VHODNÁ STRATEGIE PRO DETEKCI BEZPEČNOSTNÍCH INCIDENTŮ V PRŮMYSLOVÝCH SÍTÍCH

AN APPROPRIATE STRATEGY FOR DETECTING SECURITY INCIDENTS IN INDUSTRIAL NETWORKS

Bc. Karel Kuchař
Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, Technická 12, 616 00 Brno, Česká republika
Kontakt: xkucha24@vutbr.cz

Bc. Eva Holasová
Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, Technická 12, 616 00 Brno, Česká republika
Kontakt: xholas08@vutbr.cz

Ing. Radek Fujdiak, Ph.D.
Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, Technická 12, 616 00 Brno, Česká republika
Kontakt: fujdiak@vutbr.cz

ABSTRAKT
Tento článek se zaměřuje na prostředí a bezpečnost průmyslových sítí. Průmyslová síť běžně pracuje se staršími zařízeními, které neposkytují bezpečnost na úrovni dnešních požadavků, a i samotné protokoly mnohdy zabezpečení neposkytují. Tyto bezpečnostní nedostatky je nutné řešit nejen z důvodu digitalizace. Je tak nutné využít další techniky, které napomohou bezpečnost zajistit. Z tohoto důvodu je vhodné do sítě umístit další prvky, které budou zajišťovat dodatečné zabezpečení a provádět monitorování sítě. Monitoring sítě je typicky zajištěn pomocí nástrojů IDS (Intrusion Detection System). Tyto systémy provádí detekci definovaných signatur a anomálií. Článek popisuje vytvoření metod k detekci bezpečnostních incidentů pomocí rozpoznání anomálií v síťovém provozu. Navržené metody jsou zaměřeny na detekci útoku DoS v průmyslovém protokolu Modbus a provedení operace mimo běžný interval v protokolu DNP3. Funkčnost vytvořených metod je otestována v IDS systému Zeek.
Klíčové slova: Modbus, DNP3, Průmyslové sítě, Bezpečnostní incidenty.

ABSTRACT
This article is focused on the environment and security networks. The industrial network typically works with older devices that do not provide security at the level of today's requirements, even protocols often do not support security at a sufficient level. It is necessary to handle with these security issues due to digitization. It is thus possible to provide other techniques that will help with security. For this reason, it is possible to place additional elements that will provide additional security and ensure monitoring of the network. Network monitoring is provided by Intrusion Detection System (IDS) tools. These systems recognize identified signatures and anomalies. Methods of detecting security incidents by detecting anomalies in network traffic are described. The proposed methods are focused on the detection of DoS attack in the industrial Modbus protocol and operations performed outside the normal interval in the DNP3 protocol. The functionality of the performed methods is tested in the IDS system Zeek.
Keywords: Modbus, DNP3, Industrial networks, Security incidents.

1 Úvod

V oblasti průmyslových sítí dochází k propojování IT a OT technologií, toto propojení však. To ovšem vystavuje tyto sítě možným bezpečnostním incidentům. Tato oblast také postrádá dostatečné bezpečnostní opatření (ISO/IEC 27001) a implementované bezpečnostní mechanismy nejsou na dostatečné úrovni [1]. Útočník může využít těchto bezpečnostních nedostatků a proniknout do sítě. Dodatečná implementace bezpečnostních opatření by byla velmi nákladná a mnohdy není realizovatelná z důvodu omezeného výkonu používaných zařízení. Dále není reálné provádět aktualizace v intervalech, jak je tomu v IT prostředí [1]. Dodatečné zabezpečení lze provést několika metodami, tento článek je zaměřen na bezpečnostní incidenty v průmyslových sítích s jejich následnou detekcí. Tato oblast se zaměřuje na již známé útoky a anomálie v síťovém provozu s jejich následnou mitigací. Známý útok lze detekovat pomocí signatury, které budou v síťovém provozu vyhledávány a útoky, které nejsou obecně známy, lze detekovat pomocí detekování anomálií v síťovém provozu. Aby bylo možné provádět detekci anomálií, je nutné rozpoznat běžné chování průmyslových sítí, jednotlivé sítě a procesy v nich se liší. Pro detekci bezpečnostních incidentů je tak vhodné skloubení získaných znalostí o chodu sítě a kombinace několika technik, které poskytují dodatečné zabezpečení [1, 2]. Tento článek si dává za cíl vytvoření vhodných metod detekce pro definované signatury a anomálie vyskytující se v průmyslových sítích za pomocí systému IDS (Intrusion Detection System) Zeek a Snort.

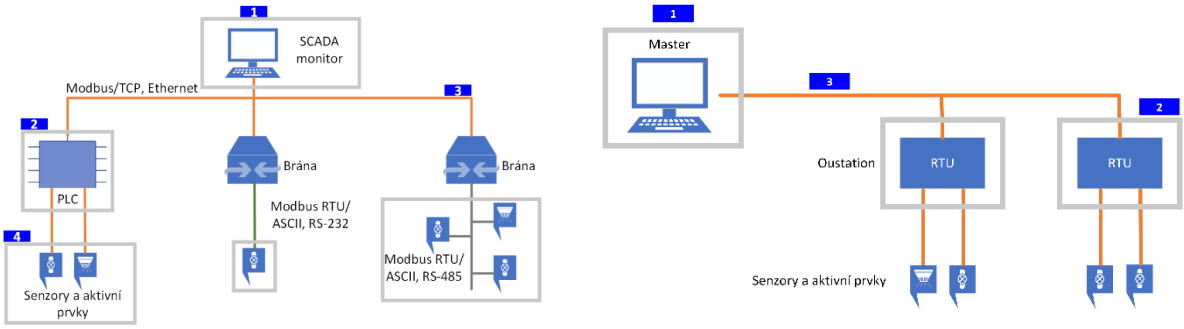
2 Aktuální stav

Mezi hlavní prvky prostředí průmyslových sítí patří PLC (Programmable Logic Controller), které komunikují se senzory a aktivními prvky (např. senzor tlaku, nebo ventil). Ke komunikaci s obsluhou se využívá HMI (Human Machine Interface). K umožnění komunikace mezi jednotlivými prvky průmyslové sítě je využíván průmyslový protokol. Mezi nejznámější protokoly patří Modbus, Profinet, PROFIsafe, DNP3, EtherCAT a další. Tyto protokoly poskytují omezený, nebo žádný stupeň zabezpečení. V rámci průmyslových sítí lze zmínit např. protokoly ModBus [3] a DNP3 [4], které jsou v oblasti energetiky a průmyslu jedny z nejrozšířenějších [5, 6] a na které se tento článek mj. také zaměřuje. Tab. 1 zobrazuje zastoupení jednotlivých protokolů [7]. Jednotlivá data jsou získána na základě datasetu obsahující nejčastěji využívané protokoly, na které byly cíleny útoky. Mezi nejpoužívanější se řadí protokol ICCP (Inter-Control Center Communications Protocol), dále protokol Modbus následovaný protokolem DNP3.

Protokol	Zastoupení protokolu*
ICCP (část IEC 60870-6)	18 %
Modbus	17 %
DNP3	11 %
Red Lion	7 %

Tabulka 1 Zastoupení průmyslových [7].

Každý protokol se liší poskytovanou bezpečností, aby bylo možné navrhnout a implementovat dodatečného zabezpečení je nutné nejprve identifikovat jednotlivé vektory útoku. Obr. 1, část (a) zobrazuje jednotlivé vektory zaměřené na Modbus. Modbus má několik implementací, mezi nejznámější patří Serial Modbus (RTU/ASCII), Modbus/TCP a Modbus over TCP/RTU. Vektory útoku tvoří: HMI (1), řídící prvek PLC (2), přenosové médium (3), senzory a aktivní prvky (4). Obdobně obr. 1, část (b) se zaměřuje na DNP3. Mezi identifikované vektory útoku patří HMI (1), outstation (2), přenosové médium (3) a jednotlivé senzory a aktivní prvky.



(a) (b)

Obrázek 1 Identifikované vektory útoků pro protokol:(a) Modbus; (b) DNP3.

Nejen u protokolu Modbus, ale u velké části protokolů průmyslových sítí není implementována autentizace. K navázání spojení postačuje znalost cílové IP adresy, portu a kódu funkce (k definování prováděné operace). Tyto informace mohou být snadno odposlechnuty a následně zneužity útočníkem, protože není implementováno šifrování přenášených paketů. Důvodem nedostatečného zabezpečení je předpoklad využívání jen v oddělených a vysoce kontrolovaných částech sítě. Dále bylo zamýšleno, že zabezpečení bude implementováno pomocí jiných technik. Samotné protokoly tak často postrádají bezpečnostní mechanismy [8]. Tab. 2 zobrazuje implementované zabezpečení poskytované protokolem Modbus [9] a protokolem DNP3 [10]. Poskytované zabezpečení „běžných" verzí je podobné s mnoha průmyslovými protokoly. Při vzniku těchto protokolů byl předpoklad že budou využívány jen v oddělených sítích s vysokým dohledem. Nové verze těchto protokolů již obsahují bezpečnostní vylepšení a umožňují tak bezpečný provoz.

Protokol	Zabezpečení	Popis
Modbus/TCP	žádné	Postačuje znalost IP adresy a kódu funkce, port 502
Modbus/TCP Security	Minimální požadavky: **Klíčová výměna**: RSA; **Šifrování**: AES 128 CBC; **Integrita**: SHA256	Využívá TLS, port 802 x.509v3 certifikát
DNP3	žádné	Postačuje znalost IP adresy a kódu funkce, port 20000
DNP3 Secure (ver. 6)	**Klíčová výměna**: RSA; **Šifrování**: AEAD-AES-256-GCM (jako TLS 1.2); **Integrita**: Message Authentication Codes (BLAKE2, SHA-3, eliptické křivky)	Využívání certifikátů

Tabulka 2 Implementované zabezpečení v průmyslovém protokolu Modbus a DNP3.

Tab. 3 zobrazuje vybrané útoky, možné dopady útoků a detekovatelnost útoku, na které se tato práce zaměřuje. Tyto útoky byly vybrány z důvodu jejich velkého dopadu ale složitější detekovatelnosti. Oba vybrané útoky se zaměřují na vektor útoku zaměřeného na HMI, popřípadě na přenosové médium. Pro účel detekce není významné, zda útočník napadl a ovládá HMI, nebo byl proveden útok Man In The Middle a zprávy pochází z nevalidního master zařízení. Tyto útoky jsou jedny z nejčastějších útoků [11, 12]. Detekce provedení operace mimo interval je navíc schopna zachytit více druhů útoků, využívá cyklického síťového provozu (plánované operace) a detekuje veškeré chování mimo tuto dobu.

Protokol	Útok	Dopady	Detekovatelnost
Modbus	DoS útok	Odepření služby, pád zařízení	Anomálie
DNP3	Operace mimo interval	Provedení zápisu/čtení, restartu apod.	Anomálie

Tabulka 3 Přehled simulovaných útoků s možností detekce.

3 Návrh metod detekce

3.1 Protokol Modbus

V rámci demonstrace detekce anomálií byl využit IDS systém ZEEK (BRO), který mimo jiné podporuje práci s protokolem Modbus. IDS ZEEK podporuje práci s Modbus protokolem, ale k jeho využití je zapotřebí definovat cestu ke zvoleným skriptům. Je tak nutné upravit soubor local.zeek v adresáři *„/usr/local/zeek/share/zeek/site“*, do kterého je následně přidána cesta k vybranému skriptu *„@load protocols/modbus/trackmemmap“*. Pro detekci anomálií byl vybrán skript „track-memmap.zeek“, který byl doplněn o událost detekce operace Write Single Register (WSR). Na této operaci lze demonstrovat, že realizace DoS (Denial of Service) útoku na tuto operaci, může způsobit odepření služby. Útok DoS na operaci WSR lze provádět zasíláním hodnoty z master zařízení na jeden vybraný registr v cyklu, dokud nedojde u slave zařízení k překročení výpočetních kapacit a odepření služby. Slave zařízení musí při této operaci odpovědět téměř stejnou zprávou, která byla odeslána z master zařízení. Tento útok může být dále prováděn z více master zařízení (DDoS, Distributed DoS), tím dochází k vyčerpání výpočetních kapacit dříve. Útok lze obdobně provádět zápisem hodnoty na jednotlivé registry v cyklu (zápis není prováděn pouze do jednoho vybraného registru). Tento typ útoku je hůře detekovatelný [13]. V rámci experimentálního testování byla zaměřena pozornost na operaci WSR a možnosti detekce zmíněných útoků na slave zařízení. Tato metoda byla vybrána z nízké výpočetní náročnosti na straně útočníka a účinnosti na straně slave zařízení. Navrženou metodu však lze implementovat na jakoukoli operaci. K určení, zda se jedná o nestandardní provoz, je využit odstup (δ) mezi jednotlivými WSR operacemi realizovanými na jeden registr v paměti slave zařízení. K vytvoření rozhodovací prahové hodnoty (Δ_M) je využito několik (x) předcházejících operací, viz rovnice 1. V době "učení" je nutné zajistit kontrolu, zda se v síti nevyskytuje útočník. K vytvoření prahové hodnoty nezáleží na době zpoždění jednotlivých zpráv z důvodu, že zpoždění je v síti přibližně konstantní, ale je zde zacházeno s rozestupem dvou WSR operací vygenerovaných z master zařízení. Jinými slovy dochází k nastavení prahové hodnoty na průměrný časový úsek od přijetí první zprávy do přijetí druhé zprávy. K detekování DoS útoku by postačovalo získat pouze nejnižší naměřený časový odstup (δ). Tato metoda je použitelná i v reálných sítích, za podmínky, že zpoždění bude konstantního charakteru.

$$\Delta_M = \frac{\sum_{n=1}^{x} \delta}{x} \quad [s]. \tag{1}$$

Z každé následující zprávy je získán odstup od předchozí zprávy (δ) a porovnán s Δ_M, viz rovnice 2. Δ_M může být dále korigována pomocí koeficientu k. V případě splnění podmínky je vyhlášen poplach oznamující potenciální DoS útok.

$$\delta < k * \Delta_M \quad [-]. \tag{2}$$

Aby bylo možné detekovat i potenciální DoS útok pomocí operace WSR i na více registrů (zápis je prováděn cyklicky na jednotlivé registry) je porovnáván odstup (δ) jednotlivého registru s předchozím odstupem stejného registru, viz rovnice 3. Pokud jsou tyto hodnoty s určitou odchylkou (r [%]) totožné, je vyhlášen alarm s upozorněním na možný DoS útok. Pokud je však časový odstup jednotlivých operací podobný může být tato rovnice využita k detekování anomálií v síti s rozdílem, že splnění rovnice bude znamenat legitimní provoz.

$$\delta_{k-1} * (1 - r) \leq \delta_k \leq \delta_{k-1} * (1 + r) \quad [-]. \tag{3}$$

3.2 Protokol DNP3

V případě protokolu DNP3 není nutné provádět změnu v souborech IDS Zeek. Podpora tohoto protokolu je nativně podporována. K vytvoření metod detekující anomálie byl vybrán skript „main.zeek“, jedná se o základní práci s DNP3 zprávami. Tento skript byl následně rozšířen o metody, pracující s jednotlivými zprávami a jejich parametry. Jeden z možných útoků zaměřených na protokol DNP3 je odstranění dané zprávy ze sítě. Popřípadě vytvoření a zaslání vlastní zprávy s dotazem na slave (outstation) zařízení. Možným způsobem, jak provádět detekci těchto útoků je zaměření na interval mezi jednotlivými zprávami. Za normálního chodu může být komunikace prováděna v definovaných intervalech a jeho případná změna může být způsobena útočníkem v síti. K definování prahové hodnoty Δ_{DNP3}, která bude považována za běžné chování sítě lze využít vztah definovaný rovnicí 4. Tento vztah využívá několika (x) mezi-rámcových mezer δ, které jsou zprůměrovány. Je tak třeba nejprve provést proces učení.

$$\Delta_{DNP3} = \frac{\sum_{n=1}^{x} \delta}{x} \quad [s]. \tag{4}$$

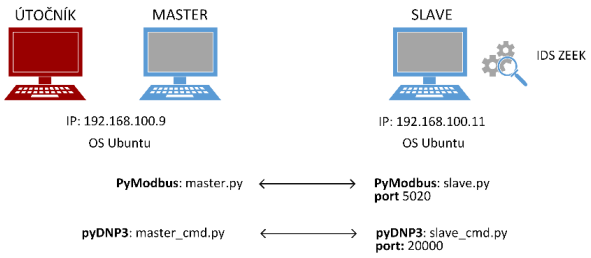
Jakmile je proces učení dokončen, je vždy porovnávána aktuální mezi-rámcová mezera s prahovou hodnotou Δ. V případě, že je momentální mezi-rámcová mezera menší, než prahová hodnota zmenšena pomocí parametru „t“, viz rovnice 5. Obdobně rovnice 6 definuje detekování zprávy, která byla zaslána později, než odpovídá prahové hodnotě zvětšené pomocí parametru u.

$$\delta < \Delta_{DNP3} * (1 - t) \quad [-]. \tag{5}$$

$$\delta \geq \Delta_{DNP3} * (1 + u) \quad [-]. \tag{6}$$

4 Experimentální prostředí

Pro simulaci prostředí průmyslových sítí byla vytvořena virtualizovaná experimentální síť, realizující protokol Modbus pomocí knihovny PyModbus [14]. Tato knihovna poskytuje plnou implementaci protokolu Modbus pomocí programovacího jazyku python. Jako hostovaný OS byl vybrán OS Ubuntu. Pro realizaci protokolu DNP3 (Distributed Network Protocol 3) byla knihovna PyDNP3 [15]. Tato knihovna poskytuje python vazbu na knihovnu opendnp3. Síťové zapojení a vybrané knihovny viz obr. 2. Útočník bude využívat Master stanici (reprezentace napadené stanice) vystupující pod stejnou IP adresou, detekce (IDS) sonda je nasazena na slave zařízení. Vybrané protokoly využívají model master-slave (klient-server). V rámci komunikace Modbus protokolu byl k otestování komunikace proveden příkaz Write Single Register (provedení jednobitové změny v registru slave zařízení), obdobně u protokolu DNP3.



Obrázek 2 Experimentální síťové zapojení.

5 Experimentální výsledky

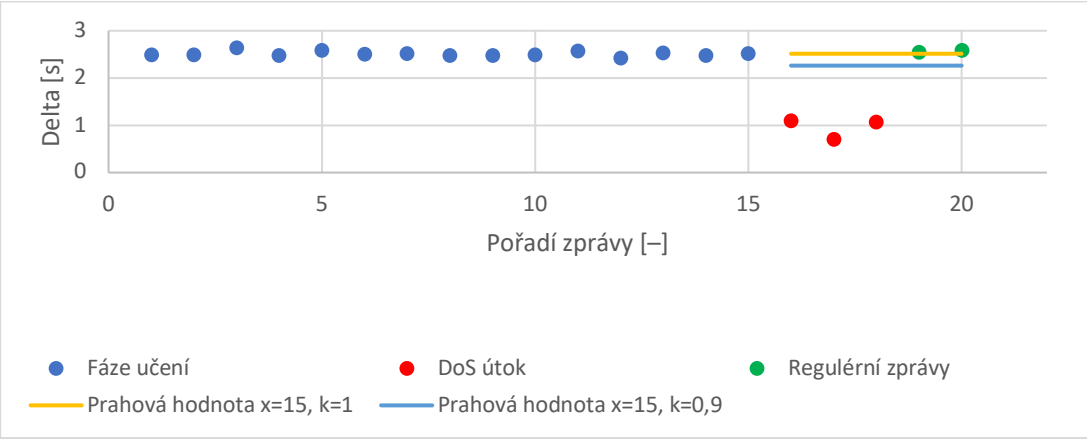
5.1 Protokol Modbus

V rámci vytvořeného skriptu jsou jednotlivé rovnice implementovány. Pokud je některá z podmínek splněna, je proveden záznam v rámci logu, viz výpis. 1. V rámci experimentálního testování byla hodnota x nastavena na hodnotu 15, parametr k na 1 a hodnota r byla nastavena na hodnotu 7 %. Tyto hodnoty byly nastaveny tak, aby byly prahové hodnoty co nejblíže regulérnímu provozu a také, aby nedocházelo ani k falešným alarmům, ani nedetekování prováděného útoku. Nejprve dochází k vytvoření (naučení) prahové hodnoty (Δ) z prvních x (x=10) WSR operací, reprezentováno výpisem „ucim se“. Sloupec „treshold“ zobrazuje vypočítanou hodnotu prahové hodnoty, dokud není hodnota vypočítána je prahová hodnota rovna „0“. Po ukončení fáze „učení“ (do řádku 20 včetně) dochází ke stanovení prahové hodnoty (viz sloupec treshold od řádku 21) a dochází tak k aplikaci rovnice 2, její splnění je reprezentováno výpisem „treshold prekrocen“, který upozorňuje na zápis v neobvyklém intervalu. Zpráva, která nesplňuje ani jednu vytvořenou podmínku je reprezentována textem „zprava prijata“.

1	#path	modbus_register_change						
2	#open	2020-04-28-08-28-09						
3	#fields	id.orig_p	id.resp_p	new_val	Delta	src_addr	dst_addr	text	treshold
4	#types	port	port	count	Interval	addr	addr	string	interval
5		44667	5020	10	-	192.168.100.9	192.168.100.11	Navazano spojeni_	-
6		33775	5020	10	2,488247	192.168.100.9	192.168.100.11	Ucim se	0
7		41443	5020	10	2,493375	192.168.100.9	192.168.100.11	Ucim se	0
8		50385	5020	10	2,636994	192.168.100.9	192.168.100.11	Ucim se	0
9		53723	5020	10	2,482671	192.168.100.9	192.168.100.11	Ucim se	0
10		39281	5020	10	2,582492	192.168.100.9	192.168.100.11	Ucim se	0
11		54969	5020	10	2,500161	192.168.100.9	192.168.100.11	Ucim se	0
12		42703	5020	10	2,521266	192.168.100.9	192.168.100.11	Ucim se	0
13		57003	5020	10	2,482172	192.168.100.9	192.168.100.11	Ucim se	0
14		40683	5020	10	2,482638	192.168.100.9	192.168.100.11	Ucim se	0
15		39729	5020	10	2,499380	192.168.100.9	192.168.100.11	Ucim se	0
16		42875	5020	10	2,577473	192.168.100.9	192.168.100.11	Ucim se	0
17		55365	5020	10	2,422150	192.168.100.9	192.168.100.11	Ucim se	0
18		41127	5020	10	2,538051	192.168.100.9	192.168.100.11	Ucim se	0
19		34571	5020	10	2,478753	192.168.100.9	192.168.100.11	Ucim se	0
20		40553	5020	10	2,513979	192.168.100.9	192.168.100.11	Ucim se	0
21		50279	5020	10	1,095429	192.168.100.9	192.168.100.11	TRESHOLD PREKROCEN	2,51332
22		33729	5020	10	0,706493	192.168.100.9	192.168.100.11	TRESHOLD PREKROCEN	2,51332
23		48083	5020	10	1,067000	192.168.100.9	192.168.100.11	TRESHOLD PREKROCEN	2,51332
24		35173	5020	10	2,547971	192.168.100.9	192.168.100.11	Zprava prijata	2,51332
25		45167	5020	10	2,594318	192.168.100.9	192.168.100.11	Zprava prijata	2,51332

Výpis 1: Vygenerovaný log.

Podobné nastavení lze provést i pro detekování DoS útoků v případě, že cílové registry nejsou voleny v cyklu, ale jsou voleny náhodně. Ke změně postačuje provádět detekci bez ohledu na jednotlivý registr. Vizualizovaný log, viz obr. 3.

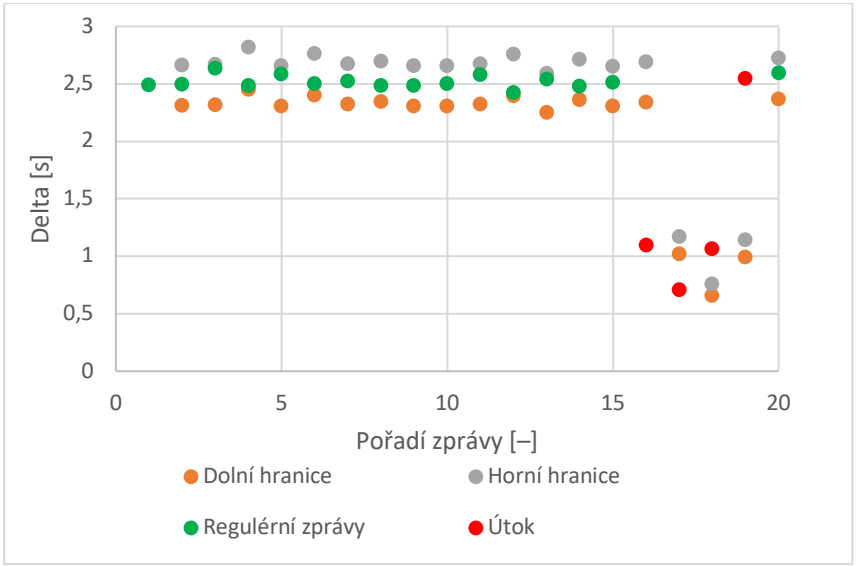


Obrázek 3 Vizualizace logu, protokol Modbus, DoS útok (rovnice 1 a 2).

Modře vyznačené body definují fázi učení. Tato fáze trvá x = 15 zpráv, po této době dochází k eliminování zpoždění mimořádného charakteru. V případě, že bude hodnota δ podobná není nutné nastavovat hodnotu x na vyšší hodnoty. Následně je vytvořena Δ_M, která je využívána v rovnici 2 uplatňována po x zprávách pro všechny následující zprávy. Aby nebyla nastavená kritéria příliš striktní a nedocházelo k falešným alarmům, lze volit parametr k v rovnici 2, který ovlivňuje hladinu, pod kterou bude zpráva ohlášena jako útok. V experimentálním testování byl parametr k nastaven na hodnotu 1, tedy maximálně striktní, což se nejeví, jako vhodné. Vhodnější je nastavení parametru k na hodnotu 0,9 (na obr. vyneseny obě varianty).

Obr. 4 zobrazuje využití rovnice 3 na totožná data. Z důvodu, že hodnoty δ jsou si velice podobné, lze tuto rovnici využít tak, že při jejím splnění bude zpráva považována za legitimní. Tato metoda nevyžaduje fázi učení. Pro každou zprávu je na základě hodnoty δ předchozí zprávy vypočtena dle rovnice 3 dolní a horní hranice. V ideálním případě nebude tato hranice překročena, její překročení vyvolává alarm.

Parametr r zde byl nastaven na hodnotu 7 %. Při vyšších hodnotách dochází k přílišnému „rozevření“ legitimní oblasti. Při nižších hodnotách dochází ke zmenšení legitimní oblasti a dochází tak k většímu počtu falešných alarmů.



Obrázek 4 Vizualizace logu, protokol Modbus, DoS útok (rovnice 3).

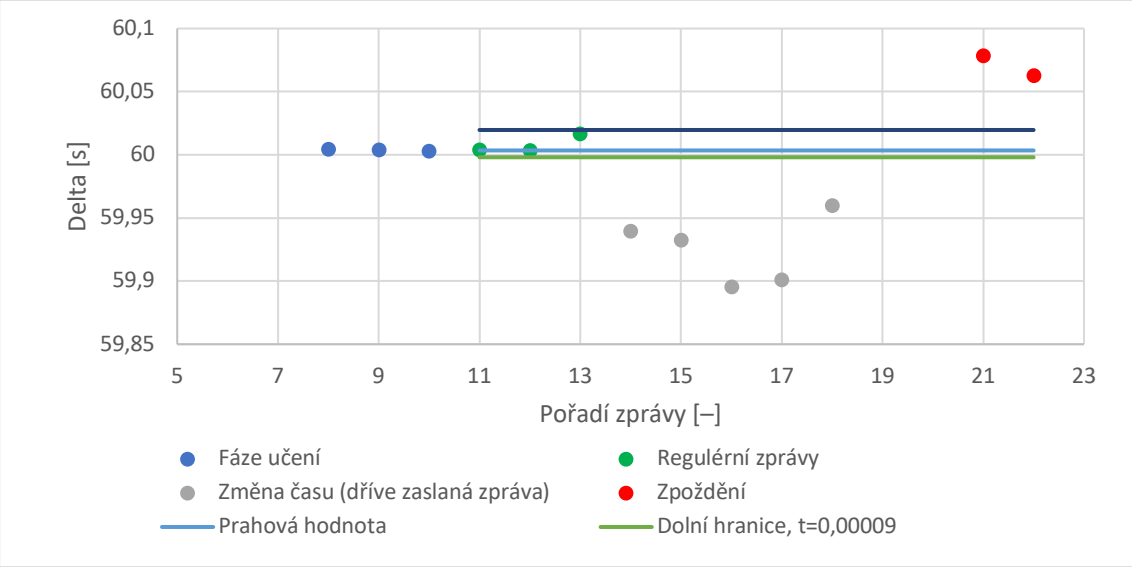
5.2 Protokol DNP3

Pro implementaci definovaných rovnic bylo nutné skript „main“ rozšířit o metodu „dnp3-application-request-header“. Tento skript detekuje jednotlivé zprávy a vytváří záznam jednotlivých operací a intervalu mezi nimi. Na základě intervalů je vytvořena prahová hodnota Δ z rovnice 4, kde byl parametr „x“ nastaven na hodnotu 3. Parametr „t“ z rovnice 5 byl v rámci experimentálního testování nastaven na hodnotu 0,00009 a parametr „u“ z rovnice 6 byl z experimentálních důvodů nastaven na hodnotu 0,00027. Tyto hodnoty vychází z experimentálního testování, kde hodnota 3 pro parametr x je dostatečně dlouhá k nastavení prahové hodnoty. Hodnoty parametru „t“ a „u“ jsou nastaveny tak, aby vytvořená oblast, ve které jsou zprávy přijaty bez alarmu, byla co nejmenší, ale nedocházelo k nerozpoznání útoku, ani k falešným alarmům. Z důvodu, že po spuštění master skriptu dochází k vygenerování zpráv, jejichž mezi-rámcová mezera δ se od ostatních zpráv významně liší, jsou první 4 zprávy vynechány. K naučení prahové hodnoty Δ tak dochází až po zaslané sedmé zprávě „READ“ (kód funkce 1), v tomto případě 60,00346 (modře zvýrazněno), viz výpis 2. Momentální odstup δ je uveden ve sloupci odstup, prahová hodnota Δ je zobrazena (po jejím vypočítání, tedy 7. zprávě) ve sloupci prumerny_odstup. Na řádku 18–22 je formou výpisu textu „Zmena casu“ (červeně zvýrazněno) upozorněno na provedení operace pod definovanou prahovou hodnotu Δ, byla tedy splněna rovnice 5. Zpoždění zprávy oproti běžnému chování, to je indikováno textem „Zpozdeni!“ řádek 23 a 24.

1	#path		dnp3						
2	#open			2020-03-25-05-24-26					
3	#fields	poradi	odstup	prumerny_odstup	Func	popisek	ip_zdroj	ip_cil	fc_request
4	#types	count	interval	interval	Count	string	addr	addr	string
5		1	0	0	21	Prijato	#addr1	#addr2	DISABLE_UNSOLICITED
6		2	0	0	0	Prijato	#addr1	#addr2	CONFIRM
7		3	0	0	1	Prijato	#addr1	#addr2	READ
8		4	0	0	20	Prijato	#addr1	#addr2	ENABLE_UNSOLICITED
9		5	0,052331	0	1	Prijato	#addr1	#addr2	READ
10		6	0,054173	0	1	Prijato	#addr1	#addr2	READ
11		7	60,016950	0	1	Prijato	#addr1	#addr2	READ
12		8	60,004150	0	1	Prijato	#addr1	#addr2	READ
13		9	60,003940	0	1	Prijato	#addr1	#addr2	READ
14		10	60,002820	0	1	Prijato	#addr1	#addr2	READ
15		11	60,003630	60,00346	1	Prijato	#addr1	#addr2	READ
16		12	60,003460	60,00346	1	Prijato	#addr1	#addr2	READ
17		13	60,016410	60,00346	1	Prijato	#addr1	#addr2	READ
18		14	59,939231	60,00346	1	Zmena casu!	#addr1	#addr2	READ
19		15	59,932317	60,00346	1	Zmena casu!	#addr1	#addr2	READ
20		16	59,895530	60,00346	1	Zmena casu!	#addr1	#addr2	READ
21		17	59,900808	60,00346	1	Zmena casu!	#addr1	#addr2	READ
22		18	59,959835	60,00346	1	Zmena casu!	#addr1	#addr2	READ
23		21	60,078300	60,00346	1	Zpozdeni!	#addr1	#addr2	READ
24		22	60,062700	60,00346	1	Zpozdeni!	#addr1	#addr2	READ

Výpis 2: Log, detekování operace mimo interval, protokol DNP3.

Obr. 5 zobrazuje grafickou reprezentaci získaného logu. Během testování byla prahová hodnota sestavena na základě x = 3 zpráv, velikost tohoto parametru se z důvodu jen velmi malých odchylek zdá v tomto případě dostatečná. Po těchto x zprávách dochází k výpočtu horní a dolní hranice, které jsou během následující komunikace neměnné. Nastavení parametrů t a u závisí na požadovaném nastavení, v tomto případě se jeví nastavení parametrů, jako dostatečné. I přes takto nízké parametry, resp. okno přijetí zpráv, jsou veškeré zprávy stále bezchybně rozpoznány.



Obrázek 5 Vizualizace logu, detekce operace mimo interval, protokol DNP3 (rovnice 4, 5, 6).

6 Diskuze

V experimentálním testování byly využity jak metody detekce bezpečnostních incidentů, které vyžadují předcházející fázi „učení“ tak ty, které ji nevyžadují. Největším rizikem metod založených na učení je možný výskyt útočníka a zejména jeho zásah do komunikace během této fáze. Může tak dojít k „naučení“ nežádoucích hodnot, které nebudou schopné útok detekovat. Tato fáze může být i vynucena pomocí restartu zařízení. U metod, které nevyžadují „učení“ je nutné provádět korekci pomocí parametrů. Je důležité tyto parametry nastavit tak, aby nedocházelo k falešným alarmům, ale ani k případům, kdy útok není detekovatelný. Práce [16] přistupuje k detekování anomálií v síti pomocí nasazení neuronových sítí, které využívají veškeré dostupné informace k detekování anomálie. Práce [17] pracuje s dvěma metodami detekce, pomocí času příchodu a srovnávání mezi jednotlivými toky dat. V článku [17] detekují mj. opakované anomálie s uspokojivými výsledky při délce trvání více než 1 hodinu, nebo v případě, že provoz má nízkou četnost událostí. Oproti tomu, metody navržené v tomto článku jsou schopny detekovat i velice ojedinělé události za předpokladu adekvátně nastavených hodnot, a pokud byl během fáze „učení“ naučen síťový provoz odpovídající běžnému chování sítě. Nejlepších výsledků je dosahováno v případě detekce anomálií u operací s cyklického charakteru. Po vygenerování je nutné vhodné zacházení s alarmy a jejich distribuce do jiných systémů. V případě, že metody pouze útok detekují, ale neprovádí žádné protiopatření, může v krátké době dojít k uvedení zařízení mimo provoz. Je tak nutné provádět distribuci alarmů do jiných systémů, které jsou schopné pokračování útoku zabránit, nebo alespoň bezpečně uložit získané informace o útoku.

7 Závěr

Z experimentálního testování bylo zjištěno, že k rozpoznávání anomálií nejen v oblasti průmyslových sítí je vhodné využívat různé nástroje, v tomto případě IDS systém ZEEK. Pro účely detekce anomálií v síti je nutné nejprve vycházet z vybraného průmyslového protokolu. Záleží tak, jaký průmyslový protokol je v síti nasazen a jakým způsobem probíhá běžná komunikace. V případě že komunikace probíhá v definovaných cyklech, je vhodné provádět detekci zpráv, které se těmto cyklům vymykají. Naopak v případě „nahodilého“ chování je nutné hledat vzniklé vzory, tedy například cykly, které neodpovídají standardu. K zacházení s provozem a rozpoznávání anomálií se využívání odstupů jednotlivých zpráv jeví, jako vhodná varianta. Je však vhodné využívat i dalších parametrů, které jsou ze získané komunikace k dispozici. V provedeném experimentálním testování byl nasazen IDS systém ZEEK. Pomocí tohoto systému byla provedena detekce útoku DoS pomocí odstupu jednotlivých WSR operací na protokolu Modbus. V případě nalezení opakujícího se chování (v případě běžně), popřípadě poklesnutí průměrného odstupu mezi jednotlivými zprávami pod vytvořenou prahovou hodnotu je provedeno upozornění na neobvyklé chování. U protokolu DNP3 bylo za využití mezi-rámcových mezer v IDS systému ZEEK detekováno chování, které neodpovídá cyklickému chování komunikace. Pokud je vytvořená prahová hodnota běžného intervalu překročena v definovaném intervalu, je na toto chování upozorněno. Výsledky ukázali, že využívání systémů IDS/IPS je vhodné v prostředí průmyslových sítí. Tyto systémy umožňují zaznamenávat velké množství parametrů komunikace. Mezi významné parametry komunikace patří IP adresy, definovaný kód funkce (operace, která se má provést), odstup jednotlivých zpráv a jiné parametry. Tyto parametry lze využít pro detekci chování, které se vymyká běžnému chování stavu sítě v definovanou dobu. Jednou z dalších možností, jak provádět detekci anomálií nejen v oblasti průmyslových sítí je nasazení strojového učení, které by bylo nasazeno na komunikaci s cílem odhalit provozní síťové anomálie.

Poděkování

Tento článek byl vytvořen za podpory Národního programu udržitelnosti v rámci grantu LO1401 a Ministerstva průmyslu a obchodu (MPO) v rámci projektu č. FV40366. Pro výzkum byla využita infrastruktura vědecko-výzkumného centra SIX.

Literatura

[1] CALDWELL, Tracey. Plugging IT/OT vulnerabilities – part 2. *Network Security* [online]. 2018, **2018**(9), 10-15 [cit. 2020-05-27]. DOI: 10.1016/S1353-4858(18)30089-8. ISSN 13534858

[2] FOVINO, Igor Nai, Marcelo MASERA, Luca GUIDI a Giorgio CARPI. An experimental platform for assessing SCADA vulnerabilities and countermeasures in power plants. *3rd International Conference on Human System Interaction* [online]. IEEE, 2010, 2010, , 679-686 [cit. 2020-05-27]. DOI: 10.1109/HSI.2010.5514494. ISBN 978-1-4244-7560-5.

[3] *MODBUS APPLICATION PROTOCOL SPECIFICATIONV1.1a* [online]. 2004 [cit. 2020-05-27]. Dostupné z: http://www.modbus.org/ docs/Modbus_Application_Protocol_V1_1a.pdf

[4] IEEE Standard for Electric Power Systems Communications-Distributed Network Protocol (DNP3)," in IEEE Std 1815-2012 (Revision of IEEE Std 1815-2010) , 10 Oct. 2012, doi: 10.1109/IEEESTD.2012.6327578.

[5] AMOAH, Raphael, Seyit CAMTEPE a Ernest FOO. Securing DNP3 Broadcast Communications in SCADA Systems. IEEE Transactions on Industrial Informatics [online]. 2016, 12(4), 1474-1485 [cit. 2020-05-27]. DOI: 10.1109/TII.2016.2587883. ISSN 1551-3203. Dostupné z: http://ieeexplore.ieee.org/document/7506334/

[6] CREMERS, Cas, Martin DEHNEL-WILD a Kevin MILNER. Secure authentication in the grid: A formal analysis of DNP3 SAv5. *Journal of Computer Security* [online]. 2019, **27**(2), 203-232 [cit. 2020-05-27]. DOI: 10.3233/JCS-181139. ISSN 18758924.

[7] BOU-HARB, Elias, Nasir GHANI, Abdelkarim ERRADI a Khaled SHABAN. Passive inference of attacks on CPS communication protocols. *Journal of Information Security and Applications* [online]. 2018, **43**, 110-122 [cit. 2020-05-27]. DOI: 10.1016/j.jisa.2018.10.002. ISSN 22142126.

[8] COLLANTES, Miguel Herrero a Antonio López PADILLA. Protocols and Network Security in ICS infrastructures. *The Spanish National Institute for Cybersecurity* [online]. 2015 [cit. 2020-05-27]. Dostupné z: https://www.incibe-cert.es/sites/default/files/contenidos/guias/doc/incibe_protocol_net_security_ics.pdf

[9] MODBUS/TCP Security: Protocol Specification. Modbus.org [online]. 2018 [cit. 2020-04-29]. Dostupné z: http://modbus.org/docs/MB-TCP-Security-v21_2018-07-24.pdf

[10] Overview of DNP3 Security Version 6. Dnp.org [online]. 2020 [cit. 2020-04-29]. Dostupné z: https://www.dnp.org/LinkClick.aspx?fileticket=hyvYMYugaQI%3d&tabid=66&portalid=0&mid=447&forcedownload=true

[11] YLMAZ, Ercan Nurcan, Bunyamin CIYLAN, Serkan GONEN, Erhan SINDIREN a Gokce KARACAYILMAZ. Cyber security in industrial control systems: Analysis of DoS attacks against PLCs and the insider effect. *2018 6th International Istanbul Smart Grids and Cities Congress and Fair (ICSG)* [online]. IEEE, 2018, 2018, , 81-85 [cit. 2020-05-27]. DOI: 10.1109/SGCF.2018.8408947. ISBN 978-1-5386-4478-2.

[12] BHAMARE, Deval, Maede ZOLANVARI, Aiman ERBAD, Raj JAIN, Khaled KHAN a Nader MESKIN. *Cybersecurity for industrial control systems: A survey* [online]. 2020, **89** [cit. 2020-05-27]. DOI: 10.1016/j.cose.2019.101677. ISSN 01674048

[13] BHATIA, Sajal, Nishchal KUSH, Chris DJAMALUDIN, James AKANDE a Ernest FOO. Practical Modbus Flooding Attack and Detection. *Australasian Information Security Conference (AISC 2014)* [online]. [cit. 2020-05-27]. Dostupné z: https://dl.acm.org/doi/pdf/10.5555/2667510.2667517

[14] Pymodbus 2.3.0: A fully featured modbus protocol stack in python. Pypi.org [online]. [cit. 2020-04-28]. Dostupné z: https://pypi.org/project/pymodbus/

[15] Pydnp3: Python bindings for opendnp3 libary. Github.com [online]. [cit. 2020-04-28]. Dostupné z: https://github.com/ChargePoint/pydnp3,

[16] REUTER, Lenhard, Oliver JUNG a Julian MAGIN. Neural network based anomaly detection for SCADA systems. *2020 23rd Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN)* [online]. IEEE, 2020, 2020, , 194-201 [cit. 2020-05-27]. DOI: 10.1109/ICIN48450.2020.9059436. ISBN 978-1-7281-5127-4.

[17] CHIH-YUAN, Lin a Simin NADJM-TEHRANI. Timing Patterns and Correlations in Spontaneous {SCADA} Traffic for Anomaly Detection. *22nd International Symposium on Research in Attacks, Intrusions and Defenses ({RAID} 2019)*. 2019, s. 73–88. ISBN 978-1-939133-07-6.