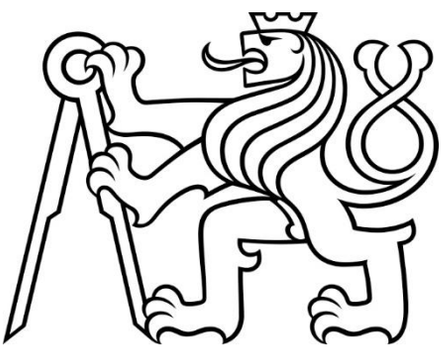
ČESKÉ VYSOKÉ
UČENÍ TECHNICKÉ
V PRAZE

FAKULTA
STROJNÍ



BAKALÁŘSKÁ PRÁCE

2019

TIMOTEJ
VRÁTNÝ

České vysoké učení technické v Praze

Fakulta strojní

Ústav přístrojové a řídící techniky

Obor: Informační a automatizační technika

Využití standardu OPC UA v průmyslové komunikaci

BAKALÁŘSKÁ PRÁCE

Vypracoval: Timotej Vrátný

Vedoucí práce: Ing. Mgr. Jura Jakub Ph.D.

Rok: 2019

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Vrátný** Jméno: **Timotej** Osobní číslo: **456164**

Fakulta/ústav: **Fakulta strojní**

Zadávající katedra/ústav: **Ústav přístrojové a řídicí techniky**

Studijní program: **Strojírenství**

Studijní obor: **Informační a automatizační technika**

II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

Využití standardu OPC UA v průmyslové komunikaci.

Název bakalářské práce anglicky:

Use of the OPC UA standard in industrial communications.

Pokyny pro vypracování:

1) stručně popsat protokol OPC UA
2) vyzkoušet protokol OPC UA na komunikaci PLC - SCADA
3) zhodnotit a otestovat možnosti šifrované komunikace

Seznam doporučené literatury:

https://opcfoundation.org/about/opc-technologies/opc-ua/

Jméno a pracoviště vedoucí(ho) bakalářské práce:

Ing. Mgr. Jakub Jura, Ph.D., U12110.3

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) bakalářské práce:

Datum zadání bakalářské práce: **26.04.2019** Termín odevzdání bakalářské práce: **12.06.2019**

Platnost zadání bakalářské práce: _____

Ing. Mgr. Jakub Jura, Ph.D. podpis vedoucí(ho) práce	podpis vedoucí(ho) ústavu/katedry	prof. Ing. Michael Valášek, DrSc. podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Student bere na vědomí, že je povinen vypracovat bakalářskou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v bakalářské práci.

26-04-2019

Datum převzetí zadání

Podpis studenta

Prohlášení

Prohlašuji, že jsem svou bakalářskou práci vypracoval samostatně a použil jsem pouze podklady (literatura, projekty, SW atd.) uvedené v této práci.

V Humpolci dne 10.6.2019 .....

Timotej Vrátný

Poděkování

Děkuji Ing. Mgr. Jakubu Jurovi Ph.D. za jeho konzultace při tvorbě této bakalářské práce, za jeho podněty a připomínky, které ji obohatily. Dále jak je dobrým zvykem, děkuji své rodině a přátelům, kteří mi poskytli nezbytnou podporu během celého studia.

Timotej Vrátný

Název práce:

Využití Standardu OPC UA v průmyslové komunikaci

Autor: Timotej Vrátný

Obor: Informační a automatizační technika

Druh práce: Bakalářská práce

Vedoucí práce: Ing. Mgr. Jura Jakub Ph.D.

Ústav přístrojové a řídící techniky

České vysoké učení technické v Praze

Abstrakt: Tato bakalářská práce je zaměřená na seznámení čtenáře s průmyslovým standardem OPC UA. Zároveň v ní je zahrnuta praktická demonstrace komunikace mezi PLC S7-1500 a SW myScada. Zaměřuje se také na bezpečnostní prvky tohoto protokolu.

Klíčová slova: OPC UA, komunikace, Scada, protokol, standard, automatizace

Title:

Use of the OPC UA standard in industrial communication

Author: Timotej Vratny

Abstract: This bachelor's thesis demonstrates connection between PLC and SW – my Scada by standard OPC UA. In the practical part of this work there is included an example of the safety communication between Scada-PLC by OPC UA. This Standard should help extended industrial 4.0. Now you can easily connect all components from diffrent distributors wich supports OPC UA.

Key words: OPC UA, communication, Scada, protocol, standard, automation

OBSAH

ÚVOD

Jedním z největších požadavků aktuální doby ať už v průmyslu nebo marketingu případně dalších odvětví je sběr dat. Kdo správně dokáže analyzovat data, jednak trhu, tak své vlastní výroby je většinou o krok napřed před konkurencí. Jedním z klíčových faktorů při sbírání těchto informací je komunikace v dané sféře. Dodržet stejný komunikační protokol je obtížné již na úrovni jednoho výrobního závodu, v situaci, kdy se jedná o komunikaci na úrovni koncernu se obtíže násobí, ale tato jednota a účelnost komunikace je klíčová pro vývoj koncernu a tržní pozici značek, které koncern produkuje. Pokud tyto data budou nepřesná mohou hrát významnou roli v dalším vývoji firmy nebo jejím zániku. Před nástupem internacionalizace výroby nebyl problém až tak velký, ale s nástupem globalizace a rozprostření výroby na více kontinentů, je potřeba jednoty komunikace základní podmínkou úspěšnosti. V nynější době globalizace a extrémní automatizace se jednotlivé podniky nacházejí i více jak na jednom kontinentu. To vše vytváří tlak ze strany výrobních podniků především v automobilovém průmyslu na jednotný standard komunikace, který vyřeší problém na veškerých platformách. Ne každý chce využívat veškerý Software jen od společnosti Windows nebo Linux, někdy to ani není možné kvůli předchozím smlouvám. V podstatě není možné aplikovat veškeré řídící komponenty (PLC atd.) od jediného výrobce. Většinu těchto problémů pomohla vyřešit komunikace OPC UA. Standard OPC UA usnadnil, a hlavně sjednotil komunikaci mezi jednotlivými zařízeními od různých výrobců. Stačí když zařízení podporuje OPC UA. Momentálně se jedná o nejžádanější komunikační standard v průmyslu. Aktuálně se nejeví pravděpodobné, že by tento protokol mohl nějaký jiný zcela nahradit, spíše je pravděpodobný další vývoj dle potřeb uživatelů.

Cílem této bakalářské práce je seznámit se standardem OPC UA v průmyslové komunikaci. Nejprve stručně popíši z jakých protokolů se OPC UA vyvíjel, dále se budu věnovat samotnému standardu OPC UA, větší pozornost budu věnovat bezpečnosti této komunikace. Většina uživatelů pouze ví, že mu systémy komunikují přes tento protokol, ale už neví, co všechno se zatím skrývá a jaké má možnosti. Dalším bodem je demonstrace protokolu OPC UA mezi PLC a Scadou, tudíž tato práce může někomu posloužit jako stručný návod k nastavení PLC, tak aby bylo schopné komunikovat přes OPC UA s další platformou. Já jsem zvolil PLC S7-1500, také použiji TIA Portál V 15.1 od společnosti Siemens a program mydesigner 8 od firmy mySCADA Technologies s.r.o.

Bližší popis použitého hardwaru a softwaru bude v praktické části této práce. Posledním bodem bude zhodnocení a testování jednotlivých možností šifrované komunikace. Zabezpečení komunikace v průmyslovém podniku je dalším z hlavních aspektů vzniku OPC UA. Nedílnou součástí každodenní výroby je také průmyslová špionáž či pouze touha po zničení konkurenčního podniku. Předešlé standardy tento aspekt poněkud pomíjely.

1 VÝVOJ OPC

OPC (OLE for Process Control) byl původně navržen několika odborníky (později známé jako sdružení OPC foundation), kteří se zabývali automatizací společně s firmou Microsoft v roce 1995. V posledních deseti letech se stala nejvíce používanou univerzální cestou ke komunikaci v automatizačním odvětví veškerého průmyslu. Za roky vývoje, se tento standard dostal od jednoduchého (DA) neboli Data access, který definuje získávání aktuálních dát dále (AE) Alarms & events, ty definují rozhraní pro události jako jsou krizová hlášení až po více složité interakce jako je (HDA) Historical data access, který popisuje přístup k archivovaným údajům a již má poměrně rozsáhlou funkcionalitu a dosah. Vždy se objeví nějaké nové mezery, které nepokryjí požadavky stále složitějších řídících systémů, a proto se každým rokem vyvíjí novější verze tohoto standardu. Veškeré dosavadní problémy byly prozatím pokryty nezávislou platformou OPC UA standard, o kterém pojednává má práce.[1]

1.1 OPC Foundation

Myslím si, že za zmínku také stojí sdružení OPC foundation bez kterých by pravděpodobně tento komunikační standard vůbec nevznikl. Tato organizace byla založena roku 1994, spojením sil pěti velkých prodejců v automatizačním odvětví (Fisher-Rosemount, Rockwell Software, Opto 22, Intellution, a Intuitive Technology) s účelem vytvoření základního kamene OLE pro procesní řízení. Tento produkt byl představen a nabídnut k použití pod názvem OPC standard roku 1996. OPC foundation byla nadále pověřena ve vývoji a měla se zejména zaměřit na interoperabilitu tzn. schopnost různých systémů vzájemně spolupracovat. Její primární úkol byl zahrnout do tohoto vývoje co nejvíce výrobců a uživatelů přístrojů, PLC, Softwarů a podnikových systémů. Tato nadace spolupracuje například s organizací MTConnect, která se podílela na podobné misi [2]. Momentálně je součástí této nadace více než 4200 dodavatelů, kteří vytvořili více než 35000 různých OPC produktů, které jsou využity ve více než 17 milionech aplikací. Odhaduje se, že nadace svými produkty ušetřila zdroje v hodnotě miliard dolarů.[3]

1.2 OPC

OPC zastává OLE pro procesní řízení (OLE for Process Control), očividně vychází z komunity Microsoft charakterizována OLE a COM/DCOM technologií. OPC je založená na komunikaci Klient/Server což znamená, že máme jeden nebo dva servery, které čekají na požadavek od několika různých klientů. Jakmile server obdrží požadavek, odpoví na něj a poté se vrátí do statusu čekání. Klient také může dát instrukce serveru, aby mu zaslal veškeré aktualizace, které se na serveru udály. V OPC je to klient kdo rozhoduje o tom kdy a jaká data se budou načítat ze základního systému.[1]

1.3 OPC protokoly

Rozdílné OPC protokoly jsou kompletně soběstačné a nejsou na sobě závislé. V modelu OPC se nachází následující protokoly: DA (Data access), AE (Alarm & Events), HDA (Historical Data Access), XML DA (XML Data Access) a poslední DX (Data eXchange). Každý z těchto zmíněných protokolů výše má svoje vlastní čtení a zapisování dat. Tudíž příkazy ovlivňují pouze jeden protokol v daném čase. Funguje to také v případě, kdy pouze jeden OPC Server podporuje několik protokolů. Nejvíce používaným a zároveň nejstarším protokolem je data access (DA).[4]

1.3.1 OPC Data Access

Jedním ze základních protokolů je OPC DA, tento protokol získává data z řídícího systému a předává je do dalších systémů provozu. Součástí je každá informace o konkrétním datovém bodu, který obsahuje požadovanou informaci. Nejprve máme samotná data ty nazýváme hodnotou a samozřejmě jejich názvem. S tímto souvisí další množství jednotlivých bodů, které popisují informaci. Prvním z nich je tzv. časové razítko, jenž udává přesný čas, kdy byla hodnota přečtena. Poslední částí je kvalita. Jednoduše sděluje základní porozumění, zda jsou data platná či nikoliv.[1]

1.3.2 OPC Alarm & Events

Jako druhý protokol byl přidán OPC AE. Tento protokol je zásadně odlišný od DA, jednoduše kvůli faktu, že probíhající událost nemá současnou hodnotu. Což znamená, že tento protokol je vždy primární službou, kde klient obdrží veškeré události, které na tento protokol působí.

Také obsahuje časové razítko, ale na rozdíl od DA není zde žádné uložiště na serveru a jakmile je událost zpracována, server na ní zapomene, jako kdyby tam nikdy nebyla.

1.3.3 OPC Data Historical Access

OPC data access (OPC HDA) povoluje přístup v reálném čase kontinuálně měnícím datům. Zatímco OPC historical access poskytuje přístup, k již uloženým datům může jich zpětně vyvolat prakticky neomezené množství. Protokol je určen pro podporu dlouhodobě se ukládajících souborů s jedním nebo více datovými body. Byl vytvořen k vytažení a rozeslání starších nebo historických dat uložených ve SCADA systémech nebo jim podobným. Nyní se již moc nepoužívá, v porovnání s OPC UA je velmi zastaralý.[5]

1.3.4 OPC XML-DA

OPC XML-DA byla první platformou, nezávislou specifikací OPC, která nahradila COM/DCOM s HTTP/SOAP a technologie webových služeb. OPC XML-DA byl vytvořen pro přístup k internetu a sjednocení komunikace v podniku. Na základě nezávislosti této platformy, byl převážně implementován ve vestavěných systémech a v jiných než Microsoft platformách. Kvůli své vysoké náročnosti na zdroje a omezenému výkonu nebyl tento protokol tak úspěšný, tím nenaplnil naděje původně do něj vkládané.[5]

1.3.5 OPC classic

OPC classic pokračoval jako nedílná část z technologického portfolia OPC. Je založen na technologii od Microsoft Windows COM/DCOM. OPC classic poskytuje oddělenou specifikaci pro výměnu procesních dat, alarmů a historických dat. Primární specifikací OPC classic je OPC DA, který charakterizuje rozhraní mezi klientem a serverem. Dalšími významnými specifikacemi jsou OPC AE a OPC HDA. OPC classic. Veškeré protokoly popisuji v dalším textu. Nakonec byl ještě vylepšen na verzi OPC. NET 4.0, aby mohl konkurovat technologickým inovacím z Microsoft platforem s poskytnutím lepšího připojení, vyšší úrovně zabezpečení, spolehlivosti a interoperability. Z čehož nakonec OPC UA vychází. [6]

2 CO JE TO OPC UA?

OPC UA – Tato zkratka se postupně vyvinula a změnil se její význam, skládá se ze dvou částí a to OPC-Open Platform Communications a UA-Unified Architecture. První část můžeme volně přeložit jako otevřená komunikační platforma a druhou část sjednocená architektura. Tato technologie je poměrně nový komunikační protokol, tím nový se myslí, že až v posledních letech se začíná více rozšiřovat do průmyslu a je vyžadován na jednotlivých zařízeních, tzv. machine to machine protokol, byl vyvinut a poté vypuštěn do světa společností OPC foundation v roce 2008. Jedná se o nezávisle orientovanou platformu, která funkčními prvky vychází z OPC classic a byla sjednocena do jedné rozšířené verze, jednotlivými částmi vývoje této komunikace (z čeho vychází) se budeme zabývat v dalších bodech mé bakalářské práce. [6]

Hlavním rozdílem mezi původní specifikací OPC, která je založena na technologii COM/DCOM tedy pracuje pouze pod OS Windows, je že OPC UA může pracovat na veškerých platformách ne pouze na OS Windows například, macOS, Android, jakákoliv verze od Linuxu, protože OPC UA je založena na komunikačních standardech, které jsou nejpoužívanější např. TCP/IP, HTTP atd. Jedním z hlavních cílů OPC UA je rozšířit jej do co nejvíce možných zařízení, přičemž se nejvíce myslí na PLC automaty, případně frekvenční měniče, nebo tradiční stolní počítače či další komponenty, které jsou nedílnou součástí každodenní výroby v průmyslovém podniku. Dále může být zabudován i jinde než v průmyslové výrobě i když je to jeho primární využití např. v chytrých telefonech, senzorech, vestavěných systémech atd.

Hlavními požadavky na novou komunikaci byly:

- Zabezpečení

- Nezávislost na platformě

- Funkční vyváženost

- Rozšiřitelnost

- Nastavitelné časové limity pro každou službu

[6, 7]

Primárním úkolem standardu OPC UA je specifikovat a ukládat data mezi jednotlivými komponenty na variabilních platformách. Jednotlivé komponenty samozřejmě musí OPC UA podporovat. Dále je mnohem více flexibilnější než veškeré předešlé standardy OPC dohromady, především tyto funkce sjednotil do jednoho komunikačního protokolu. Na taková zařízení poté prodejci kladou poměrně vysoké ceny. Není možné porovnat veškeré funkce UA s klasickými OPC rozhraními. Pouze není problém demonstrovat standartní funkce OPC na OPC UA.[8, str. 13]

OPC UA podporuje dva tradiční protokoly. Prvním je binární protokol a ten se označuje URL specifikací: opc.tcp://Server. Poskytuje velmi plynulou serializaci a deserializaci dat, které mají malé rozměry a efektivní formát na fyzické vrstvě.

Ten druhý se nazývá UA XML – je mnohem pomalejší než předchozí, ale byl aplikován, aby OPC mohlo komunikovat s jinými aplikacemi na různých úrovních fabriky. Dále ho tvoří dva typy transportních protokolů, které slouží na propojení mezi serverem a klientem UA TCP a Soap/HTTP(S).[8]

Další významný rozdíl, kterým disponuje nové rozhraní je, že na rozdíl od OPC classic nemusí být nastaveno rozhraní DCOM. OPC UA je sama o sobě síťová komunikace, tudíž musí zahrnovat jednotlivé mechanismy, které garantují bezpečný chod komunikace.[7]

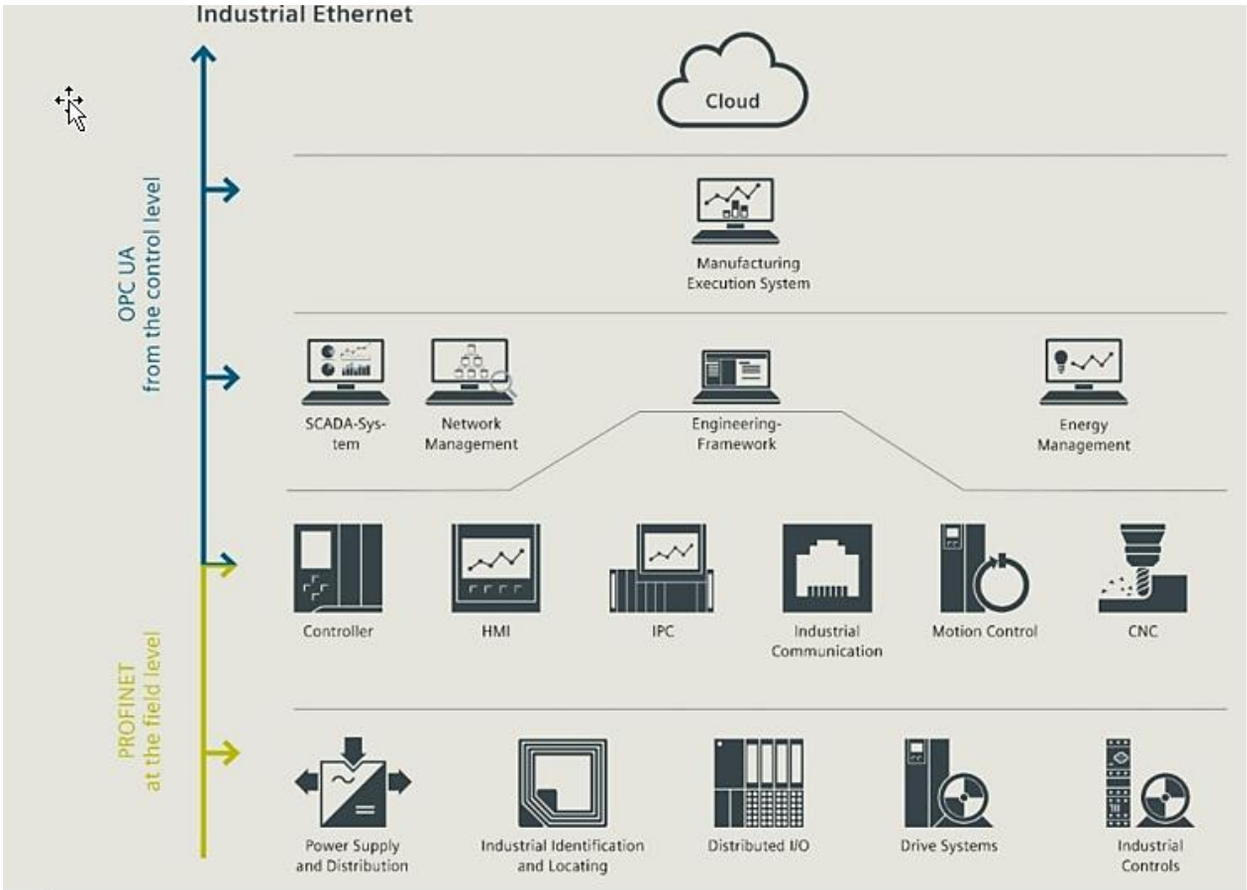
Rád bych vysvětlil, co je to OPC UA pouze jednou větou, ale myslím si, že to není úplně možné, tento komunikační standard se vyvíjel poměrně dlouhou dobu a pravděpodobně se bude vyvíjet dál kvůli narůstajícím požadavkům automatizace a rychlým vývojem průmyslu. Podle mě, ale tuto komunikaci nejlépe vystihuje věta: OPC UA je platformově nezávislý a velmi bezpečný standard pro komunikaci, který integruje veškeré individuální prvky předchozí verze OPC Classic do jedné stabilní sféry.[9]

2.1 Využití OPC UA v průmyslové výrobě

Průmyslová výroba vyžaduje řadu komplexních operací a klade velké nároky na efektivitu zpracování komunikačních požadavků. Veškerá zařízení ve výrobě jako jsou stroje, sensory, servery, aplikace atd. vytváří mnoho výstupů a datových bodů. Tato data většinou musí proudit mezi stroji a příslušnými zařízeními, a nakonec se analyzují, tak aby mohli pomoc zvýšit produkci výroby, snížit odpad výrobou vytvořený, zvýšit zisky a vytvořit nový řetězec hodnot společně s byznys plánem. V řadě případů je také nutné brát na zřetel rozdílnou zeměpisnou polohu a z toho vyplívající nutnost zvládnout národnostní specifika továren jednotlivých výrobců potravin, nápojů, automobilového průmyslu aj. Každá z těchto továren má své vlastní systémy výrobní postupy a zařízení. Většina společností se zaměřila na snížení odpadu, zvýšení produkce, a to vše v souladu s bezpečnostními prvky. Tento aspekt jsem již zmiňoval v úvodu své práce.

V souladu s těmito všemi požadavky každá továrna posílá obrovské množství dat na vzdálené servery, které rozdílným datům z několika zdrojů musí porozumět. Společnosti vyvinuli aplikaci, která funguje pouze na OS Linux, má udávat nový směr na základě přijatých dat z továren. Další aplikace byla vyvinuta pouze na OS Microsoft analyzovala nový směr na základě podání zpráv. Tato aplikace poté změnila nastavení strojů dle zjištěných zpráv. Společný cíl těchto aplikací byla výměna, přeložení a doručení dat.

Elegantním řešením těchto cílů je OPC UA. Společnost instaluje jednotlivé servery, které obdrží data od různých zařízení z odlišných továren a přeloží data do jednotného hlášení, které aplikace může dále využít. Dále můžou komunikovat mezi aplikacemi OS Windows, Linux a další. V neposlední řadě servery mohou zaslat finální upravené nastavení strojům zpět do továren v požadovaném formátu. Hlavní myšlenkou OPC UA je prolomit tradiční bariéry v průmyslové komunikaci.[10]



Obrázek 1 – Využití OPC UA ve fabrice [11]

Propojení jednotlivých sfér v moderním průmyslovém podniku za pomoci OPC UA nám krásně demonstruje obrázek 1.

2.1.1 Budoucnost OPC UA v průmyslu

Bez OPC UA by žádný pojem jako je Průmysl 4.0 neměl velký význam a ani by nemohl mít smysl. Na modelu průmyslu 4.0, digitalizizace a automatizace je postavený průmysl v příštích 10 letech a udává současné trendy nejen ve výrobě. Důsledkem toho každý produkt, který je prezentován v souladu s průmyslem 4.0 musí podporovat OPC UA. Hlanímí důvody proč využití OPC UA bude ještě více růst jsou:

- Umožňuje "chytrou" výrobu

- Přispívá k zjednodušení komunikace mezi stroji a dalšíimi přistroji a tím zvyšuje efektivitu výrobních podniků

- Jedná se o křízovou platfromu

- Může obdržet a upravit mnohonásobné datové body z rozdílných zdrojů[10, 12]

2.2 Terminologie OPC UA

V této kapitole bych rád shrnul několik důležitých pojmů a stručně je vysvětlil. Může se s nimi dostat do styku každý kdo OPC UA využívá, ale ne každý bude vědět co znamenají, tak že tato část mojí práce může posloužit i jako malá příručka. Terminologie je poněkud odlišná od předchozích druhů komunikace v průmyslové výrobě. Bude to způsobeno tím, že spojuje světy výroby a managmentu, díky tomu mohou být jednotlivé termíny poněkud zavádějící až matoucí. Uvedu pouze seznam těch nejdůležitějších.

2.2.1 OPC UA Application

V průmyslové tvorbě sítí obecně rozlišujeme koncovou aplikaci a zásobník protokolů. Koncová aplikace využije některé nastavení z definovaných funkcí. Zásobník protokolů přesune předem definována data mezi aplikací a nějakým externím zařízením ve velmi omezujícím prostředí. V případě OPC UA tomu tak úplně není. OPC UA aplikace odkazuje na koncovou aplikaci. Objektový model OPC UA a nastavení služeb OPC UA je realizováno zařízením s OPC UA.

2.2.2 OPC UA Client

Koncový bod OPC UA klienta je částí komunikace, která vyvolává komunikační relaci. OPC UA klienti mají schopnost objevit OPC UA servery. Zjistí, jak s nimi mají komunikovat, prozkoumají prostředí serverů a nakonfigurují je tak, aby se veškerá data doručila dle jejich požadavků. OPC UA klienti podporují mnoho rozdílných druhů mapování protokolů, tudíž mohou komunikovat se všemi možnými typy serverů. Tyto schopnosti jsou mnohem více přizpůsobivé než u jiných síťových klientů.

2.2.3 OPC UA server

Finální fází procesu komunikace OPC UA serveru je, že poskytuje data OPC UA klientovi. Co se týká funkcionality, výkonu či typu zařízení, tak neexistuje standard OPC UA serveru. Některé servery mohou poskytovat přístup pouze pár datovým bodům jiné zase tisícům. Servery například mohou využívat mapování s vysokou úrovní zabezpečení a nižším výkonem, jiné naopak bez zabezpečení, ale s využitím vysokého výkonu pomocí OPC UA Binary Encoding. Servery mohou být nabízeny

klientovi s pevnou konfigurací, která již nelze měnit nebo s možností úpravy nastavení.

2.2.4 OPC UA BLOB (Binary large object block)

BLOB poskytuje možnost přetvořit data, která nemají formát určený pro OPC UA. Data jsou určeny uživatelem a může se jednat o cokoliv: video, audio, složky, cokoliv jiného.

2.2.5 Protocol stack

Neboli zásobník protokolů, jímž jsou například Ethernet/IP nebo Profinet IO realizují v průmyslové síti datové modely a provoz protokolu. API (Application Program Interface) tzv. rozhraní pro programování aplikací, propojuje datový a provozní model s koncovou uživatelskou aplikací. API je nastavení softwarového prostředí, umožňuje jedné SW aplikaci využívat služby jiné [13]. Ve světě průmyslu je prezentován, jako prostředí mezi dvěma kousky SW ve stejném procesoru.

2.2.6 OPC UA encodings

Datové kódování je specifická cesta k přeměně OPC požadavku nebo odpovědi na přenos proudu bajtů. OPC UA momentálně podporuje dva typy kódování: OPC UA Binary a XML. OPC UA Binary je mnohem více kompaktní s menším přenosem dat, má méně kritických zón při chodu a vyšší výkon. Kódování je charakteristická cesta k mapování dat, které se objevují v síti. V binárním kódování jsou data mapovány do velmi kompaktních dat, které využívají méně bytů a více efektivní cestu k přenosu zpracování dat pomocí vestavěných systémů. Binární kódování je hojně využito v průmyslových automatizačních systémech, ale méně rozšířené v podnikových aplikacích. XML je běžné kódování, které je využito mnoha podnikovými systémy. Pro podnikové servery je XML méně náročné na provoz, ale vyžaduje více výkonu a vytváří víc chyb.

2.2.7 OPC UA Transport (přenos)

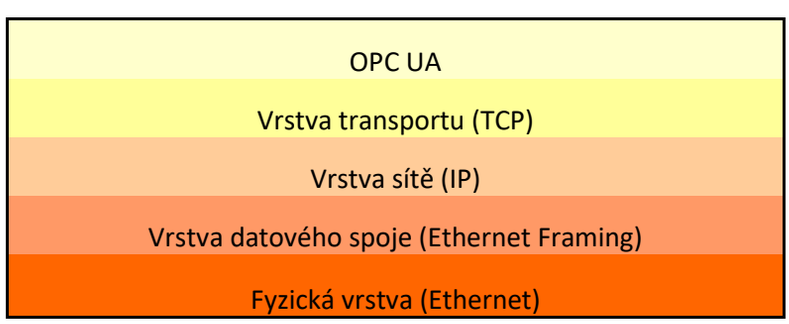
Transport je mechanismus, který přenáší zprávy mezi OPC UA klientem a serverem. Veškeré OPC UA zprávy jsou přenášeny přes TCP/IP připojení. Jakmile je pracovní zpráva zakódovaná a prošla bezpečnostní prověrkou, je připravená k odeslání. Dva transportní protokoly jsou využívány v OPC UA: TCP a HTTP/SOAP. Společnou

vlastností pro oba protokoly je standard TCP. TCP umožňuje komunikaci přes koncový bod počítačovou sítí mezi klientem a serverem. TCP je poněkud malý protokol, který založí jednoduchou komunikaci mezi klientem a serverem. Jeho největší výhodou je velikost a zanedbatelný dopad na průchod dat. HTTP je základním pilířem dění na internetu. Jedná se o protokol na nižší úrovni, který umožní klientovi např. prohlížení webových stránek z webového serveru. Podporují ho veškeré známé aplikace na internetu. SOAP rozšiřuje XML a poskytuje větší rozlet ve funkčnosti. Mimo jiné má schopnost provést vzdálené připojení k počítači a započít uvnitř proces v rámci XML struktury. Přenos pomocí protokolu HTTP/SOAP je podporován téměř všemi podnikovými procesy. Jedná se o standartní cestu k přenášení serializovaných dat mezi klientem a serverem. Serializace je proces načítání jednotlivých atributů a vytvoření série bytů, které OPC UA server může zpracovat a případně vrátit jejich hodnotu. [14]

2.3 Zabezpečení a komunikace

V této části se pokusím shrnout několik běžných aspektů zabezpečení a komunikace. Aspekty zabezpečení zahrnují šifrování a podepisování dat, která jsou přenášena mezi dvěma systémy, identifikaci aplikací (serveru a klienta), autentizaci a autorizaci uživatele klientské aplikace, přenos dat skrz firewall a audit. Při zohlednění bezpečnosti je také důležité zvážit v jakém prostředí jsou aplikace vykonány. OPC UA obsahuje specifikaci, která popisuje bezpečnostní hrozby a útoky a také jak je OPC UA navrženo, aby těmto hrozbám a útokům mohlo čelit. OPC UA jakožto standard není uzavřen do jednoho komunikačního přenosu nebo operačního systému, definuje tak bezpečnost ve vrstvě nad přenosem. To nám při přidání nových přenosů zajistí zachování bezpečnosti.

Tabulka 1 – TCP transport [15]



Tabulka 1 ilustruje TCP přenos, další přenosy můžou být dodatečně přidány do vrstvy mezi rozhraní TCP a OPC UA. Bezpečnostní prvky byly navrženy tak, aby mohly být snadno vylepšeny, bez zásahu do aplikace. OPC UA standard byl také vymodelován na základě používaných a osvědčených postupů v internetovém bankovnictví. [15]

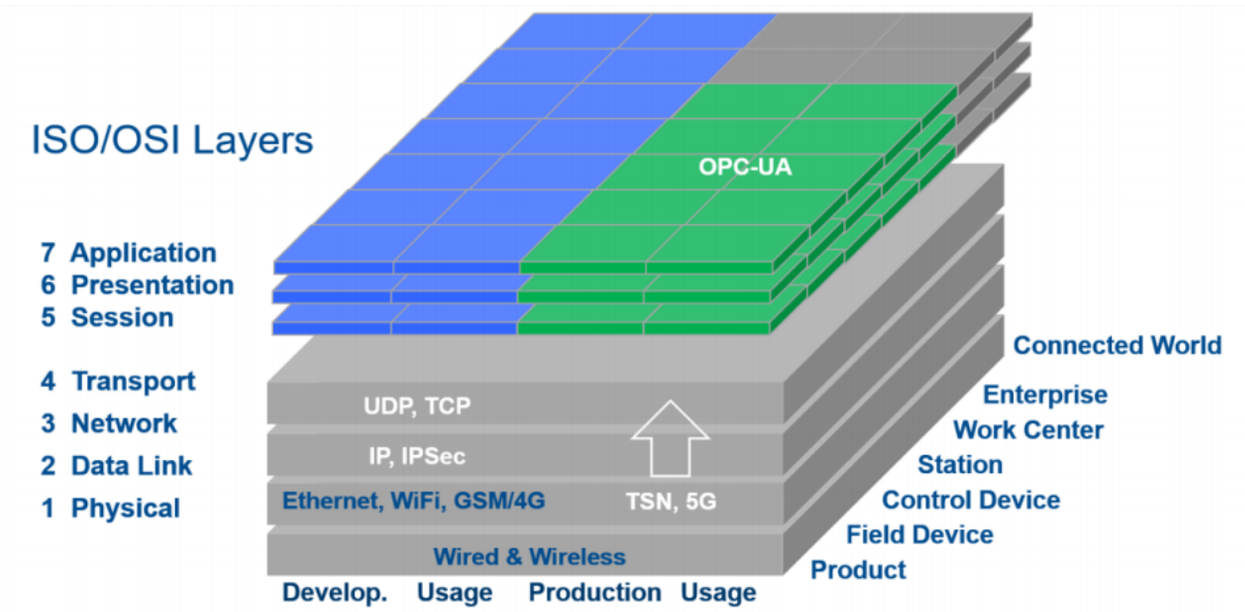
2.3.1 ISO/OSI model

ISO (International Organization for Standardization) je mezinárodní organizace pro normalizaci založená v roce 1947. Pokrývá širokou škálu technických problémů, mimo jiné globální standardy pro komunikaci a výměnu informací. Tato organizace založila v roce 1977 vlastní výbor pod názvem OSI (Open system interconnection). ISO/OSI referenční model byl vyvinut pro standardizaci počítačové sítě. V roce 1984 byl přijat jako mezinárodní norma. Referenční model je tvořen sedmi vrstvami a specifikuje protokoly na jednotlivých vrstvách a spolupracuje mezi nimi. [16, str.13]

1. Fyzická vrstva (Physical layer) – umožňuje přenos jednotlivých bitu komunikačním kanálem bez ohledu na jejich význam. Tato vrstva definuje fyzické signály využité k reprezentaci log 1 a log 0. Vrstva také předepisuje vlastnosti přenosového média, charakteristiky signálu a rychlost přenosu.
2. Linková vrstva (Data link) – úkolem vrstvy je zajistit bezchybný přenos dat mezi přímo propojenými sousedními stanicemi. Vytváří rámce (frames), které obsahují mimo vlastních přenášených informací i údaje pro adresování a zabezpečení proti chybám přenosu a údaj pro rozpoznání začátku rámce. Přidá tedy (v sítích TCP/IP, Ethernet) před paket tzv. preamble (synchronizační pole), příznak začátku rámce (1B), adresu cílovou (6B), zdrojovou adresu (6B), délku paketu (2B). Potom následuje vlastní paket a za ním kontrolní součet (CRC – cyclic redundancy check).
3. Síťová vrstva (Network) – zajištuje adresování a směrování dat v síti od zdroje k cíli přes několik sousedních prvků. Přenosová cesta se buď dynamicky mění při průchodu paket jednotlivými prvky sítě, nebo se na začátku spojení nejprve vytvoří virtuální cesta (spojově orientovaná cesta) – na této vrstvě pracuje Router.
4. Transportní vrstva (Transport) – zprostředkovává vlastní přenos dat. Přijímá data z relační vrstvy, rozkládá je na paket (nejmenší ucelená jednotka

přenášených dat) a přenese paket při každém přístupu na síťovou vrstvu. Jejím úkolem je, aby se celá zpráva dostala k příjemci v pořádku. Zajišťuje tedy i opakování zprávy v případě chyby a její opětovné sestavení po přenosu.

5. Relační vrstva (Session) – navazuje jednotlivé relace mezi koncovými stanicemi. Má na starosti práva, hesla oznámení atd…
6. Prezentační vrstva (Presentation) – převádí formát do universální podoby přístupné pro celou síť. Zajišťuje například kódování, komprimaci dat, kryptografii a po přenosu také zpětný převod.
7. Aplikační vrstva (Application) – je to nejvyšší vrstva celé architektury a tvoří rozhraní k vlastnímu programu. [17]



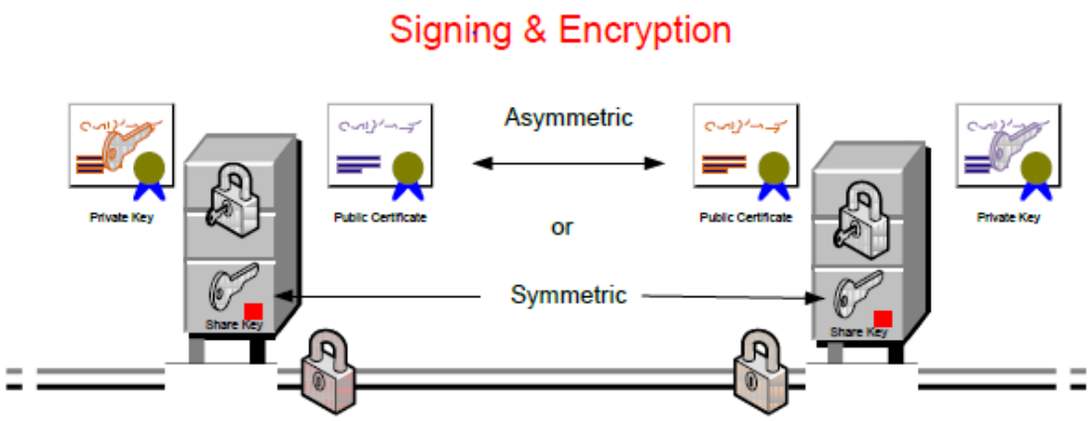
Obrázek 2 – Komunikační vrstvy s OPC UA mapovány ISO/OSI modelem [18]

Komunikační vrstvy jsou rozděleny do malých individuálních bloků každý může být popsán samostaně. OPC UA se nachází v zeleně vyznačené vrstvě na obrázku (5-7 vrstva). Šedá oblast vyznačuje oblast cloud to cloud a komunikaci mezi podniky s cloudy. Celá vývojová sekce je označena modrou barvou. Šedá i modrá zóna se stále projednává celou komunitou průmyslu 4.0.

2.3.2 Podpisy a šifrování

Podpis proudu dat (zpráv) zaručuje, že nikdo nemůže změnit co bylo odesláno a přijato. Vyžaduje generování kryptografického podpisu, který lze snadno obnovit

příjemcem zprávy. Pokud bylo cokoliv změněno příjemce neobdrží stejný podpis a může označit zprávu za změněnou. Šifrování uvádí podpisy na vyšší úroveň v tom, že nikdo jiný kromě příjemce nemůže zprávu přečíst. Využívá se šifrovací překladač obsahu zprávy, takže pouze příjemce má požadované informace k tomu, aby mohl zprávu dešifrovat. Výchozí nastavení pro OPC UA má povolené nastavení podpisů a šifrování. Konfiguraci serverů může zvolit z komunikačních možností, které jsou dostupné klientovi k připojení. Veškeré certifikované servery a klienti jsou povinni podporovat tento aspekt zabezpečení a jsou poskytovány pomocí zásobníků a balíčků které jsou využity ke stavbě aplikace. Nastavení musí být pouze aplikováno na servery, klienti musí zvolit, které z dostupných komunikačních nastavení budou využívat. [15]



Obrázek 3 – Šifrování a podpisy [15]

2.3.3 Identifikace a zabezpečení aplikací

Náplní této funkce je komunikace mezi aplikací a dalšími přiřazenými aplikacemi, zaručuje, že mezi ně nezasáhne žádný další klient nebo neodpovídající server. To je dáno tím, že klient komunikuje pouze s certifikovanými servery a ty odpovídají pouze certifikovaným klientům. Ve standartním nastavení OPC UA je tato funkce povinná a přednastavená. Stejná aplikace na dvou různých strojích může mít přístup s odlišnými přístupovými právy. Například standartní klient, který je nainstalován na dvou různých operátorských stanovištích může mít odlišný seznam serverů, ke kterým má povoleno připojení. Této funkce je úspěšně dosaženo tzv. certifikáty. [15]

2.3.4 Security policies, profiles

Profily v podstatě obsahují funkce, které aplikace musí podporovat, aby byly kompatibilní. Některé profily definují bezpečnostní funkce, jako jsou šifrovací algoritmy. Nastavení různých šifrovacích algoritmů může být podporováno aplikací OPC UA, ale jen jeden může být využit právě pro jedno připojení. Funkci zabezpečení nám definuje specifikace *Security policies*. [16, str.239]

Má čtyři úrovně:

1. 1.None
2. Basic128Rsa15
3. Basic256
4. Basic256Sha256[8, str. 15]

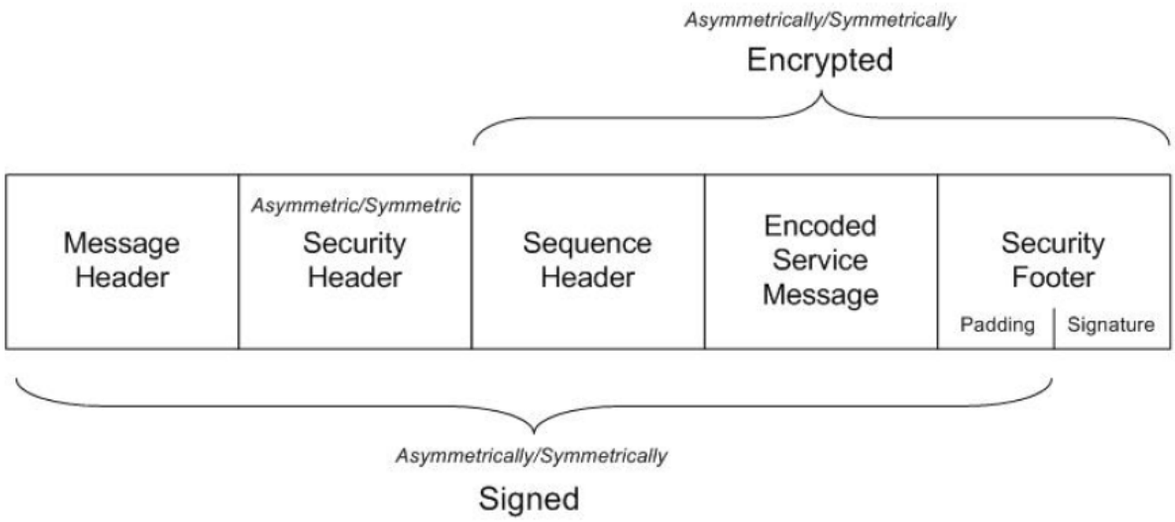
2.3.5 OPC UA Secure conversation

V prvé řadě musíme říct, že Secure conversation není další nový bezpečnostní protokol, jedná se spíše o kombinaci schválených technik a mechanismů využitých v technologických standardech TLS – (je kryptografický protokol, který nabízí možnost bezpečně komunikovat na internetu pro služby, jako jsou WWW, elektronická pošta, internetový fax a další datové přenosy) [20] a WS-secure conversation. WS-SecureConversation je protokol určený ke komunikaci v prostředí, kde jsou XML dokumenty vyměňovány například SOAP/HTTP. UA secure conversation nemapuje přímo abstraktní služby, ale využívá služební zprávy.

Message Security Mode – definuje, jak můžeme zprávy zabezpečit:[8, str. 15]

1. None
2. *Signed* - Odesílatel zašifruje svůj podpis soukromým klíčem a příjemce si veřejným klíčem zkontroluje zda je odeslaná zpráva opravdu od původního odesílatele.
3. Signed & Encrypted – Jedná se o ještě vyšší stupeň zabezpečení zprávy. Odesílatel zprávu zašifruje veřejným klíčem příjemce. Příjemce si poté zprávu rozšifruje svým soukromým klíčem

Část struktury této zprávy můžeme vidět na obrázku 4. [16, str. 213]



Obrázek 4 - Struktura šifrované zprávy [16, str. 213]

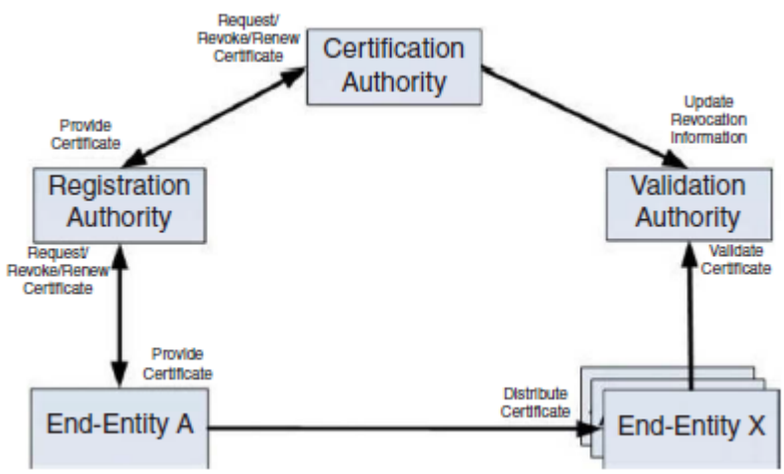
2.3.6 OPC UA certifikáty

Jedná se o poměrně složité téma, já zde vypíši pouze základní myšlenku a typy které se v OPC UA využívají. Obecně známo je certifikát oficiální dokument pojednávající o nějakých faktech. V případě OPC UA hovoříme o tzv. digitálních certifikátech, jsou to elektronické dokumenty obsahující informace potvrzené důvěryhodnou třetí stranou. V podstatě jsou certifikáty využity k distribuci veřejných klíčů, které jsou využity ke kryptografii klíčů mezi jednotlivými entitami, využívají je pro různé účely, například k šifrování dat. Veškeré aplikace mají přiřazený svůj unikátní certifikát a seznam povolení, který ukazuje, jaké další certifikáty jsou povoleny [15]. OPC UA certifikáty musí splňovat dvě hlavní myšlenky: prvním z nich je, že musí svázat speciální informaci s unikátním klíčem a společně je přiřadit specifickému majiteli, příjemce certifikátu může definovat majitele. Dalším cílem je zaručení bezúhonnosti veřejným klíčům s přidruženými daty v případě zjištění zásahu třetí stranou. Existuje mnoho formátů těchto certifikátů například X.509v3, SPKI, PGP a atributy. V OPC UA se převážně využívá X.509.3 certifikát, který je nejvíce rozšířeným typem. OPC UA využívá k založení připojení tři různé druhy certifikátů:

1. OPC UA Application Instance Certificates – Certifikáty aplikačních instancí vyžaduje ho každá instalace OPC UA. Tento certifikát identifikuje probíhající případ v OPC UA aplikaci a získá od nich důvěryhodná data pro konkrétní případ.

2. OPC UA Software Certificates – SW certifikát identifikuje charakteristickou verzi OPC UA produktu. Obsahuje dodatečné rozšíření v3, které má otestované a již prověřené OPC UA profily pro daný produkt. Výměnou těchto informací v průběhu založení připojení obě aplikace mají informace o tom, s kým mohou komunikovat a zda mohou podporovat jejich služby. OPC UA profily obecně obsahují možnosti, které aplikace musí znát. Některé profily určují bezpečnostní funkce, jako je například šifrovací algoritmus.
3. OPC User Certficates – Posledním typem je uživatelský certifikát, jehož funkcí je identifikovat aktuálního uživatele, který má v úmyslu číst data v průběhu navazování spojení. [16, str. 240-243]

K řízení digitálních certifikátů se využívá tzv. PKI (Public Key Infrastracture), jedná se o značení infrastruktury zprávy a distribuce veřejných klíčů z asymetrické kryptografie, dále umožňuje používat cizí veřejné klíče a ověřovat s nimi elektronické podpisy bez nutnosti jejich individuální kontroly, ale pouze v případě udělení důvěry jednotlivých uživatelů. Dále zahrnuje mnoho dalších bezpečnostních aspektů: šifrovací klíče, všechny možné kryptografie, certifikační autoritu (CA) atd.[21]



Obrázek 5 – Struktura subjektů PKI [16, str. 245]

CA je subjekt, který obnovuje vydává a ruší certifikáty, také informuje ostatní subjekty, například když jsou certifikáty zrušeny informuje autoritu ověření platnosti. Registrační autorita (RA) a CA jsou často kombinovány dohromady, protože jsou jejich vztahy silně propojeny. [16, str. 245]

2.3.7 Uživatelská práva pro přístup

Jedná se o omezení, která určují, k jakým datům můžeme mít přístup na daném serveru a jakým způsobem se můžeme přihlásit. Práva nám určují, zda můžeme data číst, upravovat nebo procházet adresním prostorem. [15] OPC UA nabízí čtyři základní úrovně zabezpečení.

1. Úroveň – Anonymní (Bez autentizace)

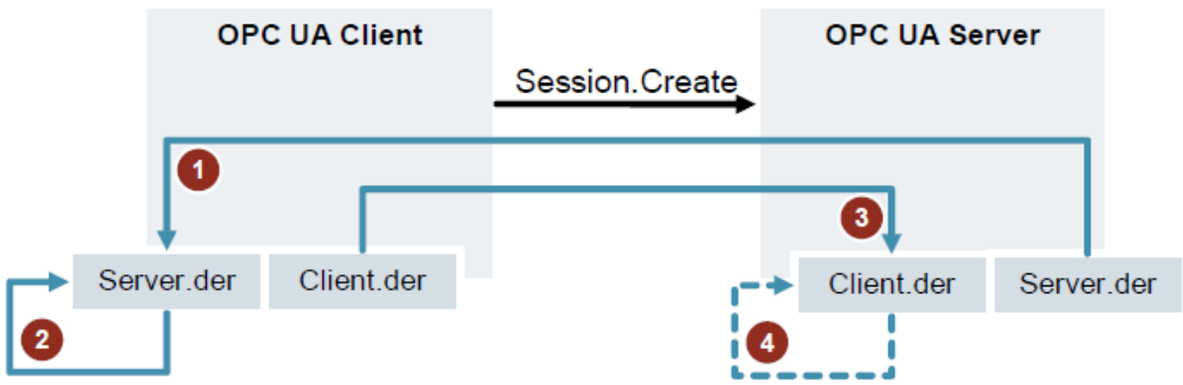
V případě, že zvolíme anonymní nastavení klient a server umožní přístup komukoliv, kdo se bude chtít přihlásit. Ve finále to znamená, že veškeré platné certifikáty budou považovány za důvěryhodné. Aplikační certifikáty jsou využity pouze pro udělení informací o další straně, které nelze ověřit. V tomto případě pro příjemce neexistuje žádná možnost, jakou by mohl prověřit, zda je poskytovatel legitimní vlastník certifikátu. Na této úrovni nejsou vyžadována žádná nastavení na straně klienta ani serveru. Anonymní úroveň automaticky uznává platnost certifikátů i když nejsou obsaženy v seznamu důvěryhodných certifikátů.

2. Úroveň – Serverová autentizace (Jméno/heslo)

Pokud zvolíme tuto možnost, tak se jakýkoliv klient může připojit na daný server. Ověření klienta (pokud je vyžadováno) se provádí za pomoci ověřovacích údajů (jméno/heslo), které jsou zaslané na server po vytvoření komunikačního kanálu. Každý klient musí udělit důvěru serverovému certifikátu, tuto možnost nastaví administrátor na straně klienta (veřejný serverový klíč také musí být uveden v seznamu důvěryhodných certifikátů, nebo musí být vystaven důvěryhodnou certifikační autoritou). Pokud serverový certifikát není dodatečně uveden na seznamu certifikát, pak klient musí podporovat DNS jméno uvedené na serverovém certifikátu s DNS jménem počítače, ke kterému se snaží připojit. Tento způsob zaručí připojení k odpovídajícímu počítači, ale nezaručí správné připojení k OPC UA serveru. Tato úroveň zajišťuje kvalitní úroveň zabezpečení. Využívá se například pro přístup do internetového bankovnictví, ale server nemůže omezit klientské aplikace na základě jejich autentizace.

3. Úroveň – Autentizace pomocí certifikátu

V tomto případě server povolí přístup pouze klientům s důvěryhodným certifikátem, a to x509v3 certifikátem více o něm je v 2.3.5 OPC UA Secure conversation. Tato úroveň se využívá v případě služeb, které udělují přístup pouze důvěryhodným klientům a současně nepožaduje legitimní server. Server poskytne data v případě, že bude důvěřovat klientskému certifikátu. Toto nastavení opět provede administrátor na straně serveru (certifikát přímo přidá do seznamu důvěryhodných certifikátů nebo musí být klientský certifikát podepsán důvěryhodnou certifikační autoritou).



Obrázek 6 – Výměna certifikátu mezi klientem a serverem [8, str.17]

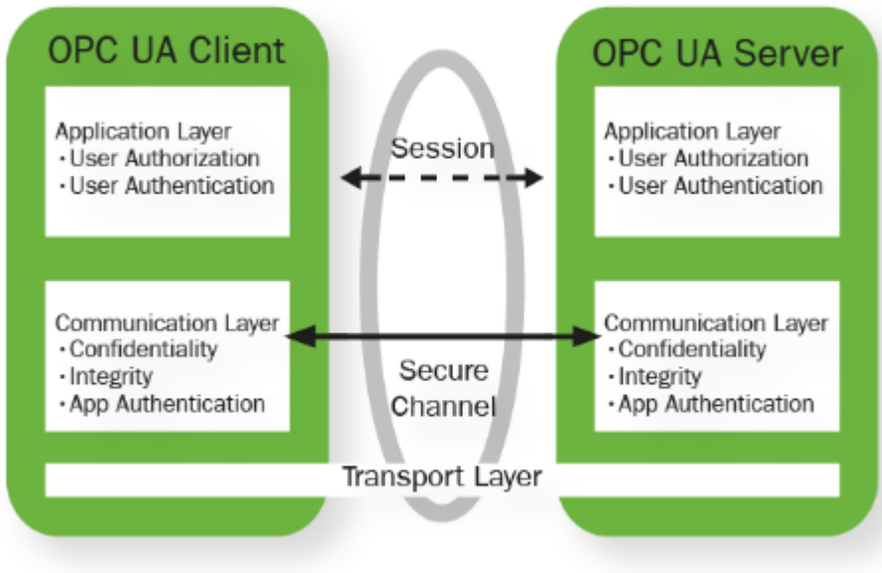
4. Úroveň – Oboustranná autentizace (*WS - SecurityToken*)

Jedná se o přihlášení pomocí tokenu vygenerovaného serverem. Poskytuje nejvyšší míru bezpečnosti, ale vyžaduje nastavení na obou stranách (klient /server). Pokud serverový certifikát není výslovně důvěryhodný, v tom případě klient ověří server stejně jako v serverové autentizaci.[7, 16, str.237-238]

2.3.8 Secure channel session

Při navazování spojení mezi klientem a serverem se nejprve vytvoří *Secure channel,* který má v popisu práce zabezpečit komunikaci na nízké úrovni, na jehož vrcholu se vytvoří *Session. Secure channel* a *Session* nemají za úkol posílat data, ale pouze zajistit spolehlivou a bezpečnou komunikaci. *Secure channel* a *Session* pracují nezávisle na sobě, při výpadku spojení se vytvoří nový *Secure channel* a přidá se k již existující *Session. Session* má předem definovaný čas, za který neproběhne žádná komunikace mezi serverem a klientem, server povolí veškeré možnosti spojené s touto *Session.* S každým požadavkem klienta na server se tento čas vynuluje. Při založení komunikace mezi serverem a klientem, server nejdříve zašle identifikační

klíč, který se využívá ve všech ostatních službách, aby byla jistota, že server komunikuje se správným uživatelem. [1, 8, str. 21]



Obrázek 7 – Navázání komunikace [1]

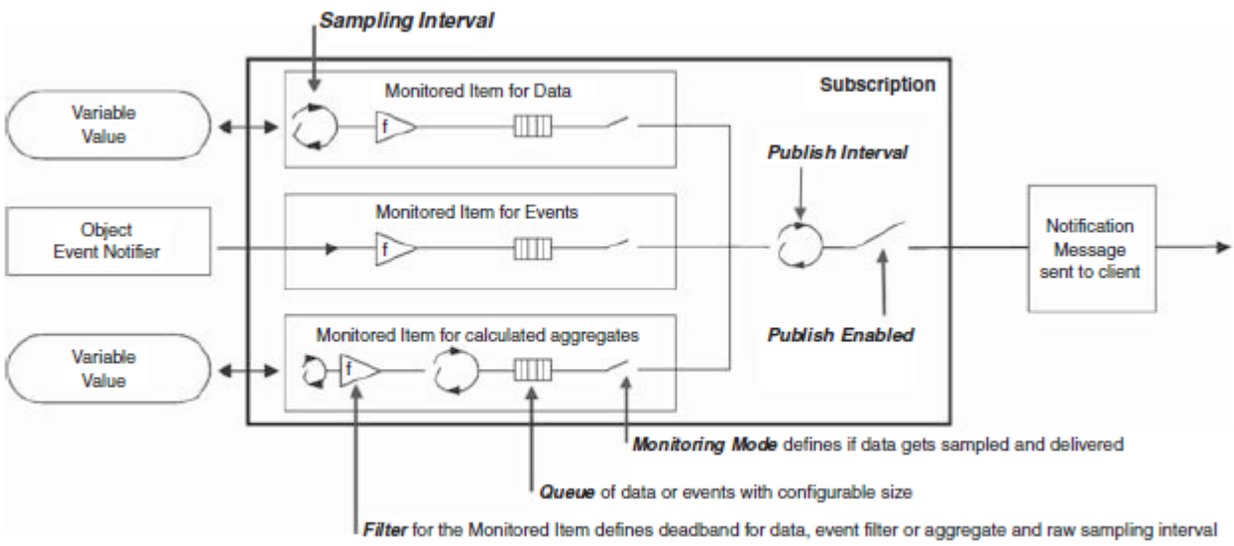
2.3.9 Subscription

Klient smí odebírat (*Subscribe*) tři typy informací: změny hodnot, událostí a sdružených hodnot. Klient při vytváření *Subscription* určuje, jak často mu má server posílat hodnoty odebíraných proměnných, tento parametr se nazývá *Publish Interval*. Poté vytváří z proměnných, které bude sledovat tzv. *Monitored Items*, kterými nastavuje při jaké změně hodnoty (*filter*) má server odeslat upozornění, jak často má server vzorkovat (*Sampling Interval*) odpovídající proměnou, kolik změn má uchovat v případě kdy neproběhne odeslání na server a typ monitorovacího módu.[8,str.22 19, str.176]

Monitorování proměnných:[19, str. 176]

1. Server vzorkuje odebrané proměnné dle definovaného času (*Sampling Interval*)
2. Po ukončení vzorkování ve filtru, zkontroluje, jestli se změnila hodnota o vyžádanou hodnotu nebo procenta.
3. Jakmile hodnota překročí přes *filtr*, proměnná se zařadí do řady. Délka této řady je definována klientem. Ve chvíli, kdy je řada plná, tak se začne přepisovat první nebo poslední hodnota v řadě podle konfigurace klienta.

4. Po skončení *Publish Interval* se hodnoty uložené v řadě odešlou klientovi ve chvíli, kdy je nastaven monitorovací mód *Reporting*. Když je nastaven mód *Sampling*, tak se hodnoty neodesílají, pouze se dále vzorkují.

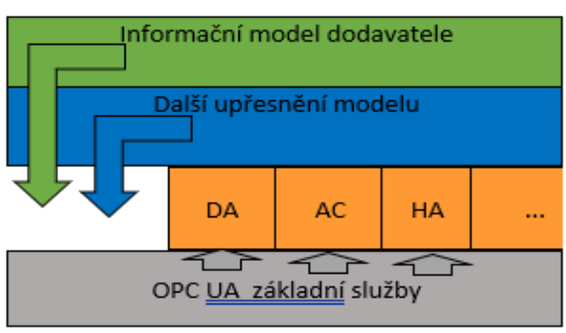


Obrázek 8 – Nastavení Subscritipiton a monitorování na serveru [19, str.176]

2.4 Služby

OPC UA služby definují data na komunikační úrovni. Služby jsou metody využívané klientem OPC UA k přístupu dat z Informačního modelu. Definici služeb můžeme chápat jako vzor, požadavek/odpověď při využití webových služeb. OPC UA využívá 37 služeb, z toho je 21 určených ke komunikaci a 16 na výměnu informací.[8, str. 19]

Informační modely jsou v OPC UA velkým trendem. Tyto modely mohou být definovány výrobci případně protokoly, ale také mohou obsahovat více než velmi komplexní vztahy, propojení mezi jednotlivými body a definovanými uzly. Existuje také možnost, aby struktura dat vždy byla zpracována a uspořádána, jako jeden celek. Toto nastavení je důležité v mnoha aplikacích, kde chceme mít jistotu, že soubor dat je obdržen současně.[1]



Obrázek 9 – Informační model

Tabulka 2 – Rozdělení služeb do oddílů [16, kap. 5]

1. Vyhledávání serverů
– FindServers
– GetEndpoints
– RegisterServer
2. Navázání spojení mezi serverem a kliente
– OpenSecureChannel/CloseSecureChannel
– CreateSession/CloseSession
– ActivateSession
– CancelService
3. Vyhledávání informací v adresovém prostoru
– Browse
– BrowseNext
– TranslateBrowsePathsToNodeIds
4. Čtení a zápis dat a metadat
– Read
– Write
– RegisterNodesService
– UnregisterNodesService
5. Změny dat a událostí
– CreateSubscriptionService
– DeleteSubscriptionsService
– ModifySubscriptionService
– SetPublishingModeService
– TransferSubscriptionsService
– CreateMonitoredItemsService
– DeleteMonitoredItemsService
– ModifyMonitoredItemsService
– SetMonitoringModeService
– SetTriggeringService
– Publish
– Republish
6. Vyvolání metod na serveru
– Call
7. Přístup k archivu dat a událostí
– HistoryRead
– HistoryUpdate
8. Hledání informací v komplexním adresáři
– QueryFirst
– QueryNext
9. Modifikace adresového prostoru
– AddNodes
– DeleteNodes
– AddReferences
– DeleteReferences

3 PŘIPOJENÍ S7-1500 - MYSCADA

K realizování ukázky OPC UA jsem využil PLC od firmy Siemens. Konkrétně model Simatic 7-1500, CPU 1511TF-1PN (6ES7511-1UK01-0AB0) + paměťová karta (6ES7954-8LF03-0AA0) bez které bych nemohl nahrát žádný program do PLC a zdroj SITOP smart 5 A (6EP1333-2AA01). Obě zařízení jsou na číslo 10 níže. PLC je připojené k počítači ethernetovým kabelem.

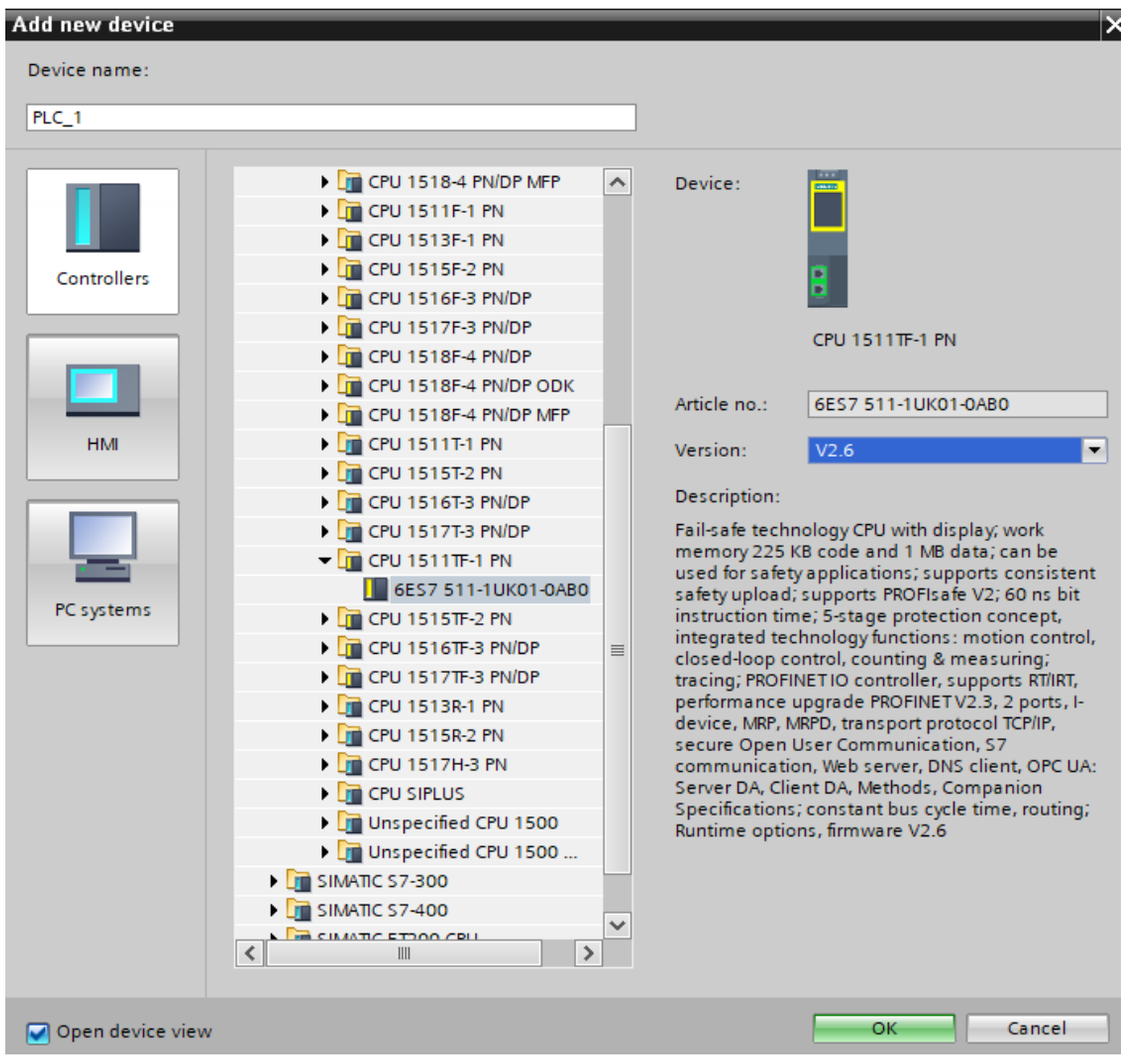
Obrázek 10 - S7 – 1500 a zdroj

Dále jsem k nastavení OPC UA serveru potřeboval správnou verzi TIA portálu, a to tu nejnovější TIA portál V 15.1. Celý balíček Softwaru si můžete stáhnout zdarma na oficiálních stránkách firmy Siemens a volně využívat po dobu třiceti dní. Po vypršení této zkušební lhůty se zablokuje většina funkcí. Tento SW je poměrně finančně nákladný. Nakonec programy myDesigner 8 a myPRO od společnosti mySCADA. Oba jsou rovněž volně dostupné, ale pouze pro nekomerční využití a mají různá omezení.

3.1 Připojení – TIA

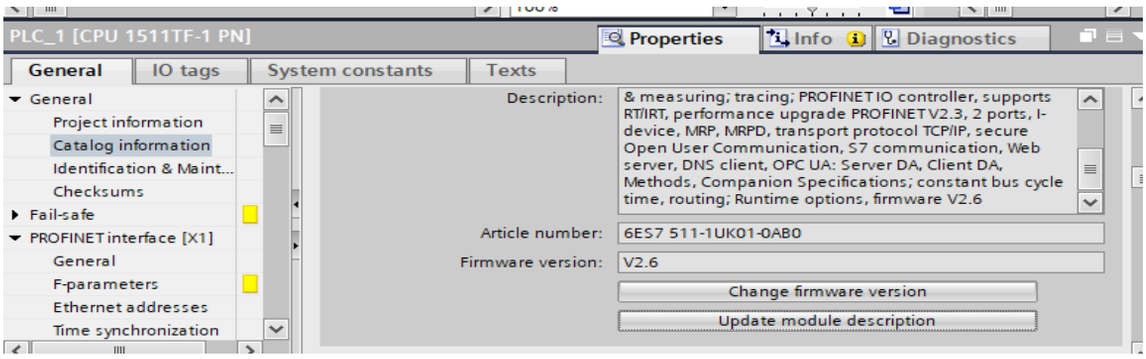
OPC UA server, který vytvořila společnost Siemens se nachází v PLC. Je nutné vždy mít správnou verzi TIA portálu (nejlépe tu nejnovější), která dané PLC podporuje, původně jsem chtěl využít V15, myslel jsem, že bude dostačující, ale tato verze ještě nepodporovala moje konkrétní PLC a nedařilo se mi k němu připojit. Nyní projdu všemi nutnými kroky, které musím provést, abych se přes OPC UA mohl připojit.

Nejdříve si vytvořím nový projekt v TIA portálu a přidám odpovídající zařízení, které chci využívat. Důležité je také nastavit odpovídající verzi firmwaru. V mém případě je to nejnovější V2.6 pro můj typ PLC. Firmware lze samozřejmě upgradovat v případě, kdy vyjde novější verze. V popisu můžeme vidět, jaké funkce tato verze firmwaru podporuje.



Obrázek 11 – Přidání zařízení

Zmíněný firmware můžeme dle potřeby měnit nebo, jak jsem zmiňoval výše upgradovat na vyšší verzi. Po přidání zařízení otevřeme *Project view* a v levém sloupci se nachází menu s funkcemi, rozklikneme *Device configuration – general – catalog information – change firrmware version.*

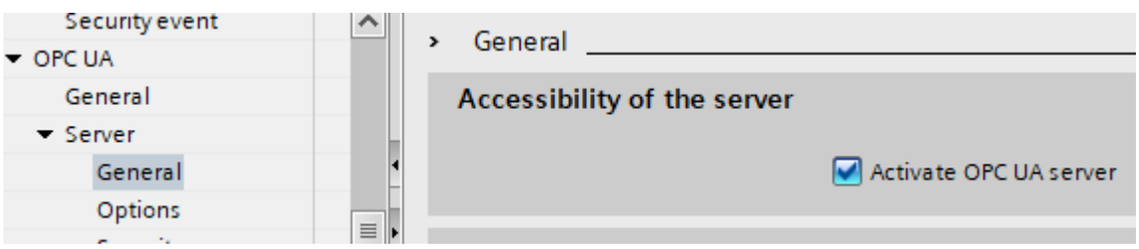


Obrázek 12 – Firmware

3.2 Nastavení OPC UA serveru - TIA

Nyní již můžeme přejít k samotným krokům nastavení OPC UA serveru, OPC UA server je v přednastavení vypnutý, musíme ho nejdříve zapnout a nastavit požadované hodnoty. Veškerá nastavení PLC provádíme v *Device configuration - properties.*

1. Povolení OPC UA serveru, postup *OPC UA - Server – General* v okénku zaškrtneme možnost Activate OPC UA server.



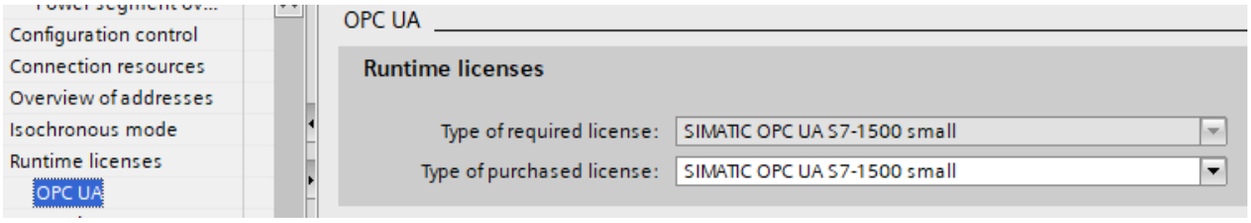
Obrázek 13 – TIA Povolení serveru

2. Musím zvolit správnou licenci, bez tohoto kroku by se nepřeložil projekt, pro jednotlivé řady PLC je nutné využít správnou licenci viz. obrázek níže.

Target system	OPC UA S7-1500 Small	OPC UA S7-1500 Medium	OPC UA S7-1500 Large
ET 200SP CPU 1510SP/1512SP/ 1515SP (Open Controller) S7-1500 CPU 1511/1513	**yes**	yes	yes
ET 200pro CPU 1516pro S7-1500 CPU 1515/1516 Software PLC 1507S	no	**yes**	yes
S7-1500 CPU 1517/1518	no	no	**yes**

Obrázek 14 – Runtime licence [22]

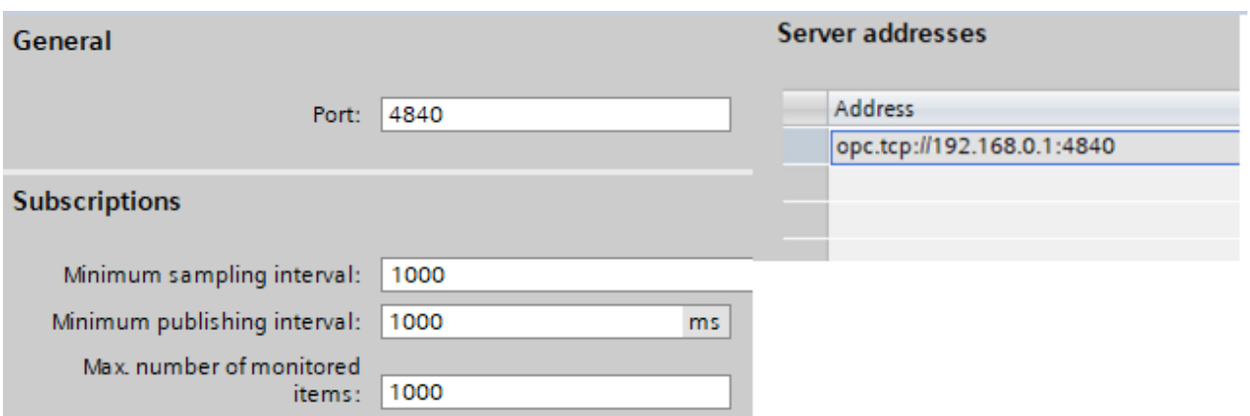
Postup v TIA portálu, *Runtime licence – OPC UA* a zvolím licenci small.



Obrázek 15 – TIA licence

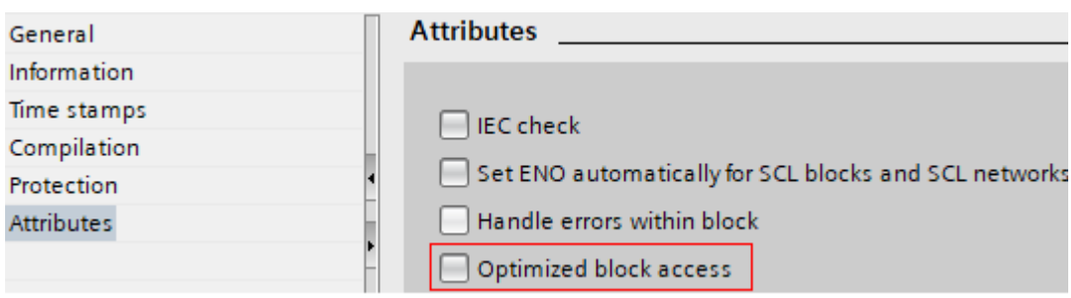
Tato nastavení nám povolí OPC UA server v PLC a zaručí základní operace. V tomto základním nastavení se k serveru může připojit jakýkoliv klient.

3. Nastavení Portu a Subscriptions, postup *OPC UA – Server – Options*. Port si můžeme nastavit jaký chceme, nechávám 4840, se změnou portu by se nám také změnila serverová adresa. Více o Subscriptions můžete nalézt v kapitole 2.3.9 Subscription. IP adresu nastavovat nemusíme, ta se přidělí automaticky.



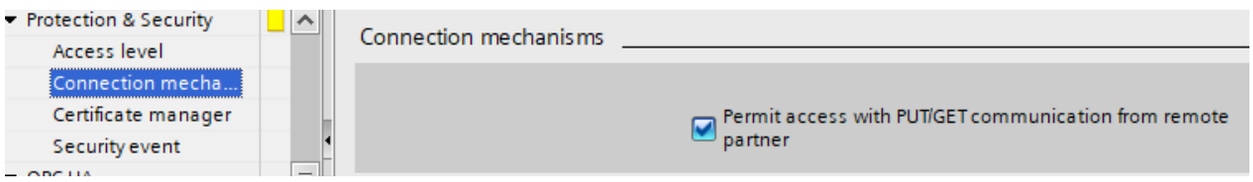
Obrázek 16 – TIA – port, subscriptions a serverová adresa

4. Nyní ještě musím provést nastavení nutná k tomu, abych se mohl připojit k myScadě (klientu). [23] MyScada podporuje pouze globální přístup k datovým blokům, tudíž musím odškrtnout variantu *Optimized block access.* Postup – v levém menu rozklikneme Program blocks – Main – Attributes



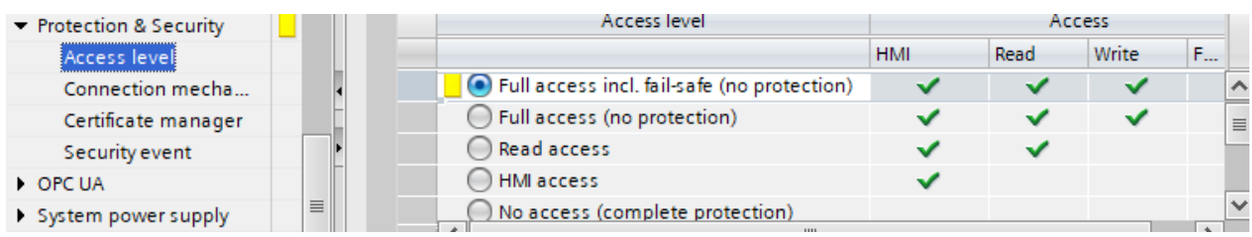
Obrázek 17 – TIA – myScada nastavení

Dále spojovací mechanismus musí povolit PUT/GET od vzdáleného partnera. Postup – Protection & Security – Connections mechanism



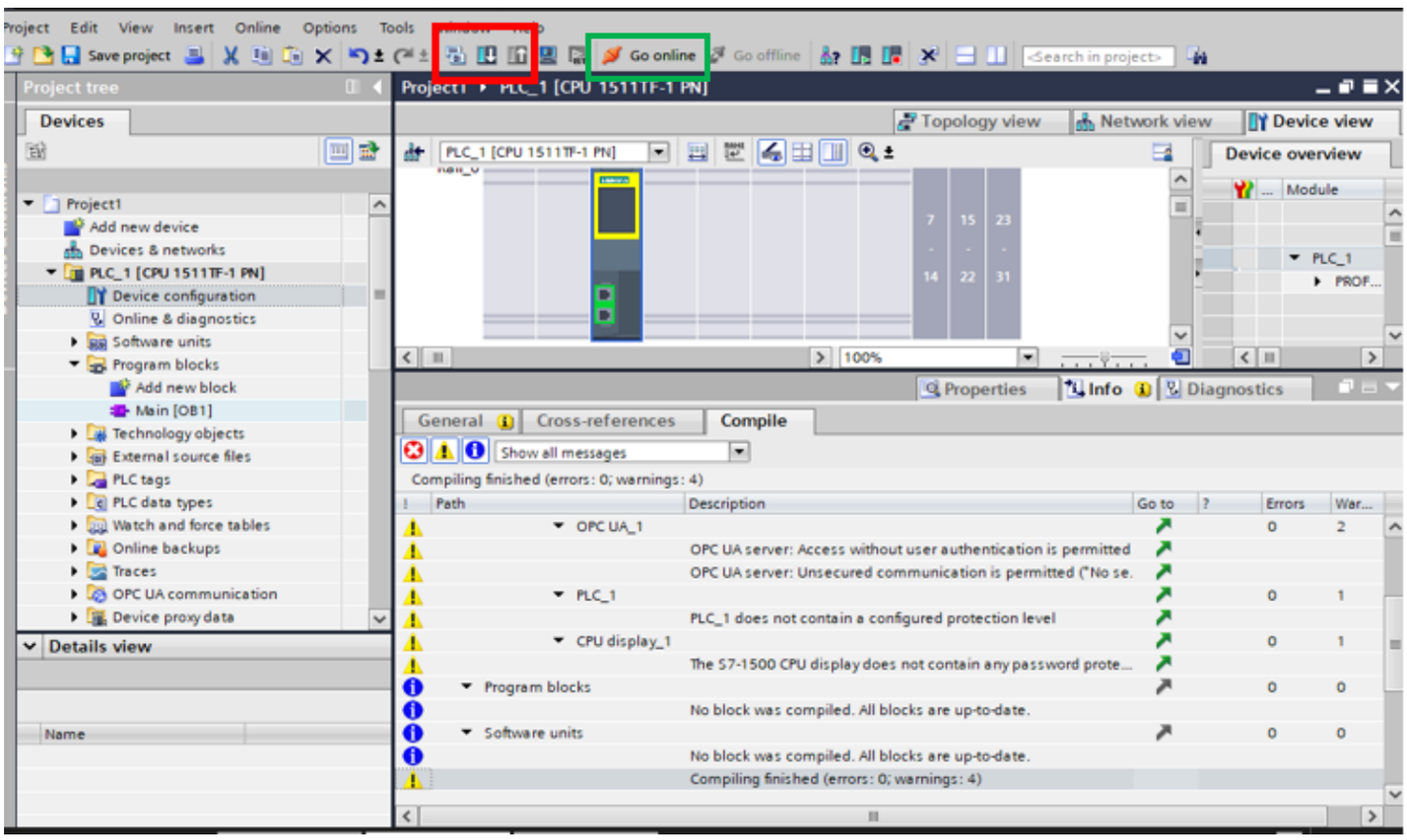
Obrázek 18 - TIA - myScada nastavení

Úroveň přístupu musí být úplná viz. obrázek 19, postup – Protection & Security – Access level.



Obrázek 19 - TIA - myScada nastavení

5. Nyní již mám nastavené vše, co potřebuji, tudíž mohu tato nastavení zkompilovat a nahrát do PLC. Klikneme na první ikonu v červeném rámečku (compile). Na obrázku č. 20 můžeme vidět veškeré změny, které jsem provedl, v případě, že některé z nastavení není v pořádku funkce (compile) nás upozorní a nedovolí program nahrát do PLC. To ovšem není můj případ. Tudíž můžeme program nahrát do PLC, tak že klikneme na ikonu vpravo v červeném rámečku (download). Poté vyskočí okno dáme load a nahráli jsme úspěšně program (nastavení) do PLC. Nyní již jen klikneme na GO online ikona v zeleném rámečku, abychom pracovali v reálném čase, nyní mám vše potřebné v TIA portálu nastavené a můžu se přesunout k nastavení myDesigner8.



Obrázek 20 – TIA kompilace

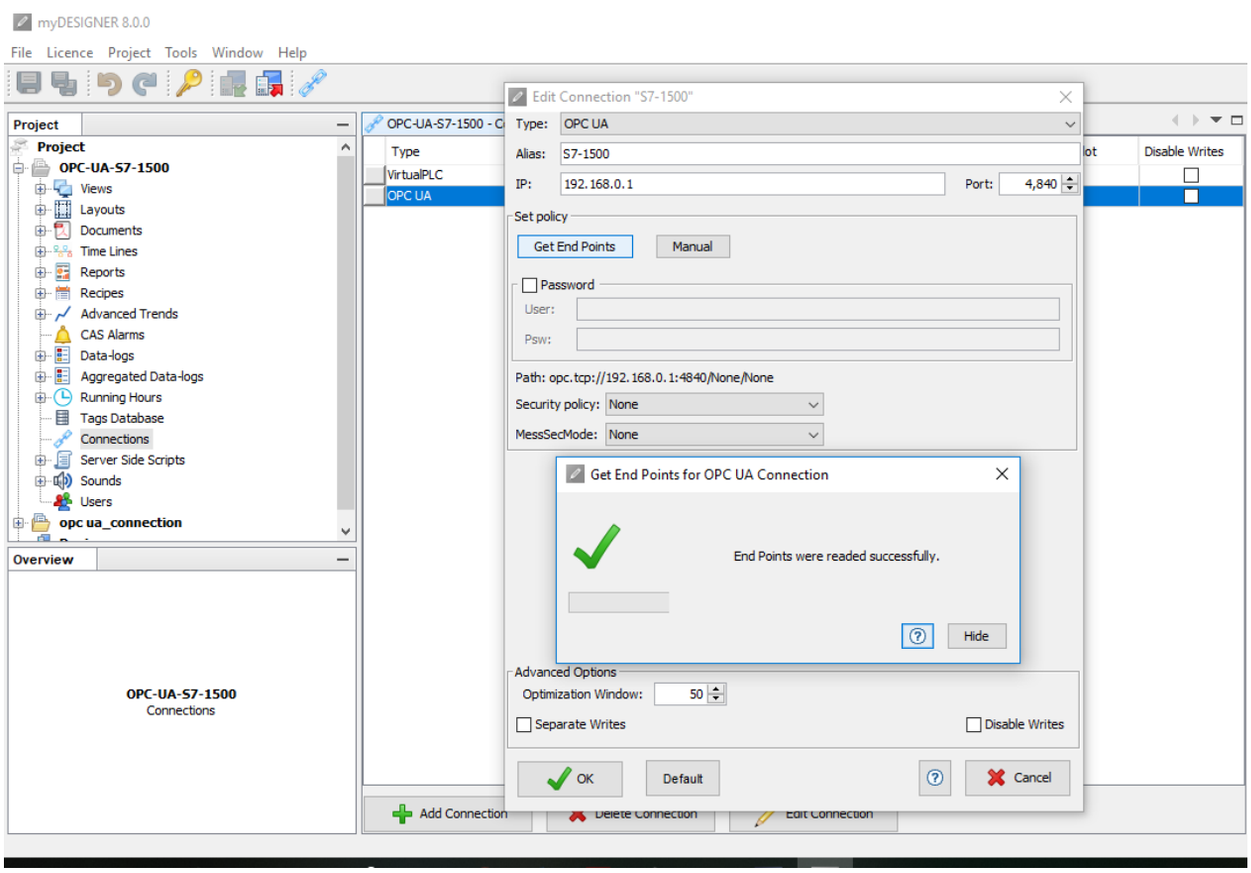
3.3 Nastavení myScada

Jakmile jsem si stáhl programy myDesigner 8 a myPro můžu pokračovat. Program myPro slouží k zobrazení projektu, který si vytvořím v myDesigeneru 8. Budu využívat webový přístup. Nesmíme zapomínat na fakt, že OPC UA server nekomunikuje přímo s PLC i přes skutečnost, že se nachází v samotném PLC. OPC UA server se nachází uprostřed a veškerá komunikace závisí na nastavení OPC UA serveru.[24]

Obrázek 21 – OPC UA myScada

3.3.1 MyDesigner 8 - nové připojení

Nejdříve si musím založit nový projekt. *Project – projects – new project – empty*. V dalším kroku přidám nové připojení.

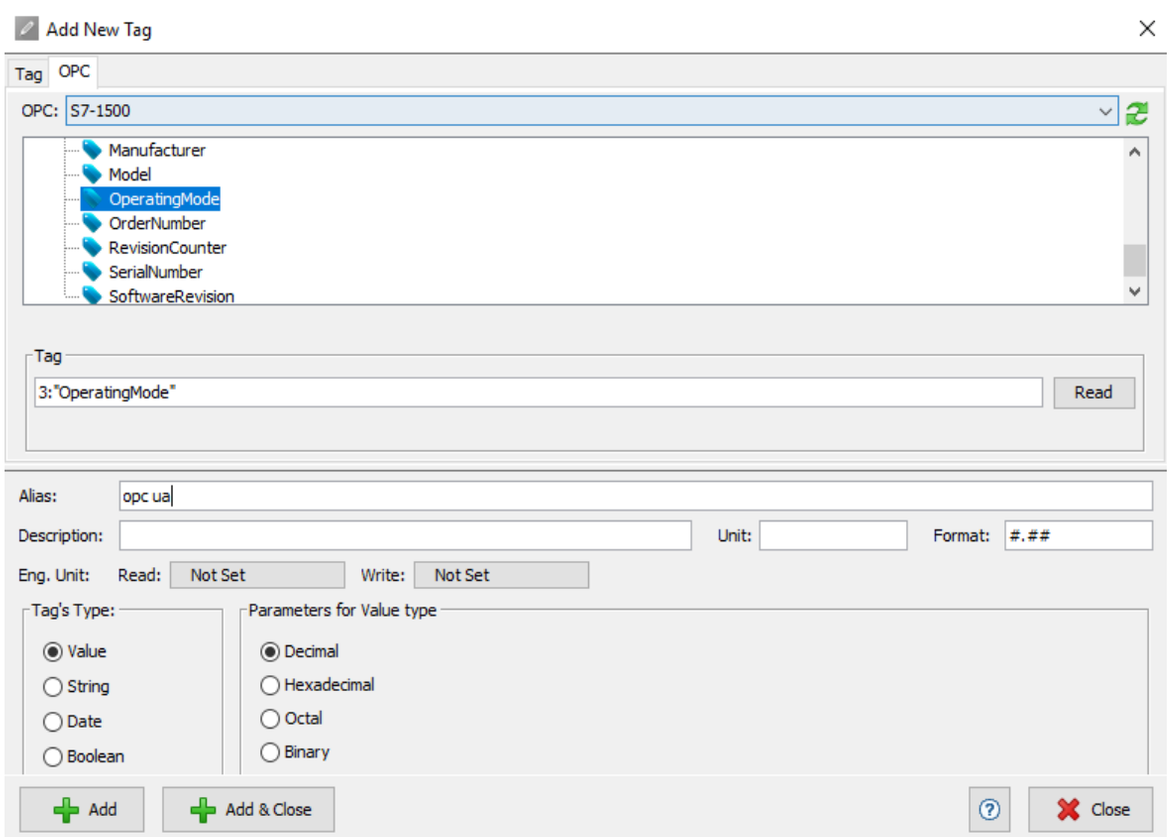


Obrázek 22 – myScada připojení

Na levé straně se nachází *Project tree* vyberu *connections – add connection* vyskočí nám nové okno a nyní definuji požadované připojení viz. obrázek 22. *Type* – zvolím OPC UA, Alias (název) – může být jakýkoliv, IP, *Port* a *Path* se mi musí schodovat s nastavením v projektu v TIA portále viz. kapitola 3.2 Nastavení OPC UA serveru. MyScada implementace OPC UA podporuje pouze binární protokol tzn. (*opc.tcp://Server),* nepodporuje webovou verzi (http://Server), více o této problematice můžeme nalézt v kapitole 2 CO JE TO OPC UA? a kapitole 2.2.6 OPC UA encodings .Zda je vše v pořádku zjistím kliknutím na *get end points*, program si automaticky načte nastavení pro připojení ze serveru.

3.3.2 Tag database

Nyní si vytvořím tag, který později přiřadím. V případě, kdybych programoval úlohu o více tagách, vyexportuji si tuto databázi tagů z TIA portálu do excelu a poté tento soubor pouze nahraji do programu myDesigner. Já budu potřebovat k mojí demonstraci pouze jeden. Z levého meny vyberu *Tags database,* poté postupuji –

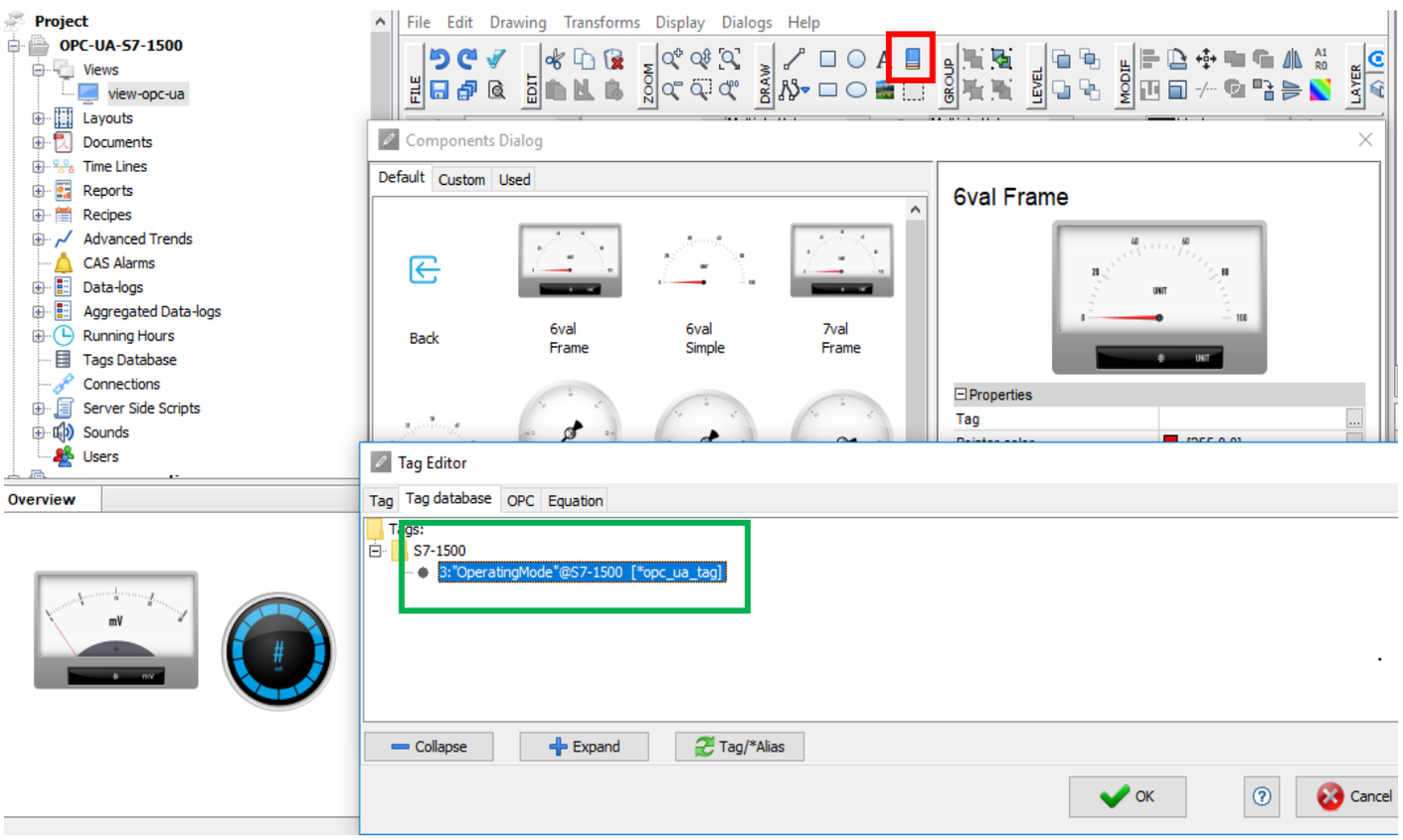


Obrázek 23 – myScada – Tags database

add new tag – OPC-PLC1-Operating mode. Nějak ho pojmenuji (Alias) a nakonec *add.* Nyní se tento tag již nachází v databázi a můžu ho využít.

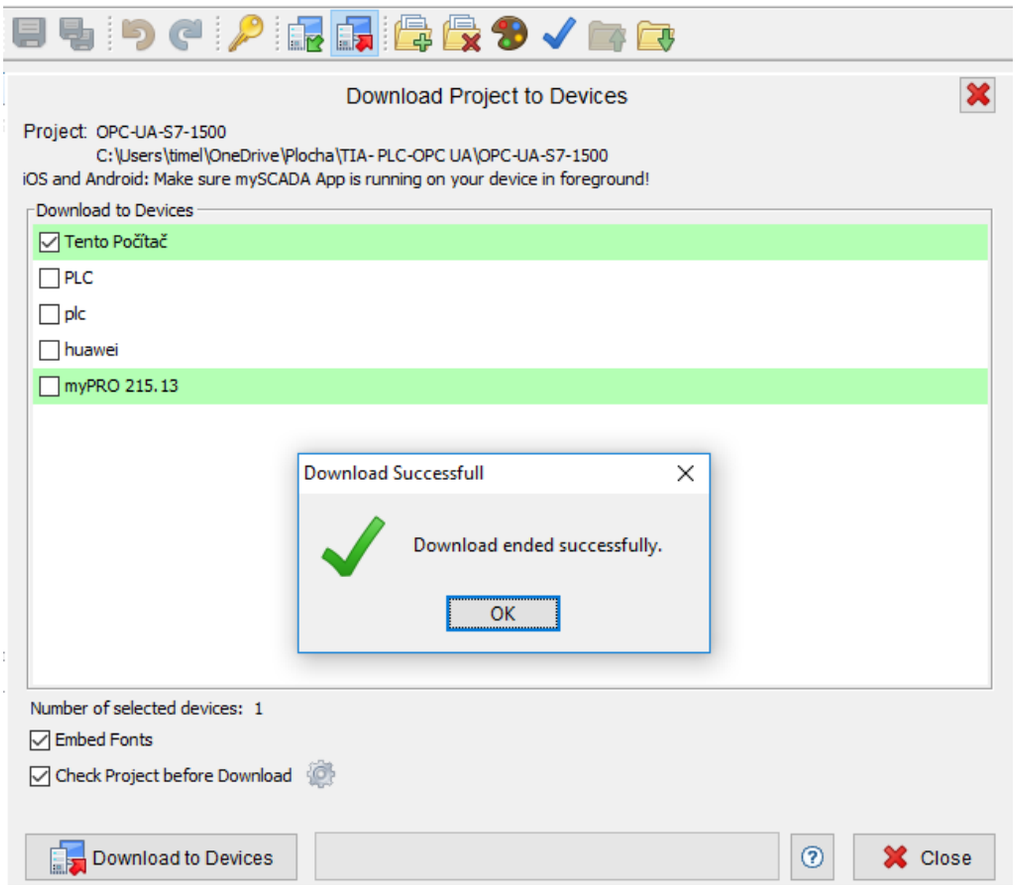
3.3.3 Náhled (View)

Nyní je potřeba vytvořit si nový *view*. Postup – views – pravým tlačítkem add new view. Poté si ho nastavíme, na které zařízení chceme zobrazení aplikovat. Já si ho nastavil na svůj počítač. Rozhodl jsem se, že budu demonstrovat OPC UA, tak že změřím hodnoty napětí na vstupu při chodu PLC v módu STOP a RUN. Tudíž nyní přiřadím voltmetr do mého projektu. Přiřadil jsem dva, a to následovně rozklikl jsem *components dialog (červený rámeček) – gauges* (měřidla), nyní si vybral dva voltmetry a oběma přiřadil příslušný tag, který jsem v předchozí kapitole 3.3.2 Tag database vytvořil viz. zelený rámeček.



Obrázek 24 – myScada view

Nyní již mám vše potřebné hotovo, stačí projekt uložit a nahrát do PC.



Obrázek 25 – myScada nahrání do PC

3.3.4 OPC UA – Měření napětí na vstupu

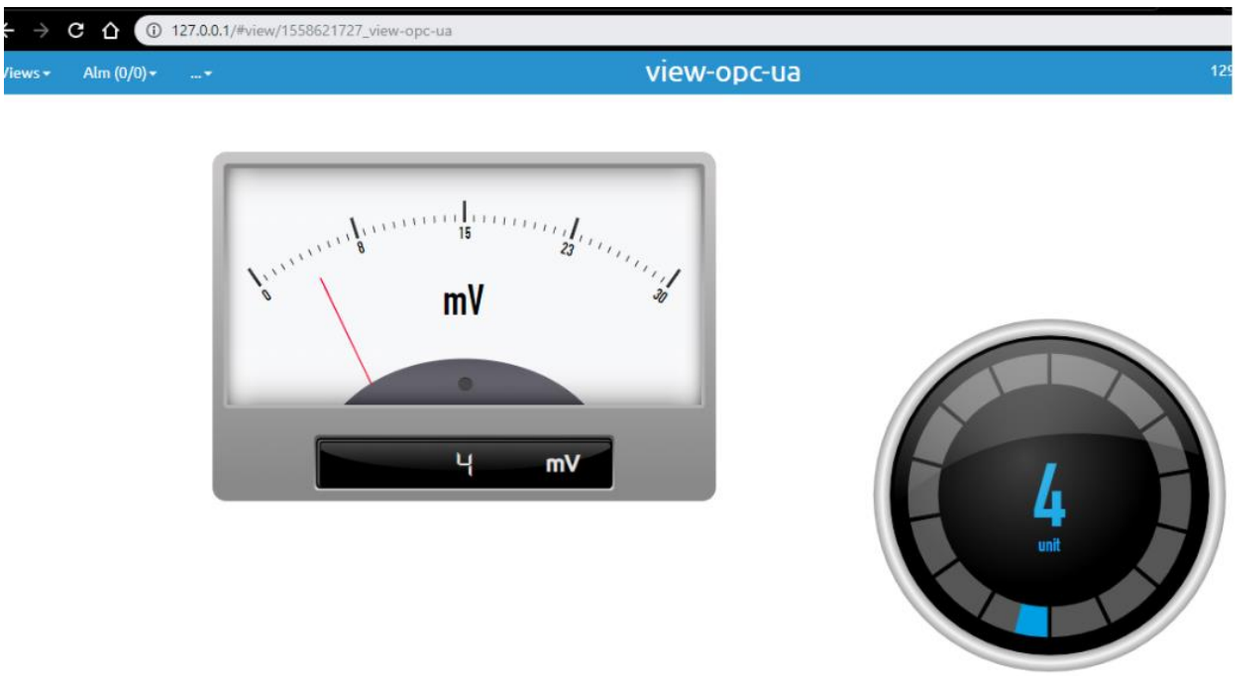
Jak jsem již zmiňoval v kapitole 3.3 Nastavení myScada, využiji tzv. webový přístup. K zobrazení mého projektu využiji webový prohlížeč. Otestované prohlížeče jsou:[23]

1. MS Edge

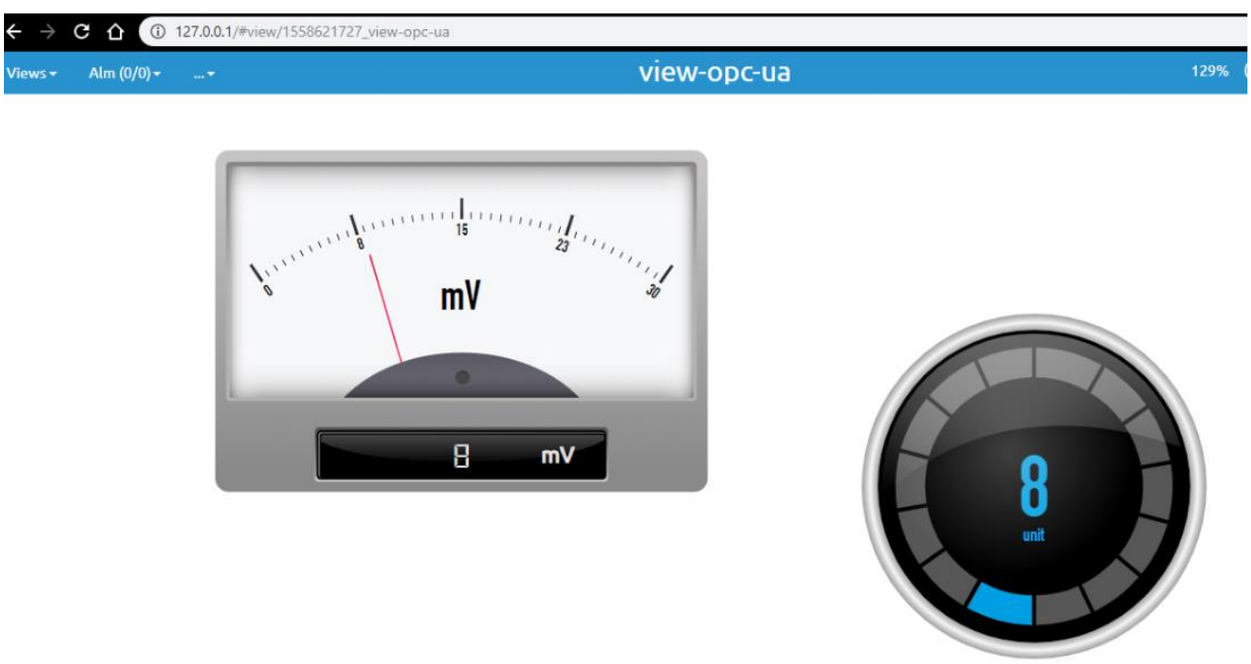
2. Chrome

3. Firefox

Nyní již otevřu webový prohlížeč a zadám adresu http://127.0.0.1 nebo https://localhost a zobrazí se mi můj *project view* z mydesigneru. S tím rozdílem, že již můžu vidět na voltmetru hodnotu napětí vstupu 4 mV ve STOP módu v reálném čase viz. obrázek 26. Poté přepnu PLC manuálně do RUN módu a nyní se změnila hodnota napětí vstupu na 8 mV, viz. obrázek 27. Celé toto připojení je v zabezpečení úrovně 1 viz. kapitola 2.3.7 Uživatelská práva pro přístup.



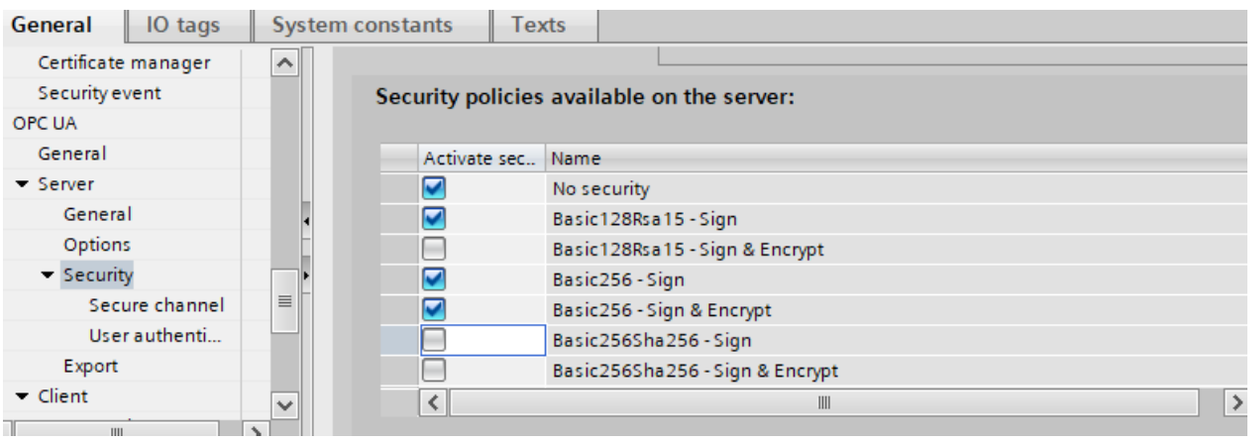
Obrázek 26 – STOP mód



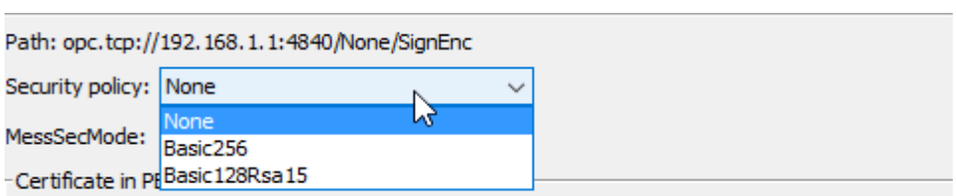
Obrázek 27 – RUN mód

3.4 OPC UA komunikace se zabezpečením

V této kapitole ukáži možnosti zabezpečení, které OPC UA server umožňuje. V předchozím nastavení jsem nevyužíval žádné zabezpečení, jednalo se o úroveň 1 viz. kapitola 2.3.7 Uživatelská práva pro přístup. Typ šifrování, které bude dostupné na serveru si můžeme nastavit následovně, postup-OPC UA – server – security, viz. obrázek 28. Zvolil jsem takové, které podporuje myScada viz. obrázek 29, více o tomto tématu v kapitole 2.3.4 Security policies, profiles.

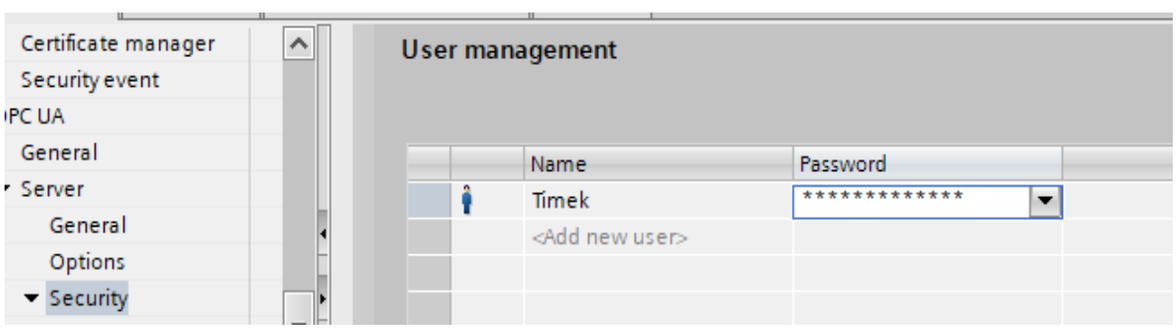


Obrázek 28 – TIA security policies



Obrázek 29 – my Scada security policies

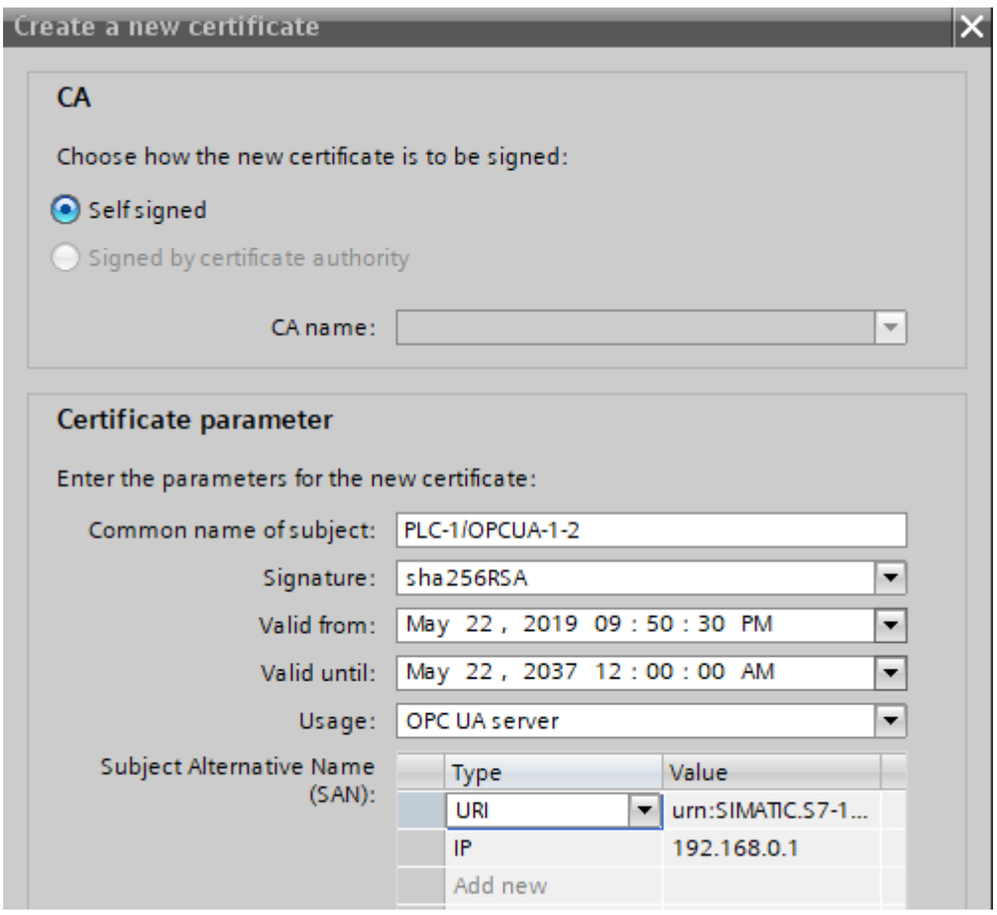
Nyní nastavím nového uživatele, který bude mít přístup k serveru.



Obrázek 30 – TIA – správa uživatelů

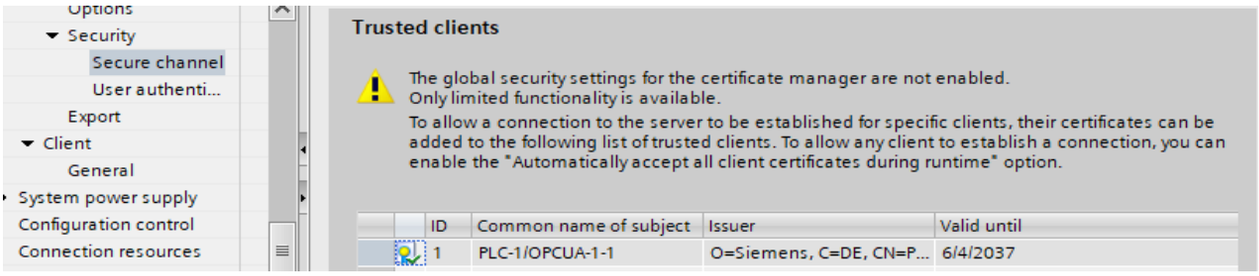
Poté jsem se již opět připojil přes myScadu stejně jako v předchozí kapitole. Toto nastavení je jedním z nejběžnějších a nejvíce využívaných. Reprezentuje úroveň 2 viz. kapitola 2.3.7 Uživatelská práva pro přístup

Další úrovni zabezpečení jsou certifikáty, více o tomto tématu se nachází v kapitole 2.3.6 OPC UA certifikáty. Já jsem si jeden vygeneroval. Můžeme nastavit například platnost certifikátu, já si ho nastavil, jak je uvedeno na obrázku. Postup – OPC UA – server – security – secure channel – server certificate – add new.



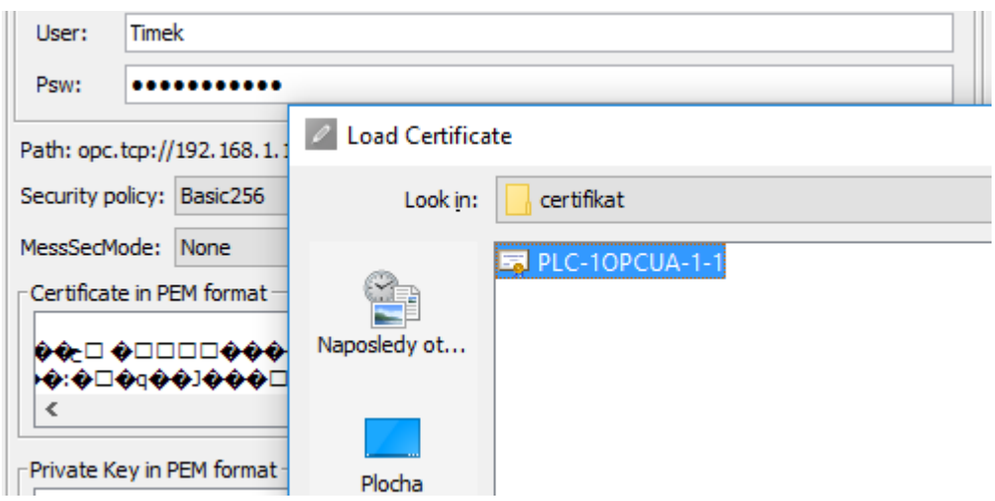
Obrázek 31 – TIA-certifikát

Poté ještě musím tento certifikát přiřadit do tzv. Trust listu viz. obrázek 32.



Obrázek 32 – TIA trust list

Nyní jsem již vše nutné vykonal a nyní tento certifikát pouze vyexportuji a nahraji do myScady a opět se přípojím.



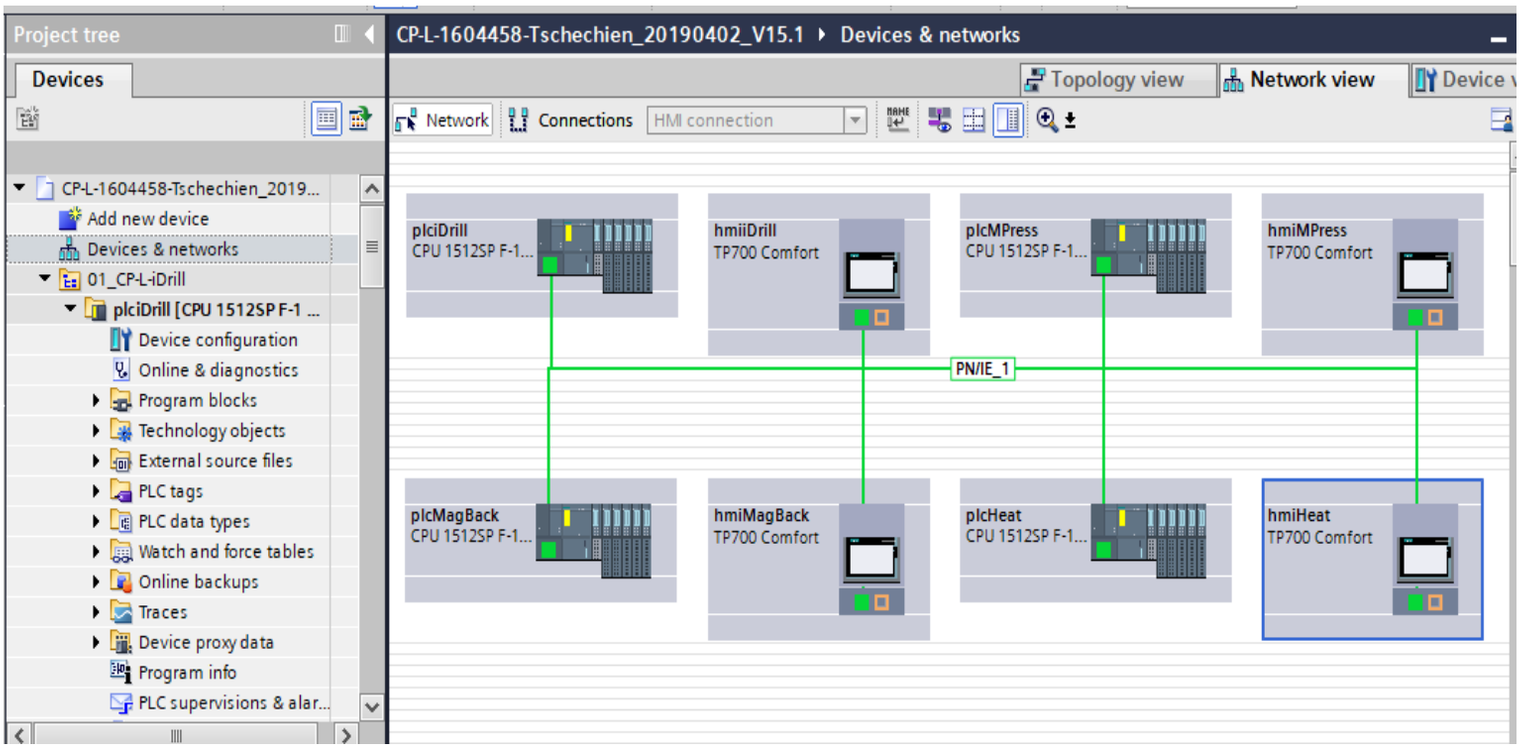
Obrázek 33 – myScada certifikát

Toto nastavení reprezentuje 4. úroveň. Klient může mít svůj vlastní certifikát, který chce využít. Administrátor ho pouze musí přiřadit do trust listu v nastavení serveru

3.5 OPC UA komunikace CP Factory / CP Lab

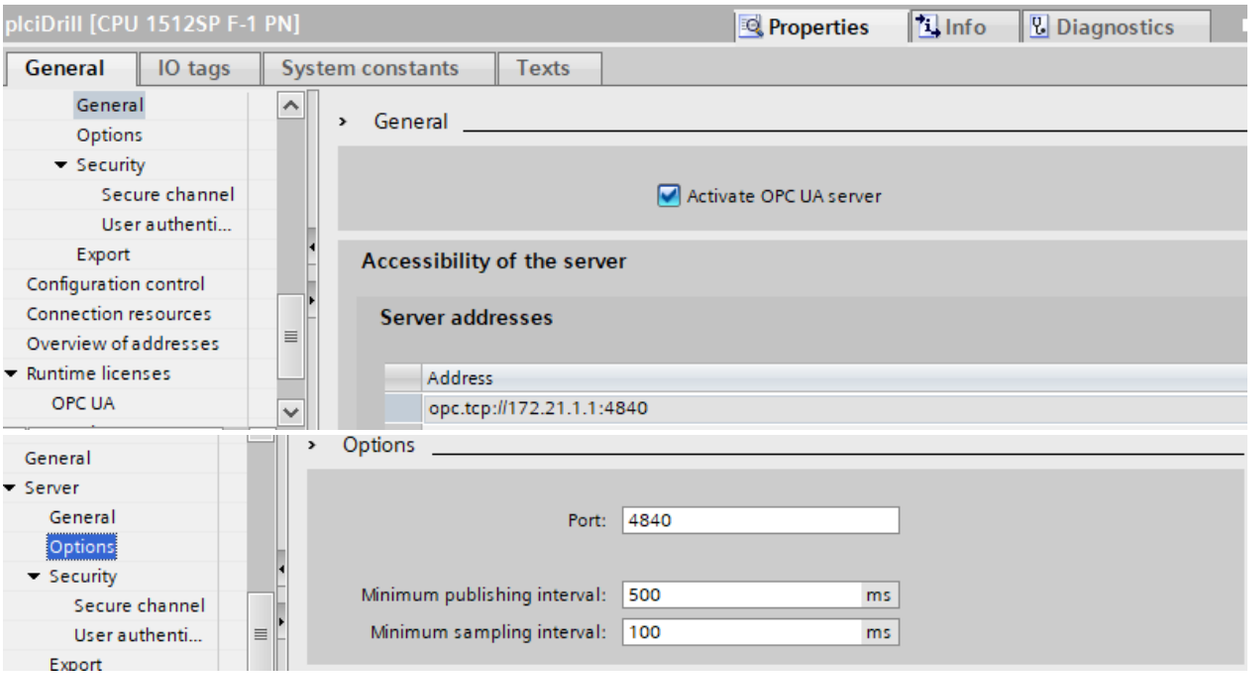
Mým posledním úkolem je popsat komunikaci na prototypu výrobní linky, která se nachází v laboratořích automatického řízení na fakultě strojní ČVUT v Praze. V prototypu výrobní linky na výrobu mobilů CP lab, se nachází čtyři PLC - S7-1512SP, které jsou propojeny se čtyřmi dotykovými obrazovkami TP – 700 Comfort přes profinet. *„PROFINET je součástí normy IEC 61158 a je založen na mezinárodním standardu Ethernet (IEEE 802.3). Jedná se tedy o „přepínaný" Fast Ethernet (100 Mbit/s). PROFINET používá odstupňovanou komunikační architekturu založenou na Ethernetu. Rozlišujeme tyto typy: standardní komunikace (TCP/IP), komunikace pro reálný čas (RT) a izochronní reálný čas (IRT)"* [25, str.4]. Každé PLC zastává funkci

jednoho stanoviště. Nachází se zde stanoviště, která jdou postupně za sebou –vrtání, podavač zadního krytu, lisování, pec. Veškerá komunikace mezi jednotlivými komponenty probíhá přes protokol OPC UA. NA obrázku 34 vidíme náhled do projektu v TIA portálu a propojení jednotlivých PLC a HMI (stanovišť) s ostatními.



Obrázek 34 – TIA Propojení PLC a HMI

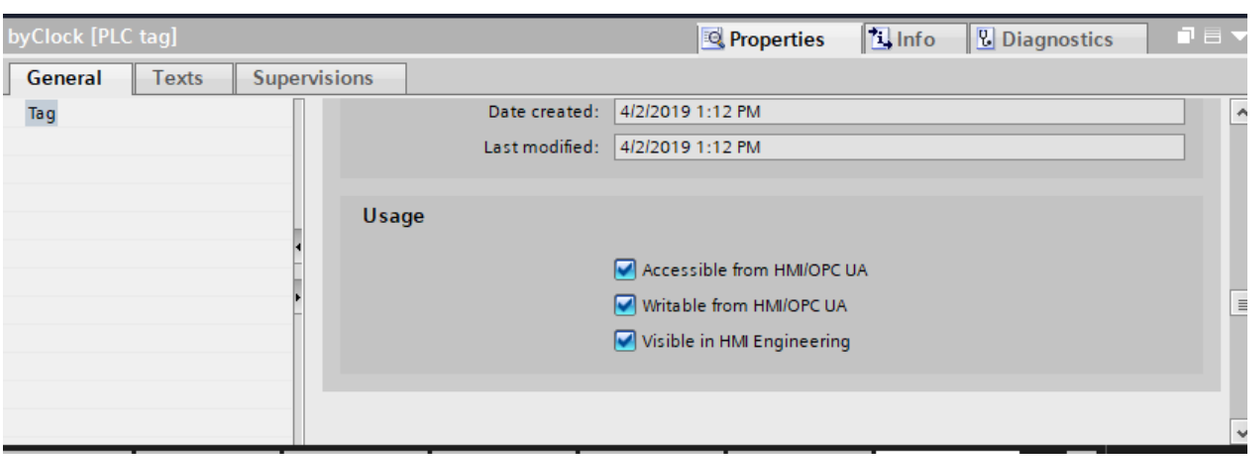
Nyní se podívám na nastavení jednotlivých kritérií OPC UA serveru. Nastavení OPC UA serveru je ve všech PLC a HMI stejné. Vybral jsem si pouze stanoviště vrtání a nastavení OPC UA na příslušném PLC a HMI. Základní nastavení OPC UA serveru můžeme vidět na obrázku 35. OPC UA server je samozřejmě aktivován. Ze serverové adresy vyčteme, že port je nastaven standardně 4840 stejně jako v mém nastavení viz. kapitola 3.2 Nastavení OPC UA serveru - TIA. To samé platí pro URL adresu opc.tcp://server, která nám značí, že využíváme binární protokol viz. kapitola 2 CO JE TO OPC UA? . Minimální interval vzorkování je nastaven na 100 m*s a minimální interval odesílání dat je nastaven na 500 m*s. Více o tomto tématu se nachází v kapitole 2.3.9 Subscription. Také je nastavená runtime licence large, více informací se nachází v kapitole 3.2 Nastavení OPC UA serveru - TIA.



Obrázek 35 – TIA nastavení OPC UA serveru

Nyní jsem prošel základní nastavení OPC UA serveru a mohu se podívat na zabezpečení serveru. To je velmi jednoduché, protože není nastaveno momentálně žádné zabezpečení, kdokoliv se může připojit k tomuto serveru. Tudíž je nastavena tzv. 1. úroveň (bez autentizace) viz. kapitola 2.3.7 Uživatelská práva pro přístup. Také je nastaven úplný přístup k PLC stejně jako v mém případě viz. obrázek Obrázek 19 - TIA - myScada nastavení.

Nakonec by bylo vhodné zmínit nastavení Tagů v projektu. Jak můžeme vidět na obrázku 35, jsou povolené možnosti přístupu, zápisu a programování v HMI, tedy můžeme s tagy manipulovat také v HMI.



Obrázek 36 – TIA nastavení tagů

ZÁVĚR

Cílem této práce bylo popsat protokol OPC UA, vyzkoušet protokol na komunikaci PLC – SCADA, nakonec zhodnotit a otestovat možnosti šifrované komunikace. Dodatečně popsat komunikaci na prototypu výrobní linky v cp lab. Tento cíl byl dle mého názoru úspěšně splněn.

V úvodu teoretické části jsem popsal, jak se protokol OPC vyvíjel a z čeho OPC UA vychází. Poté jsem se zaměřil na obecný popis a kde všude je možné protokol využít. Také jsem vypsal a vysvětlil několik důležitých pojmů, které jsou pro základní využití OPC UA nezbytné. Nejvíce pozornosti jsem věnoval zabezpečení a komunikačním prvkům.

Poté jsem se již věnoval samotnému připojení PLC a nastavení OPC UA serveru v TIA portálu, tak abych mohl komunikovat s programem myScada. Úspěšné propojení jsem demonstroval měřením napětí na vstupu PLC za pomoci virtuálního voltmetru, který jsem definoval v programu myScada. Poté jsem tento proces opakoval, ale s využitím bezpečnostních prvků, které OPC UA nabízí. Nakonec jsem stručně popsal komunikaci a nastavení OPC UA serveru v TIA portálu na prototypu výrobní linky.

Daná problematika je velmi rozsáhlá a dalo by se na ní úspěšně navázat větším projektem. Vypracováním mé bakalářské práce jsem získal mnoho nových znalostí, tudíž považuji práci za přínosnou a mohu se dále rozvíjet v daném oboru.

Seznam použité literatury

[1] *OPC and OPC UA explained* [online]. [vid. 2019-05-05]. Dostupné z: https://www.novotek.com/en/solutions/kepware-communication-platform/opc-and-opc-ua-explained

[2] *OPC Foundation* [online]. 2019 [vid. 2019-05-06]. Dostupné z: https://en.wikipedia.org/w/index.php?title=OPC_Foundation&oldid=880290352

[3] Home Page. *OPC Foundation* [online]. [vid. 2019-05-06]. Dostupné z: https://opcfoundation.org/

[4] SCHLEIPEN, Miriam, Syed-Shiraz GILANI, Tino BISCHOFF a Julius PFROMMER. OPC UA & Industrie 4.0 - Enabling Technology with High Diversity and Variability. *Procedia CIRP* [online]. 2016, 57, 315–320 [vid. 2019-05-01]. ISSN 22128271. Dostupné z: https://linkinghub.elsevier.com/retrieve/pii/S2212827116312094

[5] *NET Based OPC UA Client/Server SDK: OPC Introduction* [online]. [vid. 2019-05-06]. Dostupné z: http://documentation.unified-automation.com/uasdkdotnet/2.5.4/html/L1OpcIntroduction.html

[6] Unified Architecture. *OPC Foundation* [online]. [vid. 2019-05-01]. Dostupné z: https://opcfoundation.org/about/opc-technologies/opc-ua/

[7] *Komunikace přes rozhraní OPC UA* [online]. [vid. 2019-05-05]. Dostupné z: https://www.promotic.eu/cz/pmdoc/Subsystems/Comm/OPC/OPCUA.htm

[8] MAGÁTH, Marek. *OPC UA klient pro laboratorní model „Třídička beden"*. Brno, 2018. BAKALÁŘSKÁ PRÁCE. VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ.

[9] BALDA, Pavel. Úvod do OPC Unified Architecture. In: [online]. 2006, s. 11 [vid. 2019-05-02]. Dostupné z: https://vendulka.zcu.cz/Download/Free/IRS1/index.php?dir=&file=IRS1_12_OPC_UA.pdf

[10] *What is OPC UA and why will it continue to grow in use? | Exor International* [online]. 17. leden 2019 [vid. 2019-05-08]. Dostupné z: https://exorint.com/2019/01/17/what-is-opc-ua-and-why-will-it-continue-to-grow-in-use/

[11] OPC UA speeds up the digitalization. *siemens.com Global Website* [online]. [vid. 2019-05-08]. Dostupné z: https://new.siemens.com/global/en/products/automation/industrial-communication/opc-ua.html

[12] *There Is No Industrie 4.0 without OPC UA – OPC Connect* [online]. [vid. 2019-05-09]. Dostupné z: https://opcconnect.opcfoundation.org/2017/06/there-is-no-industrie-4-0-without-opc-ua/

[13] *API* [online]. 2018 [vid. 2019-05-09]. Dostupné z: https://cs.wikipedia.org/w/index.php?title=API&oldid=16058310

[14] OPC UA Overview. *Real Time Automation, Inc.* [online]. [vid. 2019-05-01]. Dostupné z: https://www.rtautomation.com/technologies/opcua/

[15] *OPCClassicVSUA.pdf* [online]. [vid. 2019-05-19]. Dostupné z: http://www.dsinteroperability.com/OPCClassicVSUA.pdf

[16] *main.pdf* [online]. [vid. 2019-06-01]. Dostupné z: http://home.zcu.cz/~honza801/dp/main.pdf

[17] *teps-02.pdf* [online]. [vid. 2019-06-01]. Dostupné z: http://www.ped.muni.cz/wtech/03_studium/teps/teps-02.pdf

[18] SHI-WAN LIN (THINGSWISE). Architecture Alignment and Interoperability [online].[vid. 2019-05-25]. Dostupné z https://www.iiconsortium.org/pdf/JTG2_Whitepaper_final_20171205.pdf

[19] MAHNKE, Wolfgang, Stefan-Helmut LEITNER a Matthias DAMM. *OPC unified architecture*. Berlin: Springer, 2009. ISBN 978-3-540-68898-3.

[20] *Transport Layer Security* [online]. 2019 [vid. 2019-05-18]. Dostupné z: https://cs.wikipedia.org/w/index.php?title=Transport_Layer_Security&oldid=17229283

[21] *PKI* [online]. 2013 [vid. 2019-05-18]. Dostupné z: https://cs.wikipedia.org/w/index.php?title=PKI&oldid=9856843

[22] *SIMATIC OPC UA - TIA Portal - Siemens* [online]. [vid. 2019-06-02]. Dostupné z: http://w3.siemens.com/mcms/automation-software/en/tia-portal-software/step7-tia-portal/simatic-step7-options/opc-ua-s7-1500/pages/default.aspx

[23] mySCADA TIA portal connector. *mySCADA Technologies* [online]. [vid. 2019-06-02]. Dostupné z: https://www.myscada.org/mydesigner-manual/

[24] OPC UA driver. *mySCADA Technologies* [online]. [vid. 2019-06-02]. Dostupné z: https://www.myscada.org/mydesigner-manual/

[25] *profinet_04_2005_cz.pdf* [online]. [vid. 2019-06-11]. Dostupné z: http://stest1.etnetera.cz/ad/current/content/data_files/automatizacni_systemy/prumyslova_komunikace/profinet/profinet_04_2005_cz.pdf

Seznam použitých zkratek

SCADA - Supervisory Control And Data Acquisition

PLC - Programmable Logic Controller

OPC UA - Open Platform Communications Unified Architecture

OPC - OLE for control process

OLE - Object Linking and Embedding

COM - Component Object Model

DCOM - Distributed Component Object Model

TCP/IP - Transmission Control Protocol/Internet Protocol

HTTP - Hypertext Transfer Protocol

XML - Extensible Markup Language

API - Application Programming Interface

SOAP - Simple Object Access Protocol

SW – Software

SPKI - Simple Public Key Infrastructure

PGP - Pretty Good Privacy

TLS – Transport Layer Security

WS – Web Services

PKI – Public Key Infrastructure

CA - Certification Authority

RA – Registration Authority

ISO - International Organization for Standardization

OSI - Open system interconnection

HMI – Human machine interface

URL - Uniform Resource Locator

Seznam obrázků

Seznam tabulek