

Uživatelská příručka

Software pro generování datového provozu pro
monitorovací a řídicí systémy TSO na protokolech ICCP-
TASE.2 a IEC 60870-5-101

Software byl vyvinut v rámci projektu „Národní centrum kompetence pro kyberbezpečnost“ č. TN01000077 a byl spolufinancován se státní podporou Technologické agentury v ČR v rámci programu Národní centra kompetence 1.

Obsah¹

1.	Zprovoznění testovacího polygonu	3
1.1	Kompilace MZ Automation ICCP/TASE.2 knihovny	3
1.2	Kompilace zdrojových souborů generátoru Apache JMeter	3
2.	Grafické uživatelské rozhraní a ovládání generátoru	4
3.	Testovací scénáře	6
3.1	PUSH scénář	6
3.2	PULL scénář	6
4.	Popis komunikace protokolu TASE.2	8
4.1	Inicializace TASE.2 komunikace	8
4.2	Komunikace skrze Information Message	8
4.3	Komunikace typu PULL	9
4.4	Komunikace typu PUSH	9
5.	Zprovoznění testovacího polygonu	11
5.1	Instalace knihovny Real Thoughts COMPROTware: Library pro IEC 60870-5-101	11
5.2	Kompilace zdrojových souborů generátoru Apache JMeter	11
5.3	Zprovoznění zkompilevané verze Jmeter	12
6.	Grafické uživatelské rozhraní a ovládání generátoru IEC101	13
7.	Testovací slave scénář IEC 60870-5-101	15
8.	Popis komunikace protokolu IEC101	16
8.1	Schéma komunikace	16
8.2	Inicializace IEC101 komunikace	16
8.3	Průběh IEC101 komunikace	17
8.4	Zpracování dat z komunikace IEC101	17

¹ Verze manuálu: 06/2021

1. Zprovoznění testovacího polygonu

V následujícím textu budou popsány prerekvizity nutné k úspěšnému nastavení a spuštění vytvořeného generátoru. Jedná se zejména o kompilaci knihovny ICCP/TASE.2 IEC 60870-6 od společnosti MZ Automation a kompilaci zátěžového testeru Apache JMeter s rozšířením pro generování dat protokolu TASE.2.




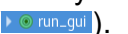
1.1 Kompilace MZ Automation ICCP/TASE.2 knihovny

Kompilace knihovny v systému Linux je poměrně jednoduchá a sestává se pouze z několika kroků. Je však nutné splnit alespoň minimální požadavky, tj. Cmake alespoň ve verzi 3.0. Po stažení knihovny ze stránek výrobce rozbalte archiv na vámi vybrané místo.

1. Přesuňte se do složky s rozbaleným archivem a vytvořte podsložku **build**.
2. Vytvořte konfigurační soubor pro kompilaci zavoláním příkazu **cmake ..**
3. Zkompiluje knihovnu zadáním příkazu **make**
4. Po úspěšné kompilaci pak už stačí pouze soubory **libtase2.a**, **libtase2.so** a **libtase2.so.x.x** nakopírovat do složky **/usr/lib**

1.2 Kompilace zdrojových souborů generátoru Apache JMeter

Pro správnou kompilaci programu JMeter je nutná instalace Java JDK alespoň ve verzi 8. Dále je pro vývoj vhodné využít program Eclipse, který je volně dostupný bez nutnosti instalace. Pro správný import vytvořeného zásuvného modulu je nutné postupovat následovně.

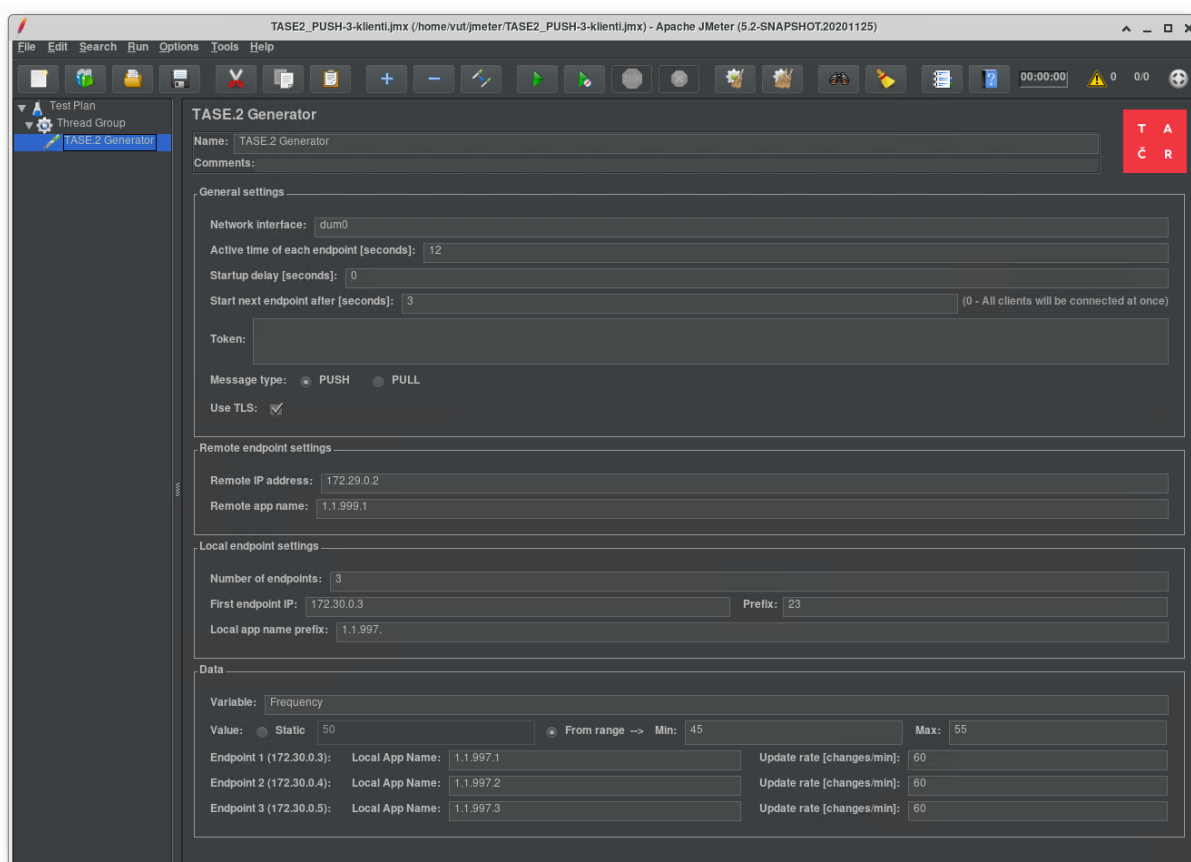
1. Ze serveru (Odkaz na Git) stáhněte zdrojové soubory programu JMeter obsahující také vytvořená TASE2 modul.
2. Otevřete aplikaci Eclipse a z nabídky *File->Open Projects from File System* zvolte možnost *Directory* a vyberte složku se staženými zdrojovými kódy. Program Eclipse by měl ve složce automaticky najít správný projekt. Poté stačí jen zmáčknout tlačítko *Finish*.
3. Nyní je nutné nastavit Ant build file umožňující spuštění kompilačního procesu. Nejprve je však nutné přepnout Eclipse do rozvržení Ant. To provedete skrze hlavní nabídku *Window->Show View->Ant*. Nyní by se měla v pravé části monitoru zobrazit nabídka pro prostředí Ant. Kliknutím na tlačítko *Add Buildfiles (*)* se otevře nabídka pro s volbou souboru. Z této nabídky zvolte soubor *Build.xml*.
4. Nyní je nutné nainstalovat všechny potřebné závislosti. To lze provést spuštěním akce dostupném v nabídce ant příkazů jako *download_jars* (). Akci můžete spustit dvojklikem nebo označením položky a stisknutím tlačítka (). Po úspěšném provedení akce je nutné knihovny přidat do cety projektu. To lze provést skrze hlavní nabídku *Project->Properties* a položku *Java Build Path*. Zde skrze tlačítko *Add JARs...* vyberte všechny *.jar soubory ve složce *lib*.
5. Nyní je již možné projekt úspěšně zkompilovat i s modulem TASE2. Nejprve je nutné spustit ant proceduru *install-tase2* () z nabídky ant. Poté je již možné spustit kompilaci samotného JMeter GUI skrze *run_gui* ().

2. Grafické uživatelské rozhraní a ovládání generátoru

Pro ovládání generátoru provozu TASE.2 byl vyvinut nový modul do programu Apache JMeter. Prostřednictvím tohoto modulu lze nastavovat jednotlivé parametry zasílaných zpráv a následně generovaný provoz spouštět či zastavovat pomocí standardních komponent programu JMeter.

Grafické rozhraní JMeteru s modulem TASE.2 je zobrazeno na Obrázek 1. Rozhraní modulu je rozděleno do čtyř celků (General settings, Remote endpoint settings, Local endpoint settings, Data), ve kterých je možné nastavit jednotlivé parametry generovaného provozu. V první oblasti General settings lze nastavit tyto parametry:

- Network interface – síťové rozhraní, ze kterého bude proudit vygenerovaný datový tok,
- Active time of each endpoint – doba, po kterou je každý lokální endpoint aktivní (tzn. odesílá zprávy v případě metody PUSH, nebo naslouchá příchozím žádostem v případě metody PULL),
- Startup delay – počáteční zpoždění spuštění generátoru,
- Start next endpoint – časový rozestup vytváření dalších TASE.2 endpointů (pokud je hodnota 0, všechny endpointy se vytvoří najednou),
- Token – token pro vzdálený TASE.2 endpoint,
- Message type – typ generovaných TASE.2 zpráv (PUSH nebo PULL)
- Use TLS – zabezpečení komunikace protokolem TLS. Všechny potřebné certifikáty se automaticky načtou z domovského adresáře aplikace Jmeter.



Obrázek 1: Grafické rozhraní generátoru

V další sekci Remote endpoint setting je nutné specifikovat parametry vzdáleného endpointu, konkrétně jeho IP adresu a jméno aplikace. Tyto parametry je nutné upravit pro navázání komunikace se správným protějším TASE.2 endpointem.

V další sekci Local endpoint settings je možné nastavit množství, IP adresy a jména aplikací lokálních endpointů, které budou v rámci tohoto generátoru generovat výsledný datový tok. Parametr Number of endpoints udává množství lokálních endpointů, které mají být vytvořeny. Parametrem First endpoint IP lze přesně specifikovat IP adresu prvního endpointu. Pokud je parametrem Number of endpoints nastaveno více endpointů, tak se IP adresa dalších endpointů postupně inkrementuje od IP adresy prvního endpointu. Podobně se pro jednotlivé endpointy nastavuje i jméno aplikace. Parametrem Local app name prefix se nastaví prefix jména aplikace, za které se pro každý endpoint automaticky připojí unikátní číslo. V případě více endpointů je toto číslo opět inkrementováno.

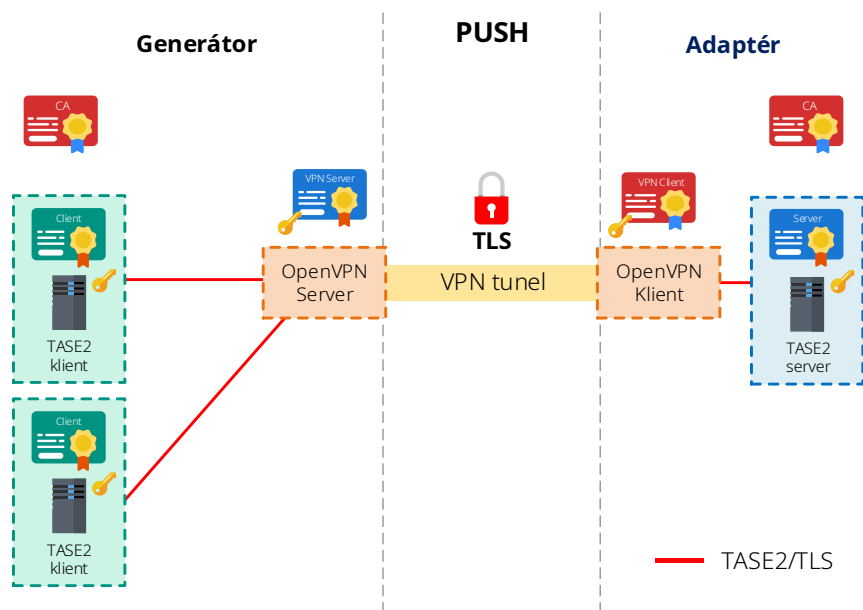
V poslední sekci Data lze specifikovat obsah a frekvenci zasílaných zpráv. Parametrem Variable lze specifikovat název přenášené proměnné. Generátor byl od počátku zamýšlen k zasílání aktuální hodnoty frekvence elektrické sítě, proto je zde předvyplněn text "Frequency". Parametrem Value lze nastavit, jaká hodnota frekvence má být zasílána. Je možné zvolit statické číslo, nebo náhodné číslo z definovaného rozsahu. Ve zbývajících částech této sekce je možné upravit jméno aplikace a specifikovat, jak často mají endpointy měnit a zasílat hodnotu frekvence (v případě metody PUSH), nebo jak často se má hodnota frekvence pro konkrétní endpoint měnit (v případě PULL metody).

3. Testovací scénáře

3.1 PUSH scénář

V tomto případě vystupuje JMeter v roli aktivních endpointů. Po spuštění testovacího scénáře vytvoří JMeter 1 až N instancí aktivních klientů TASE.2, které se po vytvoření připojují k pasivnímu endpointu (serveru) na straně Adaptéru. Aktivní klienti pak svůj stav pravidelně aktualizují dle nastavené periody. Každá změna hodnoty je propagována přes metodu Update Online Value, která na server odešle objekt Transport Data Set obsahující novou hodnotu. Přenášená hodnota obsahuje také rozšířenou formu časové značky UNIX (počet mikrosekund od 1. 1. 1970). Jelikož jsou data na serveru přijímána skrze callback volání (mimo hlavní smyčku) ihned po přijetí objektu, je možné i přesné určení doby zpoždění z rozdílu doby přijetí a odeslání. Po skončení testu se všichni aktivní klienti ze serveru odhlásí a uvolní přidělené IP adresy.

Komunikace typu PUSH využívá pouze jeden TASE2 server (pasivní endpoint) na protější straně, viz obrázek 2. Klientskou část zde zastupuje generátor JMeter. Jelikož v tomto scénáři vystupuje pouze jeden TASE2 server, je distribuce certifikátů výrazně ulehčena. Všechny stanice totiž musí být vybaveny certifikáty vydané stejnou certifikační autoritou. Je tak možné sdílet certifikát mezi všemi klientskými stanicemi, což je i případ aktuální implementace. Druhou možností je pak vygenerování certifikátů pro každého klienta. Avšak pro tento účel je stále nutné použít stejnou certifikační autoritu.



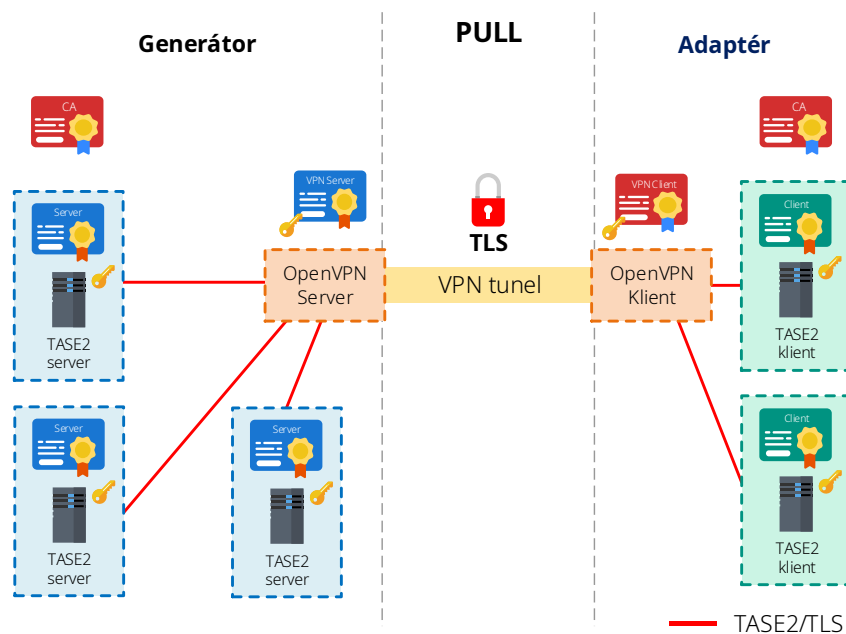
Obrázek 2: Struktura PUSH komunikace

3.2 PULL scénář

Zde JMeter vystupuje v roli pasivních endpointů (serverů) ke kterým se připojují aktivní endpointy z generátoru. Jelikož generátor před spuštěním testu nemá informace o počtu aktivních endpointů, je nutné tyto informace doručit skrze boční kanál. Pro tento účel je na straně generátoru vytvořen REST endpoint očekávající na vstupu JSON objekt s informacemi o počtu aktivních klientů, periodě vyčítání, názvu přenášené proměnné a IP adresu serveru pro připojení. Jelikož použitý REST endpoint vyžaduje autentizaci, obsahuje hlavička HTTP dotazu také pole s autentizačním tokenem.

Po spuštění testu pak jednotlivé servery aktualizují své příslušné proměnné. Změna stavu se v tomto případě přímo nedistribuuje, ale je pravidelně vyčítána z aktivních endpointů. Při každém čtení se aktivní endpoint nejprve připojí a po úspěšném vyčtení hodnoty je spojení opět uzavřeno. Jelikož v tomto případě není zaručena synchronizace mezi změnou hodnoty a jejím vyčtením, není bohužel možné jednoduché zjištění přenosu při zpoždění. Pokud je odečtena doba přijetí zprávy a časová značka z přijaté zprávy, výsledkem je pouze doba mezi přijetím zprávy a změnou hodnoty proměnné. Stejně jako v případě PUSH komunikace jsou po skončení testu přidělené adresy serverů uvolněny.

V komunikaci typu PULL zajišťuje generátor JMeter služby TASE2 serverů (viz obrázek 3), ke kterým se připojují TASE2 klienti. Z důvodu zjednodušené distribuce jsou certifikáty mezi jednotlivými stanicemi sdíleny. Certifikáty jsou sdíleny vždy mezi všemi servery nebo klienty. Je tak možné se připojit z libovolného klienta na libovolný server. Je však možné nastavit každé stanici také unikátní certifikát. V případě, že jsou certifikáty vydány stejnou certifikační autoritou, je navázání zabezpečené komunikace stále možné. Poslední možností jsou certifikáty podepsané odlišnými certifikačními autoritami, zde však nastává problém s distribucí. Navázání zabezpečené komunikace je možné vždy jen mezi stanicemi, které obsahují pár server-klient certifikátů vydaného stejnou certifikační autoritou. Tato funkcionality není aktuálně implementována a vyžadovala by vytvoření sekundárního kanálu (REST API) umožňujícího nastavení certifikátů jednotlivým stanicím.



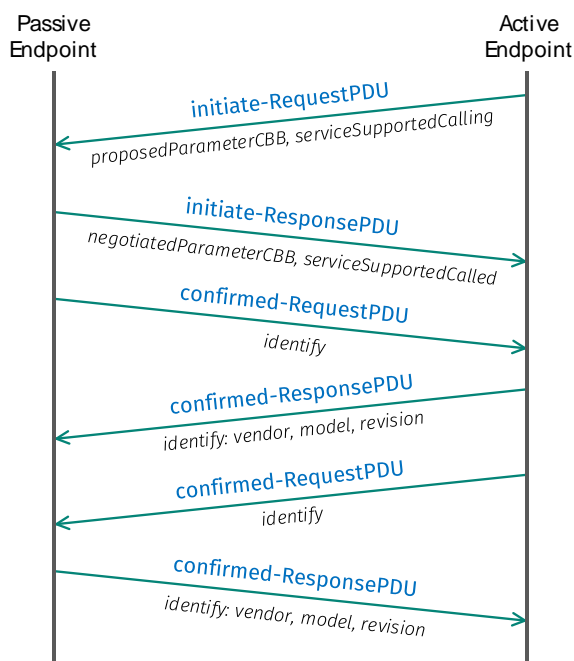
Obrázek 3: Struktura PULL komunikace

4. Popis komunikace protokolu TASE.2

Scénář pro popis komunikace protokolu TASE.2 se sestává z dvou koncových bodů (EndPoints), které obsahují klientskou i serverovou část protokolu. EndPoint 1 je v pasivní roli, čeká tedy na příchozí spojení. Naopak EndPoint 2 představuje aktivní stanici, která se připojuje k EndPoint 1. Demonstrační scénář obsahuje ukázky inicializace komunikace mezi oběma entitami, přenos zpráv skrze Information Message, PULL (metoda Read Point) a PUSH (metoda Transfer Data Set).

4.1 Inicializace TASE.2 komunikace

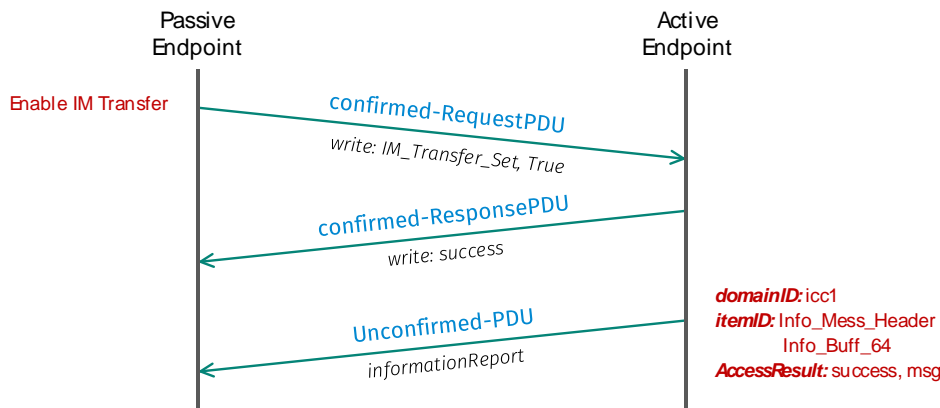
Nejprve je nutné spustit EndPoint 1, který naslouchá na příchozí spojení od EndPoint 2. Celé spojení začíná vysláním inicializačního požadavku na EndPoint 1. Tento požadavek obsahuje informace o navržených CBB (Conformance Building Block) a podporovaných volání služeb (getStatus, read, write, . . .). Na tuto zprávu EndPoint 1 odpoví vyjednanými parametry a podporovanými službami. V poslední fázi už probíhá pouze vzájemná identifikace klientů obsahující informace o výrobci, modelu a revizi. Tato komunikace je znázorněna na obrázku Obrázek 4.



Obrázek 4: Komunikace při připojení zařízení TASE.2

4.2 Komunikace skrze Information Message

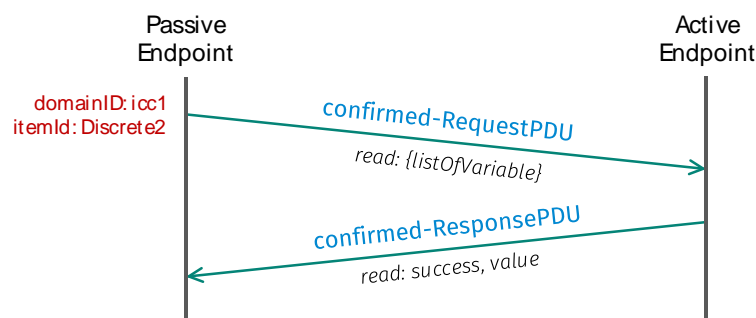
V tomto případě odesílá EndPoint 2 zprávu do EndPoint 1 s využitím služby Information Message. První krok procesu však zahajuje EndPoint 1, který odešle požadavek na povolení IM přenosu na EndPoint 2. Ten poté EndPoint 1 informuje o úspěšné změně nastavení. EndPoint 2 poté může odesílat informační zprávy v libovolný okamžik a EndPoint 1 je úspěšně přijímá. Zprávy jsou odesílány v doméně icc1, samotná zpráva pak obsahuje hlavičku Info_Mess_Header a buffer s vlastní zprávou, hexadecimálně kódovanou. Průběh odeslání Information Message znázorňuje obrázek Obrázek 5.



Obrázek 5: Průběh odeslání informační zprávy TASE.2

4.3 Komunikace typu PULL

V tomto případě vyčítá EndPoint 1 hodnotu z EndPoint 2. EndPoint 1 zahájí komunikaci vytvořením požadavku obsahující seznam žádaných proměnných (v tomto případě Discrete2). EndPoint 2 pak přímo odpoví v následné odpovědi obsahující hodnoty žádaných proměnných. Tato komunikace, znázorněná na obrázku Obrázek 6, odpovídá komunikačnímu model PULL.

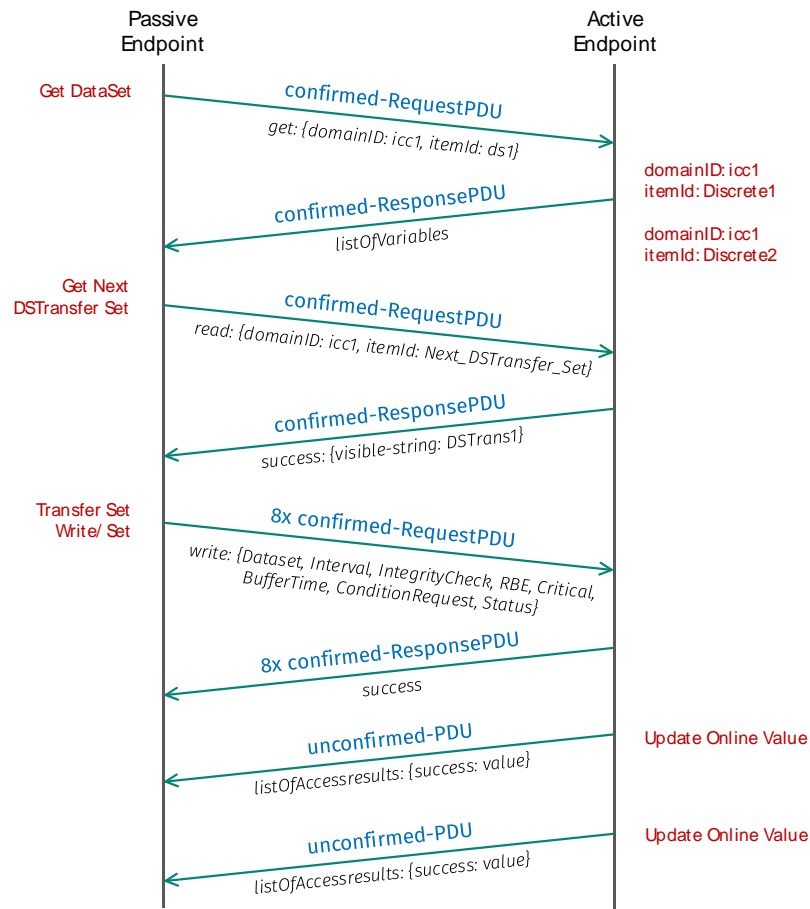


Obrázek 6: Komunikace typu PULL

4.4 Komunikace typu PUSH

V tomto případě EndPoint 1 zahajuje komunikaci odesláním požadavku na získání dostupných proměnných v doméně icc1 v data setu ds1. EndPoint 2 ve své odpovědi odešle seznam dostupných veličin (v tomto případě je využita pouze proměnná s názvem Discrete1). EndPoint 1 si v následující zprávě vyžádá další DSTransfer Set z domény icc1. EndPoint 2 pak ve své odpovědi odešle odpovídající DSTransfer Set (v tomto případě DSTrans1). EndPoint 1 poté odešle 8 požadavků na nastavení parametrů DSTransfer Setu na EndPoint 2 (mezi nimi i četnost vyčítání hodnot). Na všechny tyto zprávy následně EndPoint 2 odpoví zprávou Success.

Proměnná Discrete1 je každé 2 sekundy s inkrementována o hodnotu 1. Aby se tato změna projevila, tak je nutné hodnotu propagovat přes metodu Update Online Value. V tomto okamžiku je nová hodnota odeslána z EndPoint 2 do EndPoint 1. Z EndPoint 1 je úspěšné přijetí zprávy potvrzeno pouze na transportní vrstvě skrze TCP ACK.



Obrázek 7: Komunikace typu PUSH

5. Zprovoznění testovacího polygonu

V následujícím textu budou popsány prerekvizity nutné k úspěšnému nastavení a spuštění vytvořeného generátoru. Jedná se zejména o instalaci knihovny IEC 60870-5-101 od společnosti Real Thoughts a kompilaci zátěžového testeru Apache JMeter.


5.1 Instalace knihovny Real Thoughts COMPROTware: Library pro IEC 60870-5-101

Knihovna od společnosti Real Thoughts je již distribuována ve zkompilevané formě, proto je nutná pouze správná instalace. Avšak tento krok vyžaduje jisté prerekvizity pro ověření platnosti použitého licenčního klíče.

1. Ze stránek WIBU Systems ([URL](#)) je nutné stažení ovladače pro ověření licenčních klíčů.
2. Stažený WibuKey runtime je nutné nainstalovat do systému (pro systém Cent OS lze využít příkazu **sudo yum install <název souboru>.rpm**).
3. Soubor **CPLB64SwKey.lib** z kořenové složky knihovny je nutné nakopírovat do složky **/usr/lib**.

5.2 Kompilace zdrojových souborů generátoru Apache JMeter

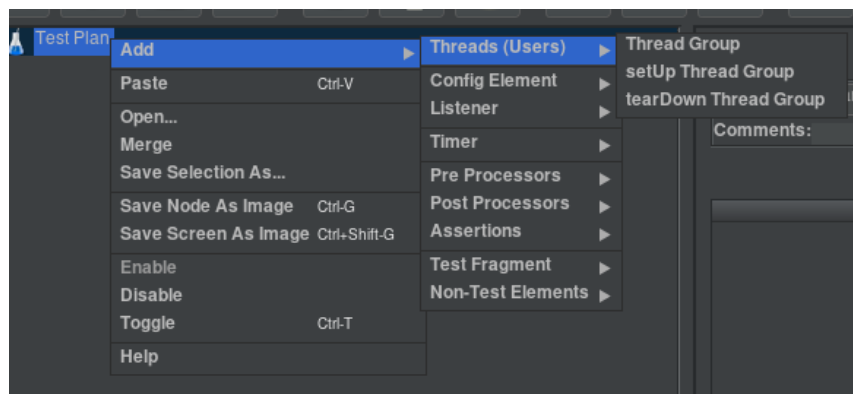
Pro správnou kompilaci programu JMeter je nutná instalace Java JDK alespoň ve verzi 8. Dále je pro vývoj vhodné využít program Eclipse, který je volně dostupný bez nutnosti instalace. Pro správný import vytvořeného zásuvného modulu je nutné postupovat následovně.

6. Ze serveru (Odkaz na Git) stáhněte zdrojové soubory programu JMeter obsahující také vytvořený IEC101 modul.
7. Otevřete aplikaci Eclipse a z nabídky *File->Open Projects from File System* zvolte možnost *Directory* a vyberte složku se staženými zdrojovými kódy. Program Eclipse by měl ve složce automaticky najít správný projekt. Poté stačí jen zmáčknout tlačítko *Finish*.
8. Nyní je nutné nastavit Ant build file umožňující spuštění kompilačního procesu. Nejprve je však nutné přepnout Eclipse do rozvržení Ant. To provedete skrze hlavní nabídku *Window->Show View->Ant*. Nyní by se měla v pravé části monitoru zobrazit nabídka pro prostředí Ant. Kliknutím na tlačítko *Add Buildfiles (*)* se otevře nabídka pro s volbou souboru. Z této nabídky zvolte soubor *Build.xml*.
9. Nyní je nutné nainstalovat všechny potřebné závislosti. To lze provést spuštěním akce dostupném v nabídce ant příkazů jako *download_jars* ([download_jars](#)). Akci můžete spustit dvojklikem nebo označením položky a stisknutím tlačítka (). Po úspěšném provedení akce je nutné knihovny přidat do cety projektu. To lze provést skrze hlavní nabídku *Project->Properties* a položku *Java Build Path*. Zde skrze tlačítko *Add JARs...* vyberte všechny *.jar soubory ve složce *lib*.
10. Nyní je již možné projekt úspěšně zkompileovat i s modulem IEC101. Nejprve je nutné spustit ant proceduru *install-iec101* ([install-iec101](#)) z nabídky ant. Poté je již možné spustit kompilaci samotného JMeter GUI skrze *run_gui* ([run_gui](#)).

5.3 Zprovoznění zkompilované verze Jmeter

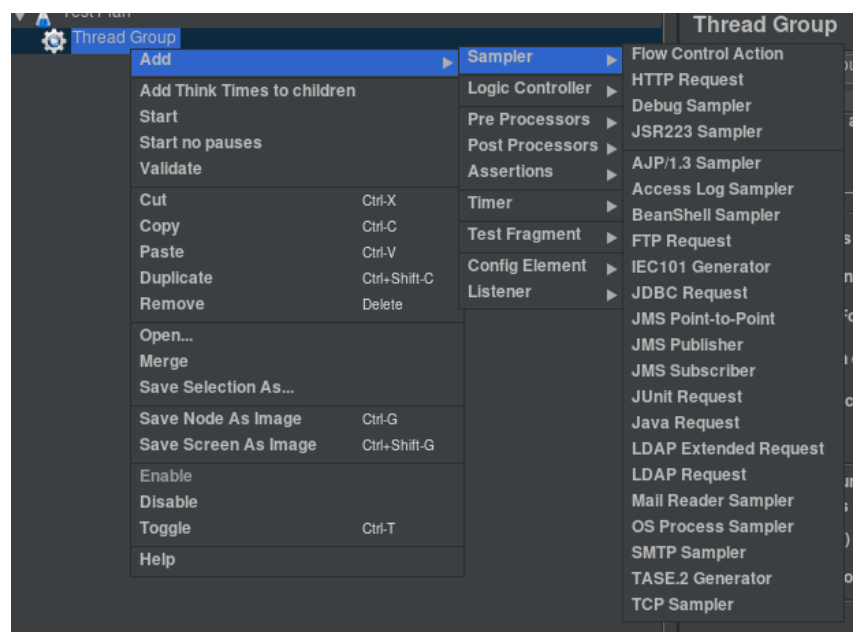
Spuštění zkompilované verze JMeter generátoru je velmi jednoduché. Je však nutné mít na paměti, že pro správné fungování generátoru musí být potřebné soubory knihoven nakopírovány do složky `/usr/lib`. Poté již stačí spustit zkompilovanou aplikaci JMeter obsahující vytvořené generátory pro protokol IEC 60870-5-101 skrze `.jar` soubor ve složce `bin`. Po přepnutí do složky `bin` je možné JMeter generátor spustit skrze příkaz `java -jar ApacheJMeter.jar`.

Samotné spuštění testů vyžaduje vytvoření scénářů skrze uživatelské skupiny procesů (Thread Group) a samotný protokolový sampler. Novou skupinu vláken lze vytvořit po stisknutí pravého tlačítka myši a nabídku `Add->Threads (Users)->Thread Group`.



Obrázek 1: Vytvoření skupiny vláken v aplikaci JMeter.

Samotné testy pak lze do nové skupiny vláken přidat po stisknutí pravého tlačítka myši a volbou z nabídky `Add->Sampler->IEC101 Generator`.



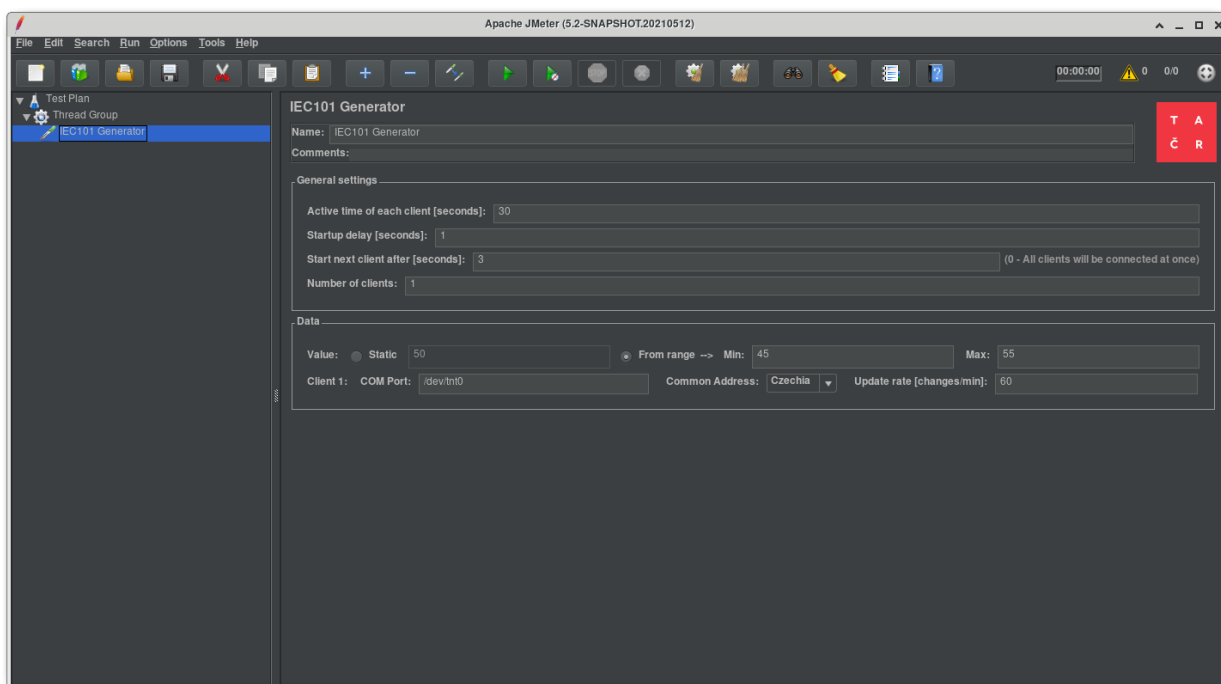
Obrázek 2: Přidání generátorů provozu.

6. Grafické uživatelské rozhraní a ovládání generátoru IEC101

Pro ovládání generátoru IEC101 byl vytvořen modul pro software Apache Jmeter. Prostřednictvím tohoto modulu lze nastavit jednotlivé parametry zasílaných zpráv a následně spouštět či zastavovat generování provozu pomocí standardních komponent programu Jmeter.

Grafické rozhraní Jmeter modulu IEC101 je zobrazeno na Obrázku 8. Rozhraní modulu je rozděleno do dvou sekcí (General settings a Data), ve kterých lze nastavit parametry generovaného provozu. V oblasti General settings lze nastavit:

- **Active time of each clients** – čas, po který každý z klientů generuje data. Hodnota se uvádí v sekundách.
- **Startup delay** – počáteční pauza před spuštěním jednotlivých klientů. Hodnota se uvádí v sekundách.
- **Start next client after** – časové rozestupy mezi inicializací jednotlivých klientů. Pomocí této hodnoty lze nastavit schodovitý průběh nárůstu počtu klientů a objemu datové komunikace při startu. Způsobí, že aktivní čas klientů se v čase s každým dalším klientem posunuje o uvedenou hodnotu. Pokud je nastavena nula, všichni klienti začnou a skončí komunikaci ve stejnou dobu. Hodnota se uvádí v sekundách.
- **Number of clients** – počet klientů.



Obrázek 3: Grafické rozhraní IEC 60870-5-101 generátoru

V další sekci Data je možné specifikovat číselnou hodnotu Value, která bude přenášena uvnitř zpráv IEC101. Lze volit a zadat buďto statickou hodnotu, nebo vybrat náhodnou hodnotu a specifikovat rozsah náhodného generování. V dalších řádcích lze specifikovat podrobnosti jednotlivých klientů. Každý řádek náleží jednomu klientu. Počet řádků se dynamicky mění podle hodnoty specifikované v poli Number of clients. U každého klienta lze specifikovat COM port, Common address (v tomto případě výběr regionu z rolovací nabídky). K výběru a nastavení Common address je v modulu využit slovník, kdy každý kód země je zastoupen jejím názvem.

V grafickém prostředí stačí pouze vybrat název země/regionu, a modul do IEC101 zpráv automaticky vloží patřičný kód. Aktuální implementovaný slovník je následovný:

- Czechia => 1
- Slovakia => 2
- Germany => 3
- Austria => 4
- Poland => 5

Poslední parametr, který lze u jednotlivých klientů nastavit je Update rate. Tato hodnota se určuje frekvenci zasílání zpráv a uvádí se v počtu zpráv za minutu. Generování provozu lze následně spustit pomocí standardního tlačítka “play” v horní liště programu Jmeter.

7. Testovací slave scénář IEC 60870-5-101

V případě protokolu IEC 60870-5-101 vystupuje generátor JMeter v roli slave endpointů, ke kterým se skrze sériové rozhraní připojují master instance IEC 60870-5-101 ze strany Adaptéru. Každý slave pak dle nastavení generátoru posílá skrze sériovou linku data na instanci master. V případě, že není k slave instanci připojen žádný master endpoint, jsou měnící se hodnoty ukládány do fronty na straně slave. V případě, že se master instance připojí až po spuštění generování hodnot na straně slave, jsou všechna data z jeho fronty okamžitě odeslána na master instanci (data jsou odeslána bez prodlení bez ohledu na nastavenou rychlost generování nových vzorků). V rámci provozu jsou odesílány hodnoty s plovoucí řadovou čárkou bez časové značky, dle protokolu IEC 60870-5-101 se jedná o datový typ 13. Vygenerované hodnoty jsou ze slave instancí odesílány přes sériovou linku přímo k master jednotkám. Na straně slave tedy neprobíhá žádné další zpracování dat, vygenerované hodnoty jsou pouze vypisovány do konzole. Veškeré zpracování a ukládání hodnot probíhá až na straně Adaptéru a master instancí. Po skončení testu jsou všechny slave instance ukončeny a přidělené systémové prostředky jsou uvolněny.

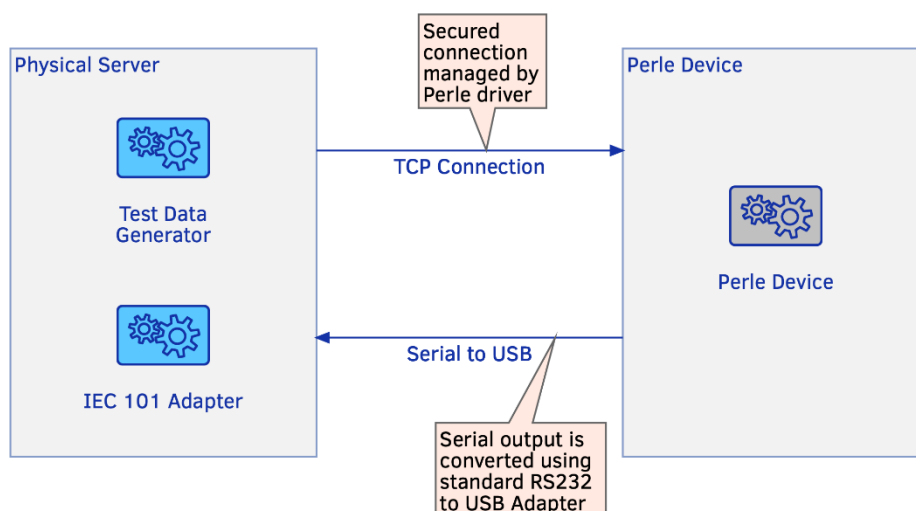
8. Popis komunikace protokolu IEC101

Ke komunikaci přes IEC 60870-5-101 protokol se využívá knihovny od firmy Real Thoughts. Jde o knihovnu v C/C++ a k ní dostupný Java wrapper. Komunikace má pouze jediný scénář. Adaptér funguje v roli Master stanice. To znamená, že se připojí na sériový port na kterém očekává zprávy od Slave stanice. Přicházející zprávy adaptér zpracovává, přičemž plně implementováno je jenom zpracování zprávy typu 13 – číslo s plovoucí řadovou čárkou bez časové značky. Zpracovávání zprávy je minimalistické, extrahuje se hodnota frekvence a číslo krajiny ze zprávy, přidá se čas přijetí zprávy a takhle se uloží přímo do databáze pro pozdější využití.

Všechna konfigurace aplikace probíhá při startu aplikace, přes proměnné v **application.properties**, které lze přepsat přímo, existencí stejnojmenné proměnné v operačním systému, nebo parametrem JVM. Jelikož aplikace běží jako Docker container, doporučený způsob je systémová proměnná.

8.1 Schéma komunikace

U komunikace IEC 101 budou obě části řešení, tedy generátor i adaptér, nasazeny na jednom dedikovaném fyzickém stroji. K tomuto serveru bude připojen Perle Device. Perle Device slouží jako převodník signálu pro sériovou komunikaci se širokou škálou nastavení a možností připojení. V tomto případě bude Perle Device připojen k serveru prostřednictvím ethernetového kabelu. Na serveru bude nainstalován Perle Driver, který zajišťuje to, že z pohledu aplikace se jedná o standardní sériový port. K RS232 výstupu na Perle Device bude připojen standardní převodník RS232/USB, který se USB stranou připojí zpět k serveru. Tímto bude zajištěno, že signál projde sériovou linkou.



Obrázek 4: Navržený model komunikace IEC 101

8.2 Inicializace IEC101 komunikace

Nejprve je nutné spustit master instanci, který naslouchá na příchozí spojení od slave instance. Inicializace probíhá pouze na úrovni linkové vrstvy, je-li to masterem vyžadováno. V takovém případě je třeba po změně stavu komunikace masterem zaslat zprávu s CoT (Cause of Transmission) s hodnotou 4, což je potvrzení inicializace spojení. Od té chvíle může slave instance zasílat data.

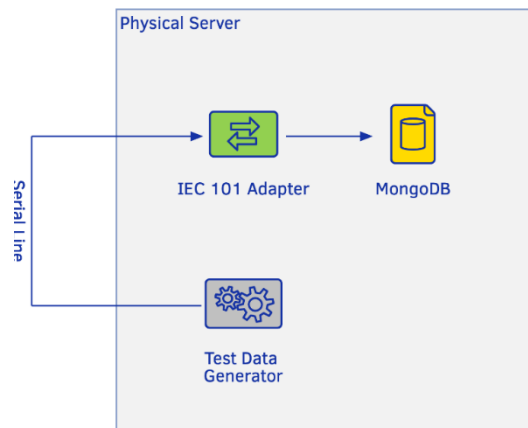
8.3 Průběh IEC101 komunikace

Standard IEC 60870-5-101 pro přenos dat využívá sériové rozhraní. Komunikace je řízena pomocí asynchronního kanálu ustanoveného mezi rozhraními DTE a DCE. Vzhledem k podpoře vyváženého i nevyváženého režimu přenosu je možné využívat různé topologie dvoubodové nebo vícebodové používající hvězdicové, liniové nebo kruhové spoje. IEC 101 rámec používá 1 start bit, 1 stop bit, jeden paritní bit a 8 datových bitů. Jsou definovány rámce s proměnnou či fixní délkou aplikačních informačních prvků ASDU a také jednoznakové zprávy určené pro potvrzení příjmu rámců.

Využívá se architektura master-slave, přičemž master uzel řídí komunikaci. Při použití vyváženého režimu může zaslání zpráv iniciovat master i slave, naopak při použití nevyváženého režimu jsou zprávy iniciovány jen master uzlem.

8.4 Zpracování dat z komunikace IEC101

Adaptér a příjem dat je sloučen do jedné aplikace. Po přijetí dat adaptérem jsou data rozparsována a uložena do persistentního úložiště, v tomto případě MongoDB. Veškerá přijatá data jsou dostupná skrze REST API aplikace.



Obrázek 5: Zpracování a ukládání dat do database

REST rozhraní pro IEC101 obsahuje jedinou metodu – **findReportedFrequency**. Ta slouží na získání zpracovaných dat z databáze. Částečně tedy nahrazuje rozhraní v Data Processing aplikaci, s omezenou funkcionalitou. Umožňuje pouze získat data seřazeny sestupně, dle času přijetí. Metoda podporuje stránkování pomocí parametrů **Page** a **PageSize**.