerova-

nie v rámci lokálneho segmentu siete, čo umožňuje fungovanie útoku ako takého. Druhou zraniteľnosťou, ktorú sme schopný týmto útokom využiť, je prenos citlivých informácií cez nezašifrovaný HTTP protokol. V prípade úspešného útoku, je útočník schopný prečítať pakety tečúce medzi obeťou (zariadením sady MioSMART) a serverom (cloudom).

ARP spoofing [17] je útok, pri ktorom sa útočník snaží presmerovať pakety určené na doručenie obeti na seba. Toto presmerovanie dosahuje podvrhovaním ARP odpovedí (obrázok 5.1) v rámci lokálnej siete. Účelom takto podvrhnutej správy je presvedčiť zariadenie, zvyčajne výstupný bod lokálnej siete, že na IP adrese obete sa nachádza zariadenie s MAC adresou útočníka, čo má za následok presmerovanie správ určených pre IP adresu obete na zariadenie útočníka. Pre úspešný zápis takto podvrhnutej odpovede do ARP tabuľky nie je potrebný prvotný dotaz zo strany presvedčovaného zariadenia, čo znamená že automaticky po obdržaní odpovede dôjde k jej zapísaniu. Záznamy v ARP tabuľke, ktoré sú generované po prijatí odpovede, sú obnovované v určitých časových intervaloch. K ich obnoveniu dochádza zo strany zariadenia odoslaním ARP dotazu s dotazom na zmapovanú IP adresu. Keďže na tento dotaz by zariadenie obete odpovedalo svojou MAC adresou, tak by došlo k zmene mapovania do korektného stavu, čo by malo za následok prerušeniu účinku útoku (presmerovanie komunikácie). Práve preto je pri útoku nutné, aby k odosielaniu podvrhnutých ARP odpovedí zo strany útočníka dochádzalo v čo najnižších časových intervaloch počas celého trvania útoku, čo zabezpečí neustále obnovovanie podvrhnutého záznamu v ARP tabuľke.

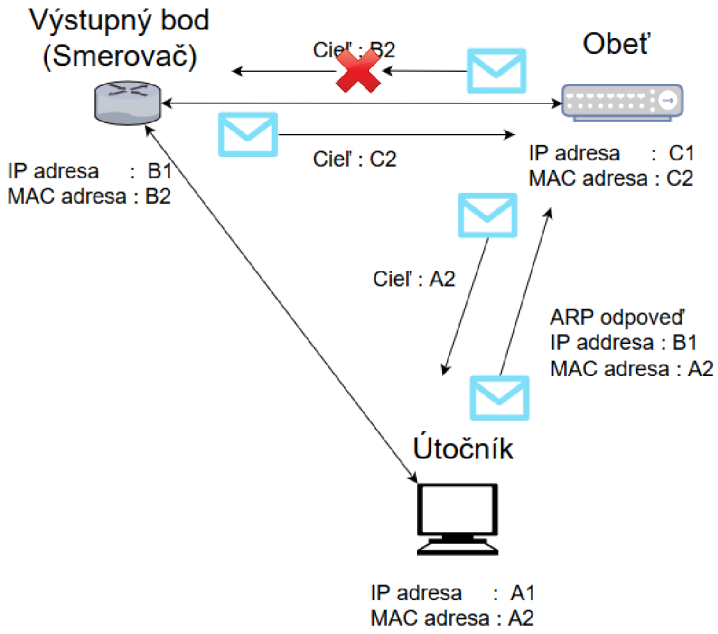


Obr. 5.2: Úspešný ARP spoofing

Týmto útokom dôjde k odchytu správ smerovaných na obeť útoku. Takto presmerované správy (obrázok 5.2) je tak útočník schopný využiť pre vyčítanie prenášaných informácií. ARP spoofing sa vo väčšine prípadov používa na otvorenie ďalších útokov, čo znamená, že pri akomkoľvek zásahu do odchytenej komunikácie sa už jedná o ďalší typ útoku. Tento útok tiež neznamená prerušenie odchytených dát, ale po ich obdržaní dôjde k následnému preposlaniu na cieľovú adresu. K tomuto preposlaniu nedochádza vo väčšine systémov automaticky, ale je potrebné povoliť *packet forwarding*.

5.1.1 ARP spoofing na zariadenie sady MioSMART

Typycký útok tohto typu teda presmeruje komunikáciu smerujúcu na zariadenie obete na útočníka. Toto presmerovanie by sa však pri komunikácii zariadení inteligentnej domácnosti z sady MioSMART vyhlo účinu. Citlivé informácie, ktorých zistenie je účelom tohto útoku, sú posielané pri komunikácií v smere zo zariadenia na cieľový server (cloud). Odchyt odpovedí z cloudu, ktoré sú zložené z jednoduchých potvrdení, by tak nemal význam. Z tohto dôvodu je potrebné tento útok viedsť na opačnú stranu komunikácie. Oproti klasickému útoku, kde sa útočník snaží presvedčiť výstupný bod sieťe o tom, že práve on sa nachádza na IP adrese obete, je tak účelom presvedčiť obeť, že výstupný bod je práve útočiace zariadenie. ARP odpoveď je tak mierená na zariadenie MioSMART sady s informáciou, že MAC adresa výstupného bodu je práve MAC adresa útočiaceho zariadenia (obrázok 5.3). Týmto spôsobom je možné docieliť stav, kedy správy zo zariadenia obete (MioSMART zariadenie) posielané na vzdialený server mimo lokálnej siete budú smerované na zariadenie útočníka.



Obr. 5.3: ARP spoofing na zariadenie sady MioSMART

Na prevedenie tohto útoku mi poslúžil nástroj arpspoof, ktorý je distribuovaný ako súčasť balíčka dsniff[1] určený pre operačný systém Linux. Pomocou tohto nástroja som bol schopný generovať podvrhnuté ARP odpovede, ktoré presvedčili obeť (zariadenie sady MioSMART), že správy posielané smerom na výstupný bod lokálnej siete má smerovať na MAC adresu môjho zariadenia (útočník). Takto presmerované správy som bol schopný odchytiť pomocou nástroja Wireshark a následne vyčítať citlivé informácie (UID zariadenia), prenášané cez nezašifrovaný HTTP protokol a využívané pri autentifikácii zariadenia sady MioSMART voči serveru. Na základe takto odchytených správ, ktoré prikladám ako PCAP súbor, som schopný vyhodnotiť tento útok za úspešný.

[1] https://www.monkey.org/~dugsong/dsniff/

5.1.2 Možnosti monitorovania ARP spoofingu

Spôsob detekcie ARP spoofingu vychádza zo skutočnosti, že pre úspešný útok je potrebné, aby útočník odosielal veľké množstvo podvrhnutých ARP odpovedí na zariadenie, ktorému sa snaží podvrhnúť zmenu mapovania siete. Detekcia tohto útoku by teda bola možná na základe veľkého počtu prijatých ARP odpovedí v krátkom časovom úseku na zariadení, či ich sledovaním v komunikácii prechádzajúcej výstupným bodom lokálnej siete.

Pre potvrdenie tohto podozrenia či samotnú detekciu je tiež možné použiť ARP tabuľky na zariadeniach v lokálnej sieti. V prípade, že je zariadenie pod týmto útokom, tak bude jeho tabuľka obsahovať viac ako jeden záznam ohľadom rovnakej MAC adresy zmapovanej na rozdielne IP adresy. Tieto záznamy vzniknú práve kôli tomu, že pre ARP spoofing musí mať útočník prístup do lokálnej siete, pričom pri jeho pripojení dôjde k zmapovaniu jeho MAC adresy na IP adresu, ktorú v rámci siete obdrží zo servera DHCP, alebo ju má nastavenú staticky. Táto tabuľka tak bude obsahovať korektný záznam o zariadení útočníka spolu s podvrhnutým záznamom na ďalšiu IP adresu.

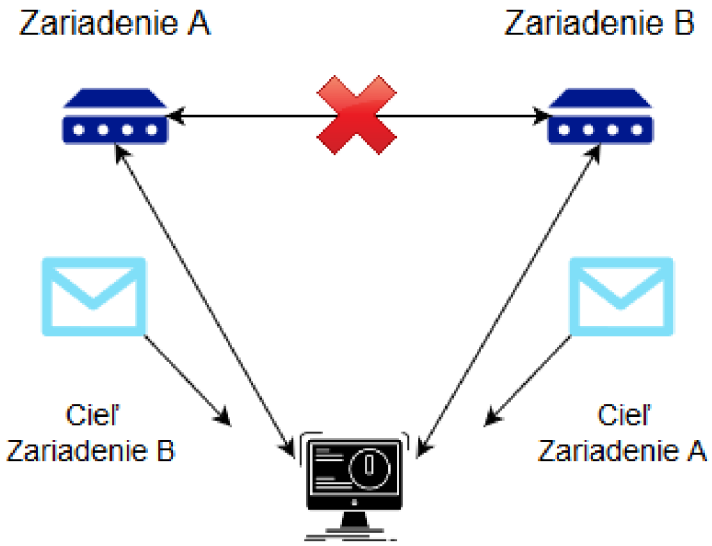
ARP spoofingu je aj veľmi jednoduché predísť použitím statických ARP záznamov, ktoré však zariadenia sady MioSMART nepodporujú. Tieto záznamy je však možné nastaviť na výstupnom bode lokálnej siete, čím predídeme presmerovaniu komunikácie smerovanej na zariadenie. Komunikácia smerujúca zo zariadenia tak môže byť aj naďalej presmerovaná.

Na obranu je tiež možné použiť *Dynamickú kontrolu ARP záznamov* [1]. Jedná sa o bezpečnostnú funkciu, ktorá zabraňuje preposielaniu neplatných ARP dotazov a odpovedí. Táto funkcia však vyžaduje zariadenie, ktoré umožňuje špehovanie DHCP záznamov a vytváranie databáze obsahujúcej priradené IP adresy na MAC adresy zariadení. Zariadenie (smerovač na úrovni L2 alebo prepínač) je na základe takto vytváranej databáze schopné zahadzovať rámce, ktorých informácie neodpovedajú uloženým kombináciám adries. Druhou funkcionalitou je obmedzenie počtu ARP rámcov, ktoré v prípade prekročenia nastavenej hodnoty (štandardne pätnásť za sekundu) spôsobia zablokovanie portu, cez ktorý boli obdržané (platí pre prepínač). Táto funkcionalita zabraňuje využitiu podvrhnutých ARP rámcov pre účely útoku DoS.

5.2 Útok Man-in-the-middle (MITM)

Tento útok podobne ako ARP spoofing samotný som si tiež vybral na základe zraniteľnosti týkajúcej sa prenosu citlivých informácií cez HTTP protokol. Túto zraniteľnosť chcem však teraz využiť pri komunikácii lokálneho užívateľa s webovou aplikáciou na jednotlivých zariadeniach, ale aj napadnúť komunikáciu medzi cloudom a zariadením inteligentnej domácnosti. Pri útoku MITM však nie len za účelom zistenia citlivých údajov, ale aj upravením či zastavením prenášaných dát.

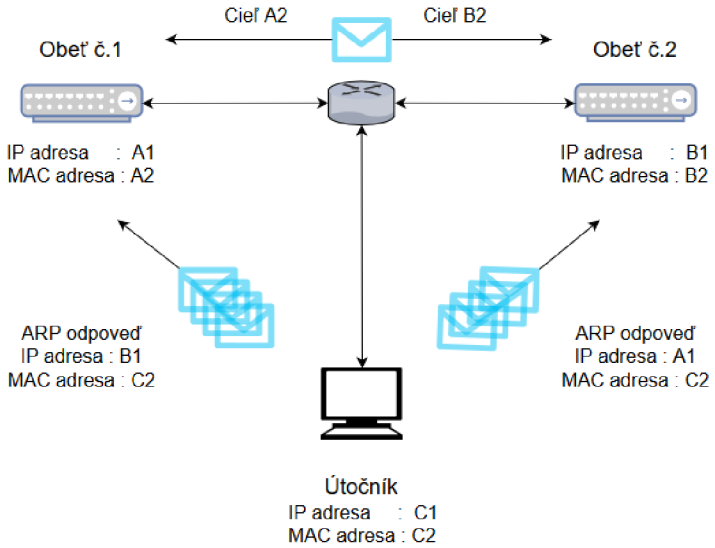
Základným princípom útoku MITM, je snaha útočníka dostať sa doprostred komunikácie medzi dvoma aktívnymi zariadeniami (obrázok 5.4), ktoré sú presvedčené o tom, že komunikujú priamo medzi sebou. Týmto spôsobom je tak útočník schopný nie len priamo sledovať prenášané dáta, ale aj pozmeniť či odstaviť komunikáciu medzi dvoma zariadeniami. V prípade, že útočník nechce odstaviť určitý typ komunikácie kompletne, tak musí byť schopný reagovať na jednotlivé požiadavky zariadení spôsobom, ktorý u nich nevyvolá podozrenie nesprávnej odozvy, a zároveň splní účel útoku zo strany útočníka. Samotný útok MITM je však možný len v prípade úspešného otváracieho útoku, ktorým útočník presmeruje komunikáciu určenú pre iné zariadenie na seba.



Obr. 5.4: Princíp útoku MITM

5.2.1 ARP spoofing ako otvárací útok

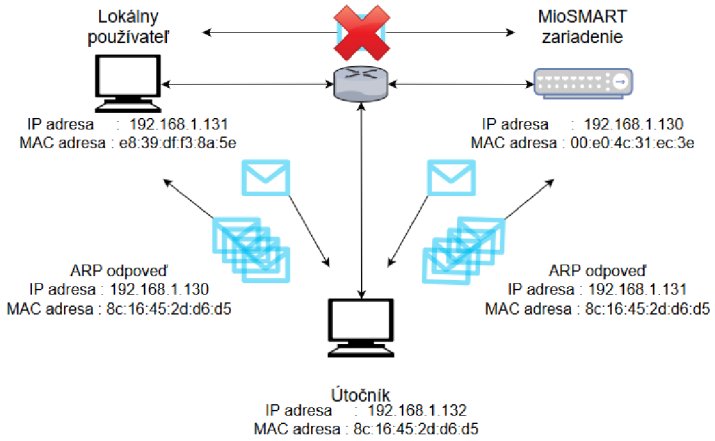
Na základe úspešného ARP spoofingu v predchádzajúcej kapitole 5.1, som sa rozhodol pre jeho využite ako otvárací útok pre útok MITM. Princíp ARP spoofingu ako takého sa pre jeho použitie nijak nemení, ale aby sa útočník úspešne dostal do stredu komunikácie medzi dvomi zariadeniami, tak je potrebné, aby tento útok viedol na obe zariadenia (obrázok 5.5) súčasne. Tým dosiahne presmerovanie komunikácie z oboch obetí na seba. Týmto spôsobom vzniká základ útoku MITM, kedy je útočník schopný sledovať komunikáciu z oboch smerov. K následnej filtrácií či úprave komunikácie dochádza priamo na zariadení útočníka úpravou odchytených prenášaných paketov.



Obr. 5.5: ARP spoofing ako otvárací útok

Na prvý pohľad sa môže zdať, že útok MITM je len akési iné pomenovanie pre dvojnásobný ARP spoofing, avšak pre útok MITM existuje viacero otváracích útokov, ktorých výsledným stavom je rovnako presmerovanie komunikácie z obetí na zariadenie útočníka. Základným rozdielom je to, že samotný ARP spoofing popisuje jednostranné presmerovanie komunikácie, pričom úprava takto presmerovaných správ môže viesť k nesprávnemu chovaniu obete a následnému prezradeniu útočníka, keďže táto úprava nemusí byť vždy korektná k opačnej strane komunikácie.

5.2.2 Útok MITM medzi zariadením a lokálnym používateľom webovej aplikácie



Obr. 5.6: Útok MITM medzi zariadením a lokálnym používateľom webovej aplikácie

Pre účel sledovania a odhalenia citlivých údajov prenášaných cez nešifrovaný HTTP protokol pri spojení medzi zariadením a lokálnym používateľom webovej aplikácie opäť postačuje využitie nástroja arpspoof, vďaka ktorému sme schopný podvrhovaním ARP odpovedí (obrázok 5.6) presmerovať komunikáciu medzi dvoma zariadeniami na zariadenie útočníka. Pre úpravu tejto komunikácie by však bolo nutné využitie ďalšieho nástroja, a preto som sa pri tomto útoku rozhodol pre nástroj Ettercap[2].

```
if (ip.proto == TCP && tcp.dst == 80) {
 if (search(DATA.data, "Authorization:")) {
  replace("Authorization:", "Authorization: AAAAAAAAAA=\r\n");
 }
}
```

Výpis 5.1: Filter pre úpravu prihlasovacích údajov pre Ettercap

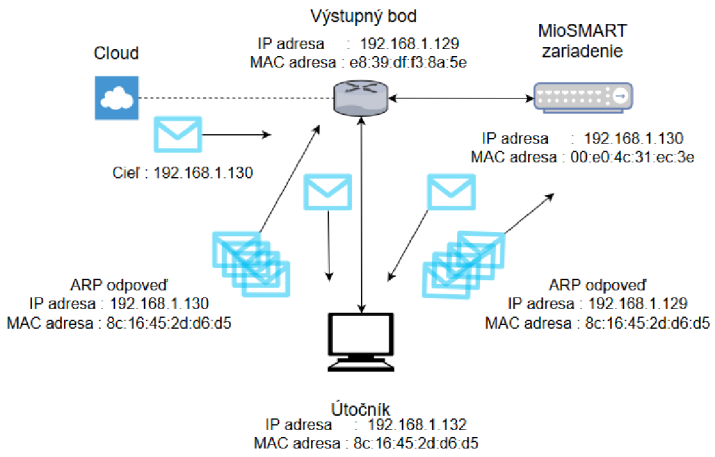
Tento nástroj má sám o sebe implementovanú funkcionalitu ARP spoofingu. Jednoduchým výberom dvoch cieľových zariadení v grafickom prostredí programu či zadaním ich IP adries pri spustení programu v príkazovom riadku pristúpi program ku generácií ARP odpovedí. Odchytenú komunikáciu vypisuje priamo v rozhraní programu či príkazovom riadku. Okrem toho však ponúka implementáciu filtrov[3] pre odchytené pakety, ktoré sú písané v syntaxi jazyka C. Filtre podporujú jednoduché konštrukcie IF, ktoré slúžia pre filtrovanie

[2] https://www.ettercap-project.org
[3] https://linux.die.net/man/8/etterfilter

konkrétneho paketu, ktorý by chcel útočník upraviť či zahodiť. Takto implementovaný filter je však potrebné pripraviť pred spustením útoku a počas doby útoku ho nie je možné meniť. Obsahovo však filter môže pozostávať z neobmedzeného počtu IF konštrukcií pre vyhodnotenie viacerých typov paketov.

Vo výpise 5.1 je filter, ktorý slúži na upravenie prihlasovacích údajov odchytených z odosielanej komunikácie lokálneho užívateľa. Táto úprava nevyžadovala zmenu paketov z opačnej strany komunikácie, keďže odpoveď zo zariadenia bola jednoduchého tvaru s oznamom zamietnutej komunikácie. Týmto útokom som tak bol schopný odoprieť lokálnemu užívateľovi prístup na zariadenie (rozbočovač či IP kameru sady MioSMART) po zadaní správnych prihlasovacích údajov. Rovnako som schopný odoprieť v tomto zapojení komunikáciu cez protokol TCP, a to oveľa jednoduchším filtrom. Tento filter by však rovnako znemožnil načítanie web stránky ako takej, čo by sice splnilo účel útoku (zamedziť prístup pre autorizovaného užívateľa), avšak oproti prvému prípadu by sa tento útok stal oveľa nápadnejším.

5.2.3 Útok MITM medzi zariadením a cloudom



Obr. 5.7: Útok MITM medzi zariadením a cloudom

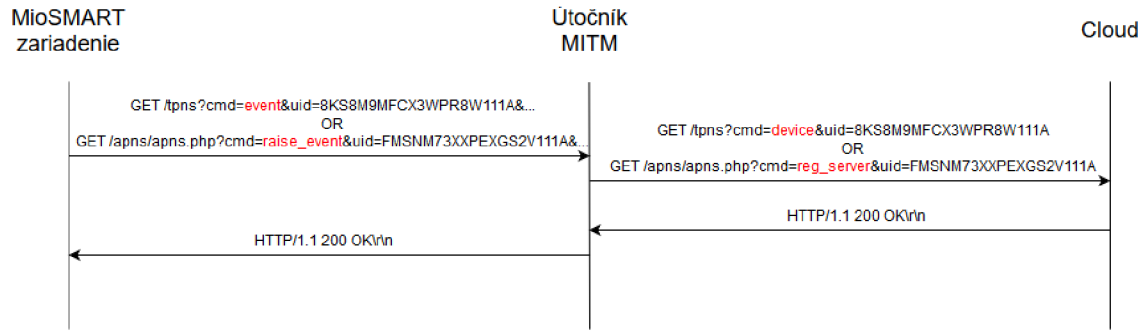
Útok na komunikáciou medzi cloudom a zariadením spočíva v presmerovaní toku smerujúceho na zariadenie z výstupného bodu siete a toku smerujúceho zo zariadenia (obrázok 5.7). Tento útok je možné viesť na rozbočovač a IP kameru sady MioSMART, keďže využívajú prenos informácií pomocou HTTP spojenia. Pri použití nástroja Ettercap sú dvomi cieľmi výstupný bod a zariadenie, ktorého komunikáciou s cloudom je cieľom napadnúť. Po úspešnom presmerovaní je tak možné sledovať dáta posielané medzi zariadením a cloudom z oboch strán komunikácie.

Hlavným účelom tohto útoku je napadnutie tejto odchytenej komunikácie. Vo výpise 5.2 je filter, ktorý slúži na zmenu paketu odpovedajúcemu správe o udalosti zo zariadenia sady MioSMART. Jeho použitím pri tomto útoku dôjde k zmene tejto správy na správu odpovedajúcu počiatočnej HTTP komunikácii zo zariadenia, ktorá slúži na počiatočne spojenie s cloudom. Informácia o udalosti, ktorá na zariadení nastala, sa tak na cloud nikdy nedostane. Na túto správu následne cloud odpovie potvrdzovacou správou, ktorá pre zariadenie značí prijatie odoslanej správy, avšak nijak nenaznačuje, že by prijatá správa bola pozmenená, a preto nie je nutné upravovať túto stranu komunikácie.

```
if (ip.proto == TCP) {
  if (search(DATA.data,
      "GET /tpns?cmd=event&uid=8KS8M9MFCX3WPR8W111A")){
        pcre_regex(DATA.data, ".+?(?=\r\n)",
        GET /tpns?cmd=device&uid=8KS8M9MFCX3WPR8W111A HTTP/1.1);
  }
  if (search(DATA.data,
     "GET /apns/apns.php?cmd=raise_event&uid=FMSNM73XXPEXGS2V111A")){
       pcre_regex(DATA.data, ".+?(?=\r\n)",
       GET /apns/apns.php?cmd=reg_server&uid=FMSNM73XXPEXGS2V111A HTTP/1.1);
  }
}
```

Výpis 5.2: Filter bez prerušenia komunikácie

Tento útok je úspešný a nenápadný aj keď pri ňom nedochádza k prerušeniu spojenia zariadenia s cloudom. Zariadenie a cloud netušia, že bola správa v spojení medzi nimi upravená (obrázok 5.8), pričom ani cloud nijak nereaguje na duplicitnú správu o zaregistrovaní zariadenia, ktorú zariadenie používa po pripojení do siete.



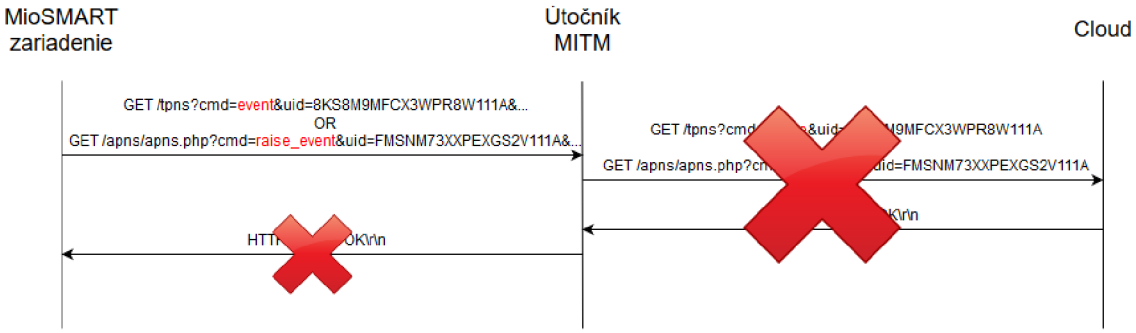
Obr. 5.8: Forma komunikácie bez prerušenia

Použitím filtra z výpisu 5.3 naopak nedochádza k zmene prenášaného paketu ako takého. Paket odpovedajúci správe o udalosti zo zariadenia MioSMART je po odchytení zahodený a nedôjde tak k jeho prijatiu cloudom. Zariadenie sa viac nepokúša o opätovné odoslanie požiadavku HTTP a dochádza tak k nesprávnej komunikácii, kedy cloud nepotvrdí prijatie HTTP správy a po určitej dobe sa pokúsi o korektné ukončenie spojenia.

```
if (ip.proto == TCP) {
  if (search(DATA.data,
     "GET /tpns?cmd=event&uid=8KS8M9MFCX3WPR8W111A")){
       drop();
  }
  if (search(DATA.data,
    "GET /apns/apns.php?cmd=raise_event&uid=FMSNM73XXPEXGS2V111A")){
      drop();
  }
}
```

Výpis 5.3: Filter pre prerušenie HTTP požiadaviek

Na ukončenie zariadenie nevie zareagovať (obrázok 5.9), pretože stále očakáva potvrdenie o prijatí správy. Vo výsledku je spojenie ukončené a zariadenie sady MioSMART sa ďalej nepokúša o jeho spätné nadviazanie.



Obr. 5.9: Forma komunikácie s prerušením

Použitím filtra z výpisu 5.4 dôjde k najnápadnejšej verzii tohoto útoku. Jednoduchým filtrom, ktorý automaticky zahadzuje odchytenú TCP a UDP komunikáciu, dôjde ku kompletnému prerušeniu spojenia medzi cloudom a zariadením. Zariadenie nie je schopné nadviazať TCP spojenie, o ktoré sa po zaznamenanej udalosti pokúsi niekoľko krát približne do dvoch minút od prvého neúspešného pokusu (približne päťkrát). Následne sa o nadviazanie nepokúša až do momentu novej zaznamenanej udalosti. Prerušené UDP spojenie spôsobí, že pre klienta je v aplikácií MioSMART zariadenie nedosiahnuteľné. Nie je možné sa dostať k jeho nastaveniam a ani k informáciám o zaznamenaných udalostiach.

```
if (ip.proto == TCP || ip.proto == UDP) {
   drop();
}
```

Výpis 5.4: Filter pre kompletné prerušenie spojenia

Pri všetkých útokoch sa mi tak podarilo upraviť či odstaviť komunikáciu, ktorá by následne vyvolala v aplikácii MioSMART oznámenie o zaznamenanej udalosti, avšak dáta o danej udalosti (čas, kedy nastala a jej typ) sa na cloud dostali pomocou UDP spojenia a bolo ich tak možné v mobilnej aplikácii dohľadať. Jediný prípad, kedy nie je možné tieto záznamy dohľadať, je v kombinácií s filtrom UDP spojenia, ktorý preruší spojenie so zariadením a cloudom. V takom prípade je načítanie záznamov na zariadení prostredníctvom aplikácie MioSMART neúspešné, avšak keďže záznamy o udalostiach sa nachádzajú priamo na zariadení, tak v prípade prerušenia útoku je opäť možné ich dohľadať priamo z aplikácie MioSMART až do reštartu zariadenia.

5.2.4 Možnosti monitorovania útoku MITM

Keďže jedným z najpoužívanejších otváracích útokov pre útok MITM je práve ARP spoofing tak medzi spôsoby monitorovania spadajú spôsoby popísané v kapitole 5.1.2. K odhaleniu útoku MITM môže dôjsť v prípade, že dôjde k detekcii duplicitných záznamov v ARP tabuľke viac ako jedného zariadenia siete, medzi ktorými k danej komunikácii dochádza alebo cez neho aspoň prechádza (výstupný bod). V prípade tejto detekcie je jasné, že je útočník schopný minimálne čítať obe strany komunikácie a je potenciálne schopný danú komunikáciu meniť. Potvrdenie o tom, či tak útočník aj činí, je možné zistiť sledovaním komunikácie

prechádzajúcej cez výstupný bod siete. Ak poznáme tvar dát prenášaných počas bežnej prevádzky, tak sme schopný porovnať dáta, ktoré mali byť prenesené v danom momente oproti dátam reálne preneseným. Rovnako je tiež isté, že k ich prenosu cez výstupný bod dôjde viac ako jeden krát. V prípade tohto útoku sú dáta prechádzajúce výstupným bodom presmerované na zariadenie útočníka, čo značí ich prvý výskyt. Ak však útočník neodstaví komunikáciu, ale snaží sa ju nenápadne pozmeniť, tak musí upravené dáta preposlať smerom na výstupný bod, kde sme teda schopný sledovať ich druhý výskyt v upravenom či pôvodnom stave. Ak je však útočník schopný viesť tento útok mimo lokálnej siete na spojenie zariadenia s cloudom, tak je bežná detekcia nemožná. K jeho odhaleniu by mohlo dôjsť len porovnaním dát odoslaných priamo zo zariadenia s dátami, ktoré dorazili na cloud.

Pri útoku medzi lokálnym používateľom a zariadením je navyše možné sledovať zmeny dát priamo v zdrojovom kóde webovej aplikácie na zariadení užívateľa. Pri úspešnom MITM útoku mohlo dôjsť k podvrhnutiu časti škodlivého zdrojového kódu, ktorý sa tam počas bežnej prevádzky nenachádza a bol podvrhnutý útočníkom za účelom nesprávnej manipulácie s nastaveniami zariadenia či vylákaním ďalších citlivých údajov od lokálneho klienta.

Obrana proti útokom cieleným na tento protokol je implementácia SSL alebo TLS šifrovania na protokole HTTP. Jedná sa o rozšírenie na protokol HTTPs, ktorý využíva dlhodobé verejné a súkromné kľúče na generovanie kľúča pre konkrétnu reláciu, ktorý je následne použitý na šifrovanie sieťového toku medzi klientom a serverom (zariadením MioSMART a cloudom). HTTPs šifruje kompletný obsah správy vrátane HTTP hlavičiek. Za predpokladu, že sa nejedná o hardvérovo náročný kryptografický útok by tak útočník mal byť schopný odhaliť maximálne IP adresy oboch zúčastnených strán spolu s názvami domén. Preposielané informácie by však zostali ochránené pred očami útočníka v zašifrovanej forme.

5.3 Odmietnutie služby - SYN flood útok

Odmietnutie služby (DOS) je typ útoku na IoT zariadenia za účelom znepristúpnenia zariadením poskytovanej služby užívateľovi, ktorý o ňu žiada. Jedná sa o útok, pri ktorom dochádza k zahlteniu IoT zariadenia neoprávenými požiadavkami o službu, ktorú sa útokom pokúšame znefunkčniť. Keďže tento útok požiadavkami zahlcuje rezervované zdroje zariadenia, môže tak obmedziť či znefunkčniť ostatné poskytované služby. Úspešný DOS útok nemusí nutne znamenať plné odstavenie poskytovanej služby, ale môže sa prejaviť jej neobvyklím spomalením či nedostupnosťou len určitej časti služby, a to obzvlášť pri vačších množstvách prenášaných dát.

SYN flood [3] útok sa pokúša o znefunkčnenie služby pomocou veľkého množstva neúplnej trojfázovej synchronizácie. Za normálnych okolností pri nadväzovaní TCP spojenia trojfázovou synchronizáciou dochádza k výmenie správ medzi klientom a serverom v troch krokoch nasledovne.

1. Klient pošle na server požiadavku typu SYN pre nadviazanie spojenia.
2. Server potvrdí požiadavku odoslaním správy typu SYN-ACK na klienta.
3. Klient potvrdí prijatie potvrdenia správou typu ACK.

Po úspešnom splnení týchto troch krokov dôjde k vytvoreniu spojenia medzi klientom a serverom, pričom následne môže dôjsť ku komunikácii ohľadom požadovanej služby. SYN flood útok funguje na základe neodoslania správy ACK na server. Toto sa dá dosiahnuť dvomi spôsobmi. Klient po prijatí správy SYN-ACK bude naprogramovaný tak, aby určite

neodoslal správu ACK, alebo pri odosielaní správy SYN použije klient falošnú IP adresu, čo zpôsobí, že klient na falošnej IP adrese buď neexistuje, alebo aj tak neodpovie na dotázanú správu, pretože pred tým neposlal správu SYN.

Server tak očakáva ACK správu po určitú dobu, pretože jej dočasná absencia môže byť spojená so spotrebovaním šírky pásma siete. Pri takto otvorených spojeniach však server využíva svoje vlastné zdroje namapované na klienta požadujúceho službu, čo môže mat za následok vyplýtvanie voľných zdrojov na servery. Ak dôjde k stavu, kedy server všetky tieto zdroje vyčerpá, nie je tak následne schopný vyhovieť ďalšej požiadavke legitímneho či škodlivého klienta. Niektoré systémy používajú zdieľané zdroje pre viaceré služby či dokonca pre svoju vlastnú systémovú funkčnosť, čo môže pri takto vytvorenom stave znamenať ich kompletné znefunkčnenie či až pád samotného systému.

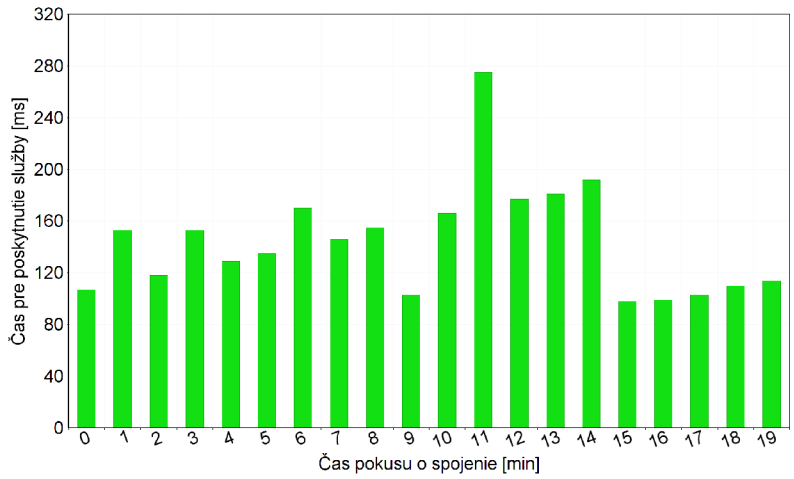
5.3.1 Útok na zariadenie sady MioSMART

Na tento útok som si pripravil vlastný program v jazyku *C++*. Pracuje na báze RAW soketov a odosiela pripravený SYN paket na cieľové zariadenie. Časová doba na odoslanie tohoto paketu nie je staticky určena a závisí len na výpočetnej rýchlosti útočiaceho zariadenia. Pomocou tohoto programu som bol schopný generovať odoslané pakety o početnosti približne pätnásť tisíc za sekundu. Keďže účelom tohoto útoku je zahltiť rezervované zdroje na zariadení, na ktoré je útok vedení, sledoval som tak jeho účinky v časových intervaloch po určitú dobu. Sledovanie útoku pre vytvorenie štatistík by zo strany útočníka bolo možné pri použití jeho vlastnej zdrojovej IP adresy, čo by však dvojnásobne zaťažovalo útočiace zariadenie, čím by došlo aj k nižšej početnosti odoslaných škodlivých paketov. Zároveň by tak odchytené PCAP súbory obsahovali veľké množstvo záznamov, ktoré pre zistenie dostupnosti zariadenia pod týmto útokom nie sú zrovna najlepšou voľbou.

Pre účeli sledovania účinkov útoku som sa tak rozhodol vytvoriť ešte jeden nástroj, ktorý počas útoku pobeží na druhom zariadení. V princípe sa tak druhé zariadenie pokúša o spojenie zo zariadením pod účinkami DoS útoku. Nástroj nadvazuje dvadsať samostatných spojení oddelených časovým intervalom a sleduje čas potrebný pre vybavenie požiadavky spolu si priatím kompletných dát pre načítanie webového prístupu. V prípade, že dôjde k prekročeniu zvoleného timeoutu, je pokus o spojenie vyhodnotený za neúspešný. Rovnako tak je pokus neúspešný v prípade nepodareného nadviazania spojenia pre komunikáciu TCP. Timeout nepredstavuje maximálny časový interval, do ktorého musí dôjsť ku kompletnému preneseniu dát, ale znamená časový interval, do ktorého musí program obdržať odpoveď na jednu odoslanú požiadavku v celkovej komunikácii. Vo výsledku tak minútový timeout môže znamenať vyhovenie požiadavky do niekoľko násobného času intervalu ale len za predpokladu, že všetky odpovede boli obdržané do minúty od požiadavky o ne. Popis použitia oboch nástrojov je zdokumentovaný v prílohe A.

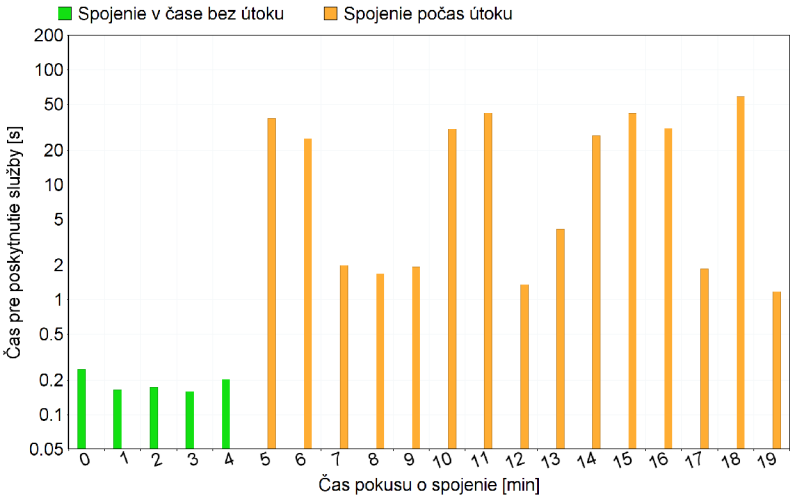
5.3.2 Štatistické výsledky

Graf na obrázku 5.10 zobrazuje časy potrebné pre kompletné spojenie lokálneho užívateľa so zariadením sady MioSMART. Táto štatistiky bola vytvorená na základe spojení nadväzovaných počas stavu bez vedeného útoku. Všetky nadviazane spojenia skončili úspešne, pričom čas potrebný na ich vyhotovenie nikdy neprekročil tristo milisekúnd. Pri bežnej prevádzke nedošlo k žiadnemu výpadku či oneskoreniu.



Obr. 5.10: Sledované spojenia počas bežnej prevádzky (bez útoku)

Pre sledovanie zmien, ktoré nastanú pri útoku, som nástroj pre nadviazanie spojení spustil počas doby, kedy nebežal útok na zariadenie s hodnotou timeoutu nastavenej na tridsať sekúnd. Po dobu piatich minút je možno vidieť, že čas potrebný na kompletné vybavenie požiadavky sa pohybuje v rozmedzí do približne dvesto päťdesiatich milisekúnd.



Obr. 5.11: Účinky DoS útoku oproti bežnej prevádzke

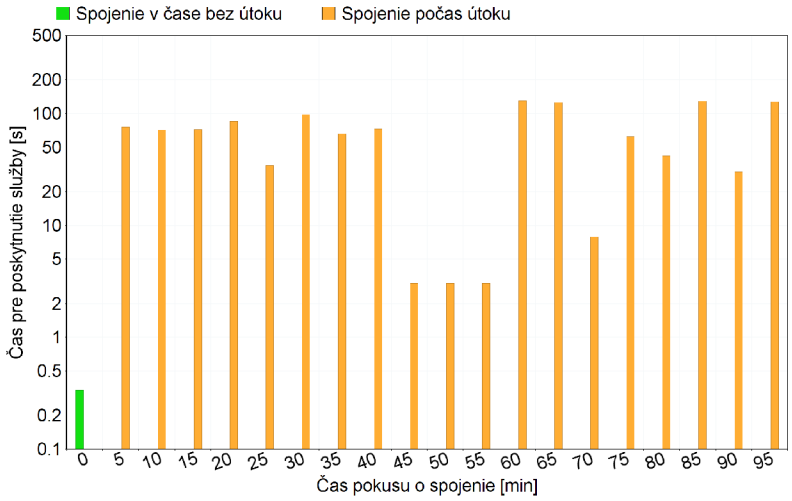
Útok bol spustený v piatej minúte a trval nasledujúcich pätnásť minút. Z grafu na obrázku 5.11 možno vidieť, že účinky útoku vysoko obmedzujú prevádzku už po jednej minúte od jeho spustenia. Pri sledovaní účinkov po dobu dvadsiatich minút sa však nedá povedať, že by sa jednalo o konzistentný stav, keďže pomerne veľká časť požiadaviek bola vybavená do časového intervalu (v rozmedzí od jednej do tridsiatich sekúnd), avšak viac ako

polovica požiadaviek končila v stave, kedy nebolo spojenie vybavené do určeného timeoutu (tabuľka 5.1). V dvoch prípadoch sa klientskej aplikácií ani nepodarilo nadviazať spojenie TCP, čo značí, že už prvá TCP požiadavka pre nadviazanie spojenia bola z dôvodu zahltenia zariadením prehliadnutá.

Tabuľka 5.1: Úspešnosť spojení pri počiatku útoku

	Požiadavky v čase	Percentuálne	Poznámka
Úspešné	0, 1, 2, 3, 4, 6, 7, 8, 9, 12, 14, 19	60%	Požiadavky pred začatím útoku tvoria 25%.
Neúspešné (timeout)	5, 10, 11, 15, 16, 18	30%	Spojenie prekročilo nastavený timeout.
Neúspešné (TCP)	13, 17	10%	Neúspešné nadviazanie TCP komunikácie.

Pri dlhodobom testovaní som sa rozhodol zvýšiť timeout na hodnotu šesťdesiatich sekúnd. Z výsledkov (obrázok 5.12) je možno vidieť, že ani pri zvýšenom povolenom čase pre vybavenie spojenia nedošlo k zvýšeniu počtu úspešných spojení. Pri dlhodobých účinkoch možno práve naopak sledovať jeho zníženie. K úspešnému odbaveniu požiadavky došlo v menšom počte oproti nevybaveným požiadavkám, ktoré končili timeoutom aj v jeho zvýšených hodnotách. Tri požiadavky v čase od päťdesiatej až šesťdesiatej minúty boli neúspešné už pri pokuse o nadviazanie spojenia čo opäť značí, že v danom čase bolo zariadenie natoľko zahltené, že nebolo schopné prijímať požiadavky o nadviazanie spojenia. Z týchto výsledkov je možné usúdiť, že útok bol úspešný, keďže počas celej sledovanej doby pod útokom dochádzalo k úspešnému spojeniu len v malom počte (tabuľka 5.2) a aj v týchto prípadoch bolo spojenie niekoľkonásobne pomalšie ako počas bežnej prevádzky.



Obr. 5.12: Účinky pri dlhodobom útoku DoS

Tabuľka 5.2: Úspešnosť spojení pri dlhodobom útoku

	Požiadavky v čase	Percentuálne	Poznámka
Úspešné	0, 10, 70, 80, 90	25%	Požiadavky pred začatím útoku tvoria 5%.
Neúspešné (timeout)	5, 15, 20, 25, 30, 35, 40, 60, 65, 75, 85, 95	60%	Spojenie prekročilo nastavený timeout.
Neúspešné (TCP)	45, 50, 55	15%	Neúspešné nadviazanie TCP komunikácie.

5.3.3 Zhrnutie výsledkov

Prevedenými testami som bol schopný potvrdiť zraniteľnosť vo forme odmietnutia služby zo strany zariadenia. Aj keď z výsledkov vyplýva, že zariadenie pod útokom bolo schopné vyhovieť určitému počtu požiadaviek, a teda z pravidla nedošlo ku kompletnému odstaveniu služby, tak treba zobrať do úvahy aj skutočnosť, že testovanie bolo prevádzané na jednotlivých na sebe nezávislých požiadavkách. V prípade reálneho využitia služby klientom by s určitosťou dochádzalo k väčšiemu počtu výpadkov spojenia, ktoré by nastávali vždy v prípade nevyhovenia jednej z na seba nadväzujúcich požiadaviek.

Útok na práve aktívny port zariadenia pre spojenie s cloudom by v tomto prípade nebol účinný, keďže sa nejedná o zariadením poskytovanú službu. Pri spojení využívajúcom práve aktívny port dochádza k jeho použitiu pre nadviazanie spojenia iniciovaného zariadením, a teda tento útok nebude schopný odoprieť danú službu, keďže sa nejedná o službu, ktorú poskytuje zariadenie pre prichádzajúce požiadavky [3]. Jeho prevedenie malo pre mňa len informatívny a potvrdzujúci účel, keďže zariadenie generovalo 100% spojení s cloudom a jediné obmedzenie bolo v mierne zvýšenej odozve približne do jednej sekundy, čo pripisujem zahlteniu šírky pásma v rámci lokálnej siete.

5.3.4 Možnosti monitorovania DoS útoku

Monitorovanie prebiehajúceho DoS útoku je možné vytvorením sondy na mieste, ktorým prechádza kompletná komunikácia dnu a von zo siete. V prípade menšej domácej siete (predpoklad, že k útoku nedochádza priamo z vnútra siete) umiestnime sondu, počítač s bežiacim programom na odchyt paketov(napr. už spomenutý Wireshark) pred výstupný bod siete. Z takto odchytenej komunikácie je následne možné odhaliť DoS útok na zariadenia v lokálnej sieti za ním na základe veľkého počtu prichádzajúcich pripojení z jednej či viacerých zdrojových adries, keďže žiadna z funkcionalít zariadení takéto pripojenia nevyžaduje. Navyše okrem adries z priestoru Amazon serverov k zariadeniam nepristupuje žiadne ďalšie vzdialené zariadenie okrem zariadení samotného klienta. Jedným z ukazovateľov je tak aj prístup zo zariadenia mimo tohto adresného priestoru, čo môže tiež značiť pokus o DoS útok. Ak sa však útočník pokúsi maskovať útok použitím falšovaných zdrojových IP adries z adresného priestoru Amazon serverov, stále je tak možné vyhodnotiť prichádzajúce pakety za DoS útok už pri rádovo desiatkach či stovkách pripojení v krátkom časovom úseku, keďže k pokusu o nadviazanie daného spojenia prichádza z vzdialeného zariadenia a toto spojenie je za normálnych okolností iniciované zo strany lokálneho zariadenia. V prípade podozrenia, že mohlo dôjsť ku skompromitovaniu lokálnej siete a útočník tak môže útočiť priamo z nej, je potrebné umiestniť túto sondu priamo pred jednotlivé zariadenia v sieti alebo priamo na výstupný bod siete.

5.4 Zhrnutie

Na základe nájdených zraniteľností som bol schopný previesť sieťové útoky na zariadenia sady MioSMART, ktorými som potvrdil využitie odhalených zraniteľností pri útoku na inteligentnú domácnosť so zameraním na najkritickejšie z nich. Útoky vedené za pomoci ARP protokolu je v rámci lokálnej siete možné sledovať pomerne jednoduchým spôsobom a je tiež možné predísť ich využitiu napríklad statickým vytvorením ARP tabuľky na výstupnom bode či jednotlivých zariadeniach. V prípade, že je útočník natoľko schopný, aby vyššie spomenuté útoky viedol z miesta mimo lokálnej siete, môže sa tak útok stať neodhaliteľným. V prípade, že sa podarí útočníkovi presmerovať na seba nešifrovanú HTTP komunikáciu tečúcu medzi lokálnym zariadením a vzdialeným cloudom, tak ju môže voľne čítať či upravovať, keďže v nej prenášané údaje neobsahujú žiadny spôsob šifrovania a tiež toto spojenie neobsahuje žiadny spôsob P2P zabezpečenia, ktorá by zabezpečila, že k nej dochádza výlučne medzi týmito dvoma zariadeniami.

Zraniteľnosť na DoS útok je naopak vždy možné monitorovať v rámci lokálnej siete, keďže samotný útok sa prejavuje vysokým množstvom prichádzajúcich paketov z jednej či viacerých zdrojových adries. Bežný tok v lokálnej sieti sa tak zvýši niekoľkonásobne, čo je samo o sebe indikátorom prebiehajúceho DoS útoku, keďže tieto zariadenia nie sú stavané na vysoký počet prístupov v krátkom časovom úseku. Testovanie však dokázalo, že okrem služieb poskytovaných zariadením pre lokálnych užívateľov nedochádza k obmedzeniu rýchlosti prenosu notifikácií zo zariadenia na cloud. Táto skutočnosť značí, že útok ako taký by mohol byť zaradení medzi menej závažnejšie, keďže nedochádza k obmedzeniu hlavnej funkcionality zariadení. Je potrebné myslieť aj prenos väčších dát zo zariadenia na cloud (napr. video prenos z IP kamery na cloud), ktorý je v prípade takto zahlcovanej siete tiež obmedzený, aj keď nie z dôvodu zahltenia zariadenia, ale z dôvodu zaplnenia šírky pásma v rámci lokálnej siete. Preto stále radím danú zraniteľnosť ku kritickejším v rámci inteligentnej domácnosti ako takej.

Menej kritické zraniteľnosti buď samy o sebe nepredstavujú závažnejšie zraniteľné miesto (časové hlavičky 4.3.5), alebo je na ich využitie potrebné väčšie úsilie s menšou pravdepodobnosťou reálneho úspechu (HTTP hlavičky zabezpečenia 4.3.2). Úspešné podvrhnutie kódu pre webovú aplikáciu na zariadení by predpokladalo interakciu s lokálnym klientom, ktorá vo väčšine prípadov ani nemusí nastať, keďže primárne sú zariadenia inteligentnej domácnosti určené pre prácu s mobilnou aplikáciou, o ktorej väčšina bežných užívateľov ani len netuší.

Kapitola 6

Záver

Zariadenia pre inteligentnú domácnosť sa v dnešnej dobe tešia veľkému dopytu. Na trhu sa objavuje veľké množstvo zariadení od neznámych či nových spoločností so zameraním práve na jednoduché zariadenia zabezpečenia či pomocných prác v domácnosti. Tieto sady pre oči bežného zákazníka predstavujú spoľahlivé vykonanie požadovanej funkcionality. Bežný zákazník je totiž schopný hodnotiť hardvérové funkcionality v spojení s funkčným vzdialeným ovládaním či oznámeniami, ktoré v prípade použitia v bežnej zabezpečenej domácej sieti pracujú podľa nastavení. Kritickým miestom je tak samotná sieťová komunikácia, ktorá poskytuje takmer nulové až žiadne spôsoby zabezpečenia. Táto komunikácia je pre bežného užívateľa akousi skrytou oblasťou, o ktorej nič nevie a ani nebude skúmať ako samotná komunikácia prebieha za predpokladu, že nedochádza na zariadení k nesplneniu ním poskytovaných funkcií.

6.1 Zhodnotenie výsledkov práce

Táto práca sa zameriava práve na spomínanú sieťovú komunikáciu. Na jej vyhotovenie mi poslúžili zariadenia pre inteligentnú domácnosť zo sady MioSMART. Popis týchto zariadení a ich užívateľských možností sa nachádza v kapitole 2. Typy komunikácií, ktoré zariadenia používajú na splnenie nimi poskytovaných funkcionalít sú rozpísané v kapitole 3.

V rámci práce som bol schopný odhaliť už spomínané nevyužitie zabezpečených spôsobov prenosu dát medzi jednotlivými zariadeniami a vzdialeným cloudom. Spojenie medzi výrobcom poskytnutou aplikáciou a cloudom naopak využíva šifrované verzie spojenia (TLSv1.2), a preto sa v práci zameriavam na komunikáciu zariadenia s cloudom a zariadenia s lokálnym užívateľom. Tieto skutočnosti sú bližšie popísané v kapitole 4 vo forme jednotlivých zraniteľností, ktoré reflektujú spôsoby komunikácie týchto zariadení. K ich odhaleniu mi poslúžili voľne dostupné nástroje pre detekciu zraniteľností domácich zariadení: Bitdefender, OpenVAS, Nmap (detekcia poskytovaných služieb).

Na základe odhalených zraniteľností som bol schopný demonštrovať ich zneužitie pri vybraných typoch útokov so zameraním na kritickejšie miesta a následne navrhnúť možnosti detekcie použitých útokov (kapitola 5).

Práca tak ponúka náhľad do problematických miest technológií inteligentných domácností a môže poslúžiť na ošetrenie kritickejších miest, alebo ako návod na sledovanie či obranu proti jednotlivým útokom vyplývajúcich z nájdených zraniteľností.

6.2 Výstupy práce

Výstupmi práce sú programy a postupy pre štatistické testovanie útokov využívajúcich kritické zraniteľnosti zariadení spolu s celkovým prehľadom odhalených zraniteľností a ich možného využitia. Výstupmi jednotlivých častí práce a jednotlivých testovaní sú súbory typu PCAP s obsahom odchytenej komunikácie počas útoku či bežnej prevádzky. Všetky zdrojové kódy, odchytené komunikácie, vstupné filtre pre program Ettercap a generované štatistické výstupy sa nachádzajú na priloženom médiu.

Literatúra

[1] Dynamic ARP Inspection. [Online; navštívené 30.04.2019].
URL https://documentation.meraki.com/MS/Other_Topics/Dynamic_ARP_Inspection?dtid=osscdc000283

[2] HTTP headers. [Online; navštívené 10.02.2019].
URL https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers#Security

[3] CERT Advisories. December 1996, [Online; navštívené 15.03.2019; revidované 29.11.2000].
URL https://resources.sei.cmu.edu/library/asset-view.cfm?assetID=496170

[4] Boucadair, M.; Telecom, F.; Penno, R.; ai.: Universal Plug and Play (UPnP). 6970, Júl 2013.

[5] Camarillo, G.: Peer-to-Peer (P2P). 5694, November 2009.

[6] Droms, R.: Dynamic Host Configuration Protocol. 2131, Marec 1997.

[7] Fielding, R.; Gettys, J.; Mogul, J.; ai.: Hypertext Transfer Protocol. 2616, Jún 1999.

[8] Jackson, B.: Hardening Your HTTP Security Headers. [Online; navštívené 10.02.2019].
URL https://www.keycdn.com/blog/http-security-headers

[9] Mills, D. L.: Network Time Protocol (NTP). 955, September 1985.

[10] Mockapetris, P.: Domain Name Services. 1035, Október 1987.

[11] Northcutt, S.; Shenk, J.; Shackleford, D.; ai.: Penetration Testing: Assessing Your Overall Security Before Attackers Do. November 2006, [Online; navštívené 30.04.2019].
URL https://www.sans.org/reading-room/whitepapers/analyst/penetration-testing-assessing-security-attackers-34635

[12] Plummer, D. C.: An Ethernet Address Resolution Protocol. 826, November 1982.

[13] Postel, J.: Internet Protocol. 791, September 1981.

[14] Postel, J.; Reynolds, J.: File Transfer Protocol (FTP). 959, Október 1985.

[15] Schulzrinne, H.; Rao, A.; Lanphier, R.; ai.: Real-Time Streaming Protocol. 2326, December 2016.

[16] Steward, B.: TCP port 23. 2002, [Online; navštívené 14.03.2019].
URL https://www.giac.org/paper/gcih/328/tcp-port-23/103233

[17] Sukkar, G. A.; Saifan, R.; Khwaldeh, S.; ai.: Address Resolution Protocol (ARP): Spoofing Attack and Proposed Defense. Júl 2016, [Online; navštívené 10.02.2019].
URL https://www.scirp.org/journal/PaperInformation.aspx?PaperID=68371

Príloha A

Programy v jazyku C++

V tejto kapitole je uvedený popis spúštania naprogramovaných nástrojov pre testovanie útoku DoS a mapovanie udalostí na zariadeniach sady MioSMART. Všetky try programy sú vytvorené príkazom make na Linux OS s nasledujúcimi názvami:

- dosattack (syn flood útok),
- dosconnect (štatistiky syn flood útoku),
- mapping (mapovanie udalostí na zariadeniach).

Dosattack

./dosattack [-d zariadenie] [-i IP adresa] [-p port]

Pre útok je minimálne potrebný parameter pre IP adresu a port. Zariadenie nie je nutné špecifikovať, avšak v prípade jeho použitia je parameter pre IP adresu ignorovaný. Pri jeho použití dochádza k útoku na IP adresu špecifikovaného zariadenia podľa obrázka 2.1.

- -d zariadenie, špecifikuje zariadenie zo sady MioSMART s možnými hodnotami:
  - 1 - rozbočovač
  - 2 - kamera
  - 3 - brána
- -i IP adresa, špecifikuje IPv4 adresu zariadenia
- -p port, špecifikuje port služby (typu TCP)

Dosconnect

./dosconnect [-d zariadenie] [-i IP adresa] [-t timeout] [-l pauza]

Jediný voliteľný parameter je IP adresa. V prípade jeho použitia dôjde k pokusom o pripojenie na špecifikovanú IPv4 adresu, ale ak zadaný nie je tak k pripojeniam dochádza na IP adresu podľa zvoleného zariadenia z obrázka 2.1.

- -d zariadenie, špecifikuje zariadenie zo sady MioSMART (nutné pre formát správy) s možnými hodnotami (brána neposkytuje webovú službu):
  - 1 - rozbočovač
  - 2 - kamera
- -i IP adresa, špecifikuje IPv4 adresu zariadenia
- -t timeout, špecifikuje timeout pre odpoveď zo zariadenia v sekundách
- -l pauza, špecifikuje časový interval medzi pokusmi o pripojenie v sekundách

Mapping

Tento nástroj je možné použiť pre mapovanie ID udalostí, na ktoré vie cloud reagovať odoslaním oznámenia pre aplikáciu MioSMART. Mapovanie nie je možné automaticky, pričom názov udalosti je možné sledovať len v prijatých oznámeniach v aplikácii. Mapovanie tak slúži len na odhalenie ostatných typov oznámení, ktoré tieto konkrétne zariadenia negenerovali a k zisteniu o štandardnom type oznámenia.

./mapping [-d zariadenie] [-p ID udalosti]

- -d zariadenie, špecifikuje zariadenie zo sady MioSMART (nutné pre formát správy) s možnými hodnotami:
  - 1 - rozbočovač
  - 2 - kamera
- -p ID udalosti, špecifikuje udalosť, ktorej oznámenie má program odoslať na cloud (v prípade nepoužitej ID cloud generuje štandardné oznámenie Älarm")