

Serviço de Descoberta Local de Dispositivos OPC-UA

Local Discovery Service for OPC-UA Devices

Alexandre Fonte¹, João M. L. P. Caldeira^{1,2}, Vasco N. G. J. Soares^{1,2}, Pedro Torres^{1,3},
Rogério Dionísio^{1,4} and Sérgio Malhão¹

¹Instituto Politécnico de Castelo Branco, Portugal

²Instituto de Telecomunicações, Portugal

³SYSTEC (Research Center for Systems & Technologies), FEUP, Faculdade de Engenharia, Universidade do Porto, Porto, Portugal

⁴DiSAC (Digital Services, Applications and Contents), Instituto Politécnico de Castelo Branco, Portugal

{adf; jcaldeira; vasco.g.soares; pedrotorres; rdionisio}@ipcb.pt; smalhao@ipcbcampus.pt

Resumo — Este artigo apresenta o contributo do Instituto Politécnico de Castelo Branco (IPCB) no âmbito do PPS1 do programa do programa Mobilizador PRODUTECH SIF – Soluções para a Indústria de Futuro, ao nível da definição de módulos para escalabilidade, adaptação, *plug-and-play*, com interoperabilidade entre processos e tecnologias inter/intra plantas industriais. Mais especificamente, a sua contribuição ao nível da definição, implementação e avaliação de serviços de descoberta de dispositivos OPC-UA. Adotando a plataforma CompactRIO desenvolvida no PPS2 como base física para uma *Smartbox*, o desenvolvimento foi estendido durante a presente tarefa com vista à incorporação e validação dos mecanismos de descoberta de serviços adotados no programa.

Palavras Chave - Serviço de Descoberta; OPC-UA; IoT; IIoT.

Abstract — This article presents the contribution of the Polytechnic Institute of Castelo Branco (IPCB) within the scope of PPS1 of the PRODUTECH SIF program - Solutions for the Future Industry, in terms of the definition of modules for scalability, adaptation, *plug-and-play*, with interoperability between processes and technologies inter/intra industrial plants. More specifically, its contribution for the definition, implementation and evaluation of OPC-UA device discovery services. Adopting the CompactRIO platform developed on PPS2 as a physical basis for a *Smartbox*, its development was extended during the present task with a view to incorporating and validating the adopted service discovery mechanisms in the program.

Keywords – Discovery Service; OPC-UA; IoT; IIoT.

I. INTRODUÇÃO

Um desafio central da Indústria 4.0 e da Internet das Coisas Industrial (IIoT) é a troca confiável e padronizada de dados entre dispositivos, máquinas e serviços localizados no mesmo domínio administrativo ou sob diferentes domínios administrativos. O Modelo de Arquitetura de Referência adotado para a Indústria 4.0 (RAMI 4.0) recomenda a adoção da Arquitetura Unificada OPC (OPC-UA) para implementação da camada de comunicação. Segundo a *Industrie 4.0 Platform*, os produtos têm que cumprir parcial ou totalmente um conjunto de

critérios chave para uma “Comunicação Indústria 4.0”, sendo classificados como Industrie 4.0 “Basic”, “Ready” ou “Full”.

A existência de um serviço de descoberta pode ter um papel fundamental na implementação de um sistema universal *Plug & Produce* para uso em redes industriais, e simultaneamente assegurar o cumprimento dos critérios da Indústria 4.0. É requerido que este serviço permita aos dispositivos OPC-UA ou *administrative shells/smartboxes* com configuração zero, a descoberta automática de dispositivos ou aplicações. Significa que, um dispositivo pode ligar-se dinamicamente a uma rede industrial, e após um procedimento de registo (como obter um endereço IP, e encontrar/especificar os protocolos e as políticas de segurança), ficar pronto a consumir ou disponibilizar os seus dados a outros componentes. Finalmente, um dispositivo deve poder desligar-se da rede de forma ordeira, desencadeando um conjunto de ações por forma a que o sistema não seja deixado num estado indesejado.

As especificações OPC-UA existentes para os Serviços de Descoberta visando cumprir com estes requisitos têm vindo a considerar três tipos de configurações: 1) os dispositivos cliente e servidor residem no mesmo *host*; ou 2) residem na mesma rede; ou 3) residem em diferentes localizações no sistema (planta ou empresa); as quais são suportadas por três tipos de servidores: um servidor local de descoberta (LDS), um servidor local de descoberta com extensão *multicast* (LDS-ME) e um servidor de descoberta global (GDS) [1]. Além disso, os servidores podem ser instalados em diferentes tipos de dispositivos, como um sensor, um dispositivo de controle ou uma estação de trabalho.

Este artigo apresenta o contributo do Instituto Politécnico de Castelo Branco (IPCB) no âmbito do PPS1 do programa Mobilizador PRODUTECH SIF – Soluções para a Indústria de Futuro, ao nível da definição de módulos para escalabilidade, adaptação, *plug-and-play*, com interoperabilidade entre processos e tecnologias inter/intra plantas industriais [2]. Mais especificamente, ao nível da definição, implementação e avaliação de serviços de descoberta de dispositivos OPC-UA. Em articulação com o PPS2, foi adotada a plataforma CompactRIO [3] desenvolvida no PPS2 como base física para a

Smartbox, cujo desenvolvimento foi estendido durante a presente tarefa com vista à incorporação e validação dos mecanismos de descoberta de serviços adotados na tarefa 1.3.3 do conjunto das atividades programadas para o PPS1 do programa PRODUTECH SIF.

Este artigo está estruturado da seguinte forma. Na Secção II é dada uma visão geral da arquitetura OPC-UA, incluindo os serviços de descoberta. Na Secção III, o funcionamento do serviço de descoberta adotado. Na Secção IV, é apresentada a arquitetura de Rede OPC-UA da implementação do serviço de descoberta a adotar no âmbito deste projeto. A Secção V apresenta os resultados que advêm da implementação do serviço de descoberta descrito, incluindo a sua avaliação. Finalmente, a secção VI conclui este documento, sistematizando o trabalho realizado e apresentando propostas de trabalho de futuro.

II. ARQUITETURA OPC-UA E OS SERVIÇOS DE DESCOBERTA

A. Visão Geral da Arquitetura OPC-UA

A Arquitetura Unificada do OPC (*Object Linking and Embedding for Process Control Unified Architecture*), ou abreviadamente *OPC-UA*, é uma tecnologia de comunicação baseada em TCP/IP (*Transmission Control Protocol/Internet Protocol*) desenvolvida pela *OPC Foundation* para permitir troca de informação em sistemas de automação, normalizada e independente do fabricante. O OPC-UA também é conhecido como um protocolo de comunicação M2M (*Machine to Machine*) [4].

São características do OPC-UA:

- A equivalência funcional – Todas as especificações do OPC clássico são mapeadas para UA;
- A independência de plataforma – Funcional em microcontroladores, autómatos, FPGAs, etc;
- A segurança – Criptografia, autenticação;
- Extensível – capacidade de adicionar novos recursos sem afetar os já existentes;
- Modelação da informação;

1) *Equivalência Funcional com o OPC Clássico*: O OPC-UA é funcionalmente equivalente ao OPC clássico, não limitado apenas a sistemas operativos Windows, acrescentando ainda:

- Funcionalidades de descoberta – Encontrar servidores OPC-UA em redes locais;
- Espaço de endereçamento – todos os dados são representados hierarquicamente (arquivos e pastas), facilitando a utilização por clientes OPC-UA;
- Assinaturas – possuir um monitor de dados/informações que permite relatar por exceção quando os valores mudam baseando-se nos critérios do cliente;
- Eventos – notificar informações importantes com base no critério do cliente;
- Métodos – os clientes podem executar programas, através de métodos definidos no servidor.

Garante a compatibilidade com versões anteriores para produtos existentes e permite a sua escalabilidade.

2) *Protocolo de Comunicação Cliente-Servidor OPC-UA*: O OPC-UA permite comunicação com uma arquitetura integrada, simples e bidirecional, baseada num protocolo de comunicações cliente-servidor. O servidor OPC-UA é responsável por disponibilizar informação sobre sensores/atuadores aos clientes que lhe solicitem a informação, tal como mostra a Figura 1.

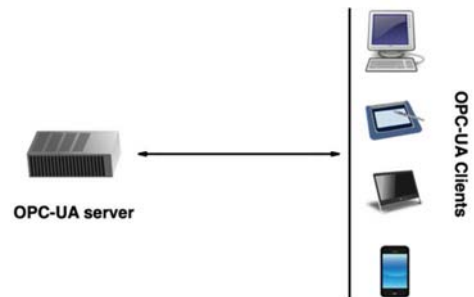


Figure 1. Arquitetura OPC-UA.

Mais especificamente, o servidor fornece acesso a dados e funções estruturados em um modelo de informações orientado a objetos com o qual os clientes podem interagir por meio de um conjunto de serviços padronizados (Discovery Service Set, SecureChannel Service Set, Session Service Set, NodeManagement Service Set, View Service Set, Query Service Set, Subscription Service Set, entre outros).

Destes serviços, o foco do presente documento é o Discovery Service Set que oferece três operações:

- FindServers();
- GetEndpoints();
- RegisterServer().

Um par de estruturas de dados de resposta a solicitação é definido para cada serviço. Essas estruturas são criadas a partir de tipos básicos, como números inteiros etc., e codificadas em um formato binário personalizado ou em XML.

O modelo de informação OPC-UA sobre os quais estes serviços operam, combina as ideias do paradigma de orientação a objeto e tecnologias de semânticas. As principais mensagens OPC-UA do serviço de Leitura, *ReadRequest* e *ReadResponse*, permitem ler os atributos dos nós OPC-UA no modelo de informação OPC-UA. A maioria dos serviços exige, contudo, que o cliente estabeleça um canal seguro e uma sessão antes de serem usados (tal como será ilustrado na secção de Implementação e Resultados).

3) *Campos de Aplicação do OPC-UA*: O OPC-UA cobre a lacuna entre o mundo da computação baseada em IP e o chão de fábrica. As interfaces, gateways e a consequente perda de informação são coisas do passado, uma vez que todos os dados do processo de produção são transferidos por meio de um único protocolo, seja dentro da mesma máquina, entre máquinas ou entre uma máquina e uma base de dados na nuvem. O OPC-UA elimina a necessidade de usar os sistemas de barramento de

campo tradicionais no nível da fábrica, sendo um padrão aberto de comunicação de dados industriais, e ao contrário do OPC Classic, o OPC-UA utiliza um modelo de informação orientado a objetos, que suporta estruturas, objetos, máquinas de estado, além de ser independente do sistema operativo.

B. Serviços de Descoberta OPC-UA

Segundo a parte 12 da especificação OPC-UA (*OPC Unified Architecture Specification Part 12: Discovery and Global Services Release 1.04 February 7, 2018*) para facilitar o processo de descoberta e ligação entre aplicações e dispositivos OPC UA podem ser implementados os seguintes tipos de servidores [5][1]:

- **Um Servidor Local de Descoberta (LDS)**, local a um *host* que mantém um registo das aplicações disponíveis no *host*.
- **Um Servidor Local de Descoberta com Extensão Multicast (LDS-ME)** que mantém um registo dos servidores locais e permite o registo de outros servidores com extensão ME através de anúncios multicast utilizando um grupo IP *multicast* específico.
- **Um Servidor de Descoberta Global (GDS)** que mantém o registo de descoberta para permitir a ligação de dispositivos entre diferentes domínios administrativos. Neste caso um dos principais desafios é assegurar elevados níveis de segurança, através do uso de certificados digitais para criação de um canal seguro (criptografia) e autenticação mútua.

No presente trabalho um dos objetivos é evitar a pré-configuração (*hard-coded*) nos clientes OPC UA. Assim, o foco foi implementar e avaliar um processo de descoberta automático para *Plug and Produce*. As próximas secções descrevem o funcionamento e arquitetura adotados.

III. FUNCIONAMENTO DO SERVIÇO DE DESCOBERTA ADOTADO

A. Processo de Descoberta: Lado Cliente

Os mecanismos de descoberta permitem aos clientes OPC-UA encontrar servidores OPC-UA na rede e descobrir com se devem ligar a eles. Após conhecer os dados de ligação a um servidor, os clientes podem guardar estes dados e usar mais tarde sempre que quiserem comunicar com esse servidor sem necessidade de executar novamente procedimentos de descoberta. Caso os dados usados não permitam ligar ao servidor em questão, o cliente deve proceder a nova descoberta de servidores.

Na implementação desenvolvida no âmbito deste projeto, a descoberta de servidores por parte dos clientes é realizada por:

- Descoberta fora de banda de um *DiscoveryUrl* para um Servidor;
- Chamar *FindServers* no LDS instalado no *host* do Cliente;
- Chamar *FindServers* num LDS remoto, onde o *HostName* do *host* remoto é inserido manualmente;

O *DiscoveryUrl* fornece todas as informações que um *Cliente* precisa para se conectar a um *DiscoveryEndpoint*.

1) *Descoberta Simples com um DiscoveryUrl*: Todo Servidor tem um ou mais *DiscoveryUrls* que permitem aceder aos seus *Endpoints*. Depois do Cliente obter o *DiscoveryUrl* para o Servidor (por exemplo, via entrada manual num formulário), lê os *EndpointDescriptions* usando o Serviço *GetEndpoints*.

2) *Descoberta Local*: Em muitos casos, os Clientes desconhecem que Servidores existem, mas eventualmente sabem quais os *hosts* podem ter Servidores. Nesta situação, o Cliente procurará o *LocalDiscoveryServer* num *host*, construindo um *DiscoveryUrl* usando os endereços conhecidos definidos na especificação. Se um Cliente encontrar um *LocalDiscoveryServer*, consultará o Serviço *FindServers* no LDS para obter uma lista de Servidores e dos seus *DiscoveryUrls*. O Cliente chamará depois o serviço *GetEndpoints* para um dos Servidores retornados.

B. Processo de Registo e Anúncio: Lado Servidor

Os serviços de descoberta a implementar no programa devem permitir que os servidores se registem para que possam posteriormente ser descobertos. O registo deve ser realizado no *Local Discovery Server*. Este procedimento permite que os servidores fiquem visíveis para descoberta local.

1) *Local Discovery Server*: Cada *host*, que pode ter várias aplicações detectáveis instaladas, deve ter um *LocalDiscoveryServer* independente instalado. O *LocalDiscoveryServer* deve expor um ou mais *Endpoints* que suportam os serviços *FindServers* e *GetEndpoints* definidos na especificação para todos as aplicações no *host*. Além disso, o *LocalDiscoveryServer* deve fornecer pelo menos um *Endpoint* que implemente o serviço *RegisterServer* para essas aplicações.

IV. ARQUITETURA

Para demonstração da solução proposta para o serviço de descoberta OPC-UA – *Local Discovery Service* (LDS), foi usada a arquitetura proposta no PPS2, apresentada na Figura 2. Esta arquitetura foi desenvolvida com base na utilização da plataforma CompactRIO da National Instruments como SmartBox [3][6]. Para demonstração e validação do serviço LSD, foi usada a plataforma cRIO-9040 da National Instruments (cRIO) e foram implementadas as arquiteturas de comunicação OPC-UA. O cRIO foi configurado como OPC-UA server e nele foi também instalado o serviço de descoberta (LDS).

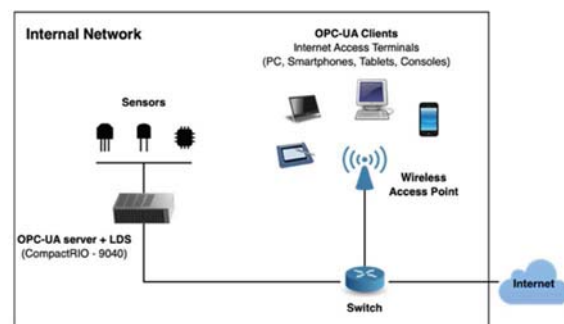


Figure 2. Arquitetura de Rede OPC-UA com o cRIO-9040.

O cRIO-9040 como servidor OPC-UA procede ao seu registo no serviço LDS, criando os parâmetros necessários para a ligação de clientes e a lista de aplicações disponibilizada. Para proceder à descoberta de um servidor (neste caso o cRIO-9040) os clientes devem aceder ao LDS e recolher a informação anunciada pelos vários servidores registados neste serviço. Os clientes, de seguida guardam as informações necessárias à ligação com os servidores alvo. Os clientes usam posteriormente os dados guardados para comunicar com os respetivos servidores e interagir com os mesmos no âmbito dos seus serviços e aplicações (ex. ler o valor de um sensor).

Caso um servidor não esteja contactável, os clientes repetem o processo de descoberta dos servidores disponíveis através do serviço LDS. Os servidores podem ficar incontactáveis devido a vários fatores, como sejam, foram desligados, perderam o acesso à rede ou alteração dos parâmetros de ligação à rede. Sempre que um servidor adiciona novas aplicações à sua lista disponibilizada (ex. foram acrescentados novos sensores) altera o seu registo no LDS em conformidade.

V. IMPLEMENTAÇÃO E RESULTADOS

Esta secção apresenta os resultados que advêm da implementação do serviço de descoberta local. Como termo de comparação, considerou-se a existência de um cenário de uma rede sem serviço de descoberta (sem servidor LDS) e um cenário de uma rede com serviço de descoberta (com servidor LDS).

A Figura 3 ilustra a configuração dos servidores OPC UA no servidor LDS para posterior acesso pelos Clientes OPC-UA, sem um serviço de descoberta. Na ausência deste, conforme se ilustra, o acesso de um Cliente OPC-UA a um servidor OPC-UA requer a introdução manual do endereço IP do servidor, neste caso obtido também manualmente através do explorador NI-MAX, no separador Network Settings onde é possível verificar qual é o endereço IP atribuído ao cRIO-9040, como ilustra a Figura 4.

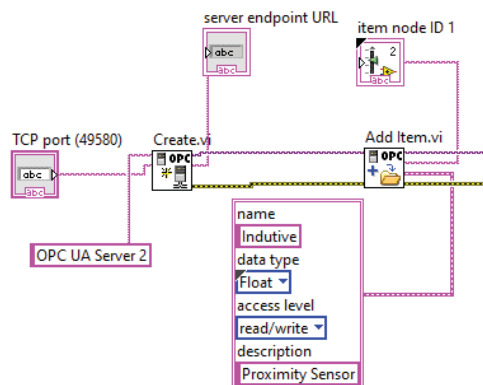


Figure 3. Configuração sem LDS

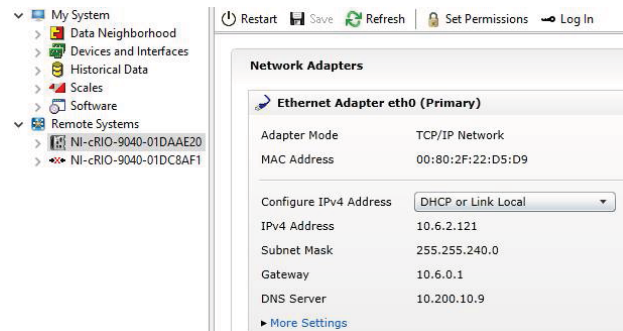


Figure 4. Endereço IP do cRIO-9040 no NI-MAX

Posteriormente, através do software Ua Expert (cliente OPC-UA) é realizada a conexão ao cRIO-9040 através do seu IP e da porta TCP definida, no caso a porta 49580 (por omissão) [7], conforme se observa na Figura 5. Na Figura 6, é possível visualizar os nós OPC-UA.

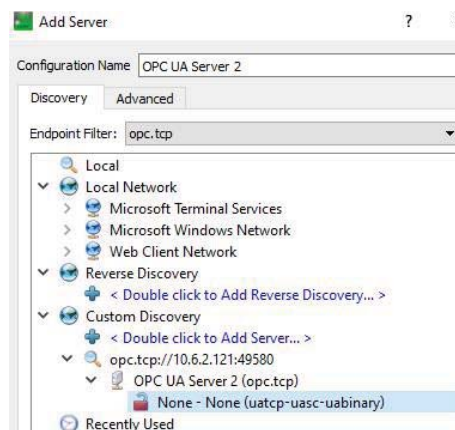


Figure 5. Conexão ao OPC-UA Server.

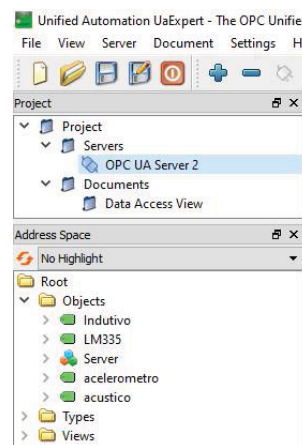


Figure 6. Conexão ao OPC-UA Server e visualização dos nós OPC-UA

Na Figura 7 é possível observar o tempo de ligação entre o cliente OPC-UA e o servidor OPC-UA.

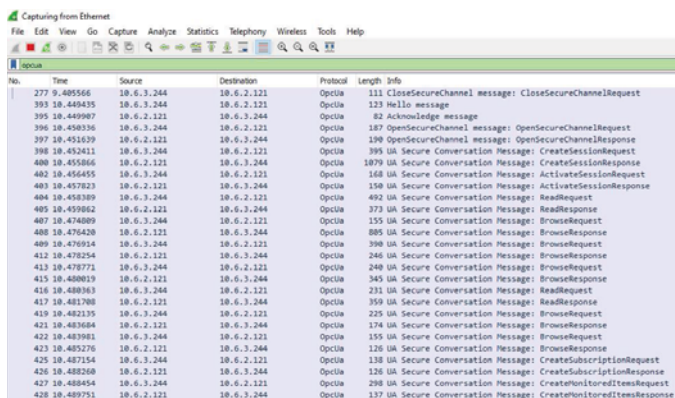


Figure 7. Tempo de ligação entre o cliente OPC-UA e o OPC-UA Server

De modo a ser possível o registo dos servidores OPC-UA no servidor LDS é necessário instalar o package do LDS disponível no site da OPC Foundation em [8].

Após a instalação do package anterior é necessário configurar o registo no servidor LDS no ficheiro de configuração do cRIO-9040 através da introdução do bloco Register Server.vi, onde é necessário posteriormente localizar o certificado digital do servidor LDS como ilustra a Figura 8.

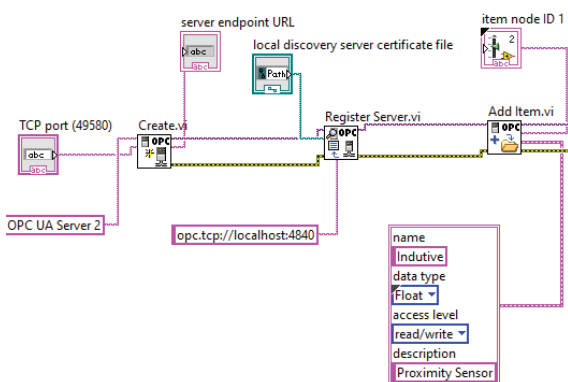


Figure 8. Configuração do LDS Server

Na configuração dos servidores OPC-UA no servidor LDS foi definido o porto 49320 para posterior acesso dos clientes OPC-UA (Figura 9).

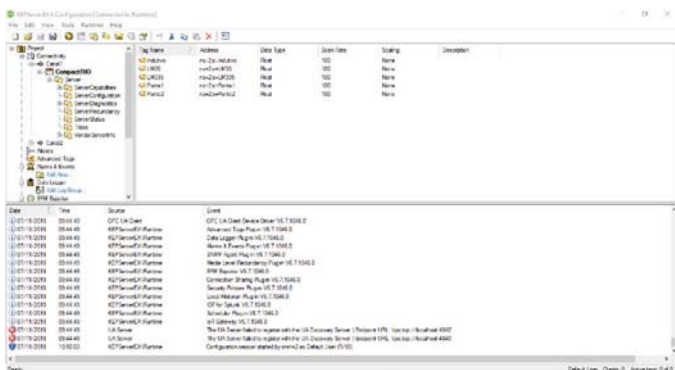


Figure 9. Configuração dos Servidores OPC-UA no Servidor LDS.

A Figura 10 ilustra a troca de Certificados de Segurança entre clientes e o servidor LDS.

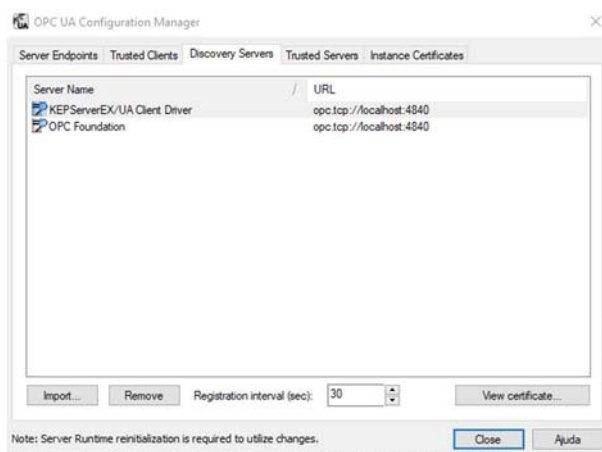


Figure 10. Troca de Certificados de Segurança.

A Figura 11 ilustra a lista dos certificados digitais dos clientes confiáveis entre clientes e o servidor LDS.

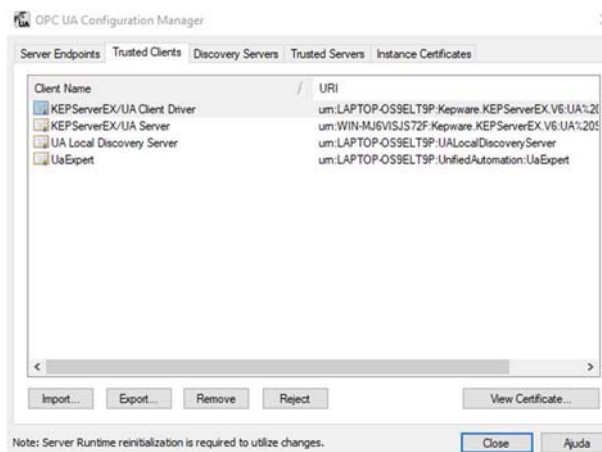


Figure 11. Certificados dos Clientes Confiáveis.

A Figura 12 mostra a lista dos certificados digitais dos servidores confiáveis de Certificados entre clientes e o servidor LDS.

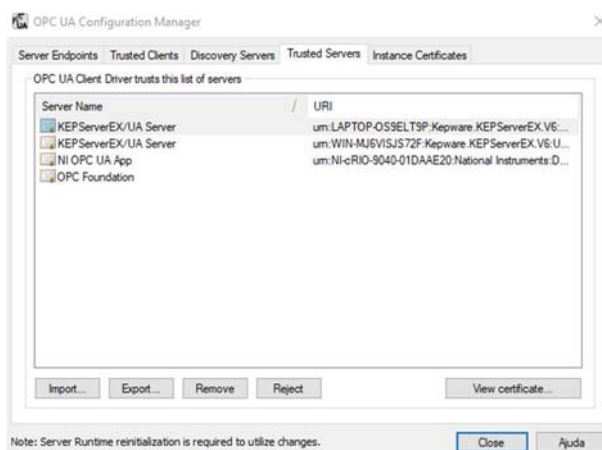


Figure 12. Certificados dos Servidores Confiáveis.

A Figura 13 (a) mostra a configuração de acesso através do endereço IP ao Servidor LDS para posterior visualização dos servidores OPC UA.

Enquanto, a Figura 13 (b) mostra os servidores OPC UA configurados no servidor LDS, e neste caso que cada servidor OPC UA foi configurado como um Canal.

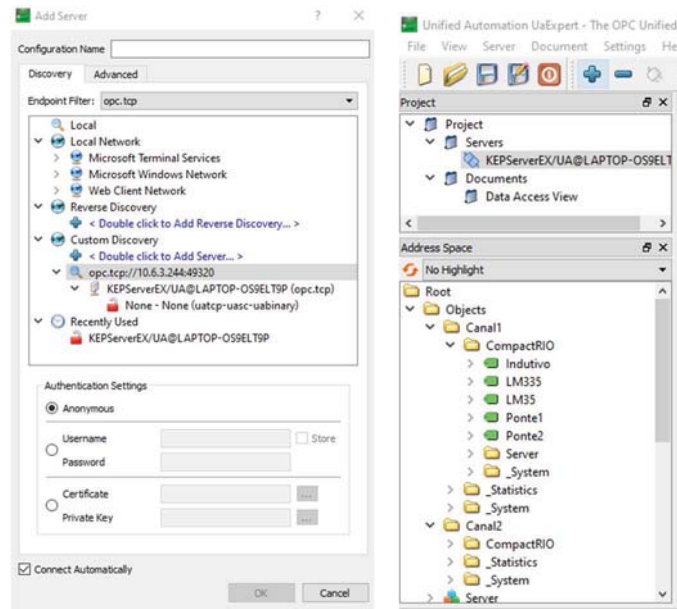


Figure 13. (a) Acesso ao Servidor LDS. (b) Servidores OPC UA Configurados no Servidor LDS.

Na Figura 14 pode-se observar o tempo necessário ao estabelecimento da ligação entre o cliente OPC-UA e o servidor LDS. Concluindo-se que a penalidade de latência envolvida é reduzida, mesmo envolvendo uma terceira entidade (neste caso uma interação adicional com o servidor LDS) antes do estabelecimento da ligação final.

No.	Time	Source	Destination	Protocol	Length	Info
132	5.275475	10.6.3.244	10.6.2.121	Opclia	95	CloseSecureChannel message: CloseSecureChannelRequest
133	5.275475	10.6.3.244	10.6.2.121	Opclia	123	Hello message
134	5.275475	10.6.3.244	10.6.2.121	Opclia	82	Acknowledge message
135	5.275475	10.6.3.244	10.6.2.121	Opclia	186	OpenSecureChannel message: OpenSecureChannelRequest
136	5.275475	10.6.3.244	10.6.2.121	Opclia	190	OpenSecureChannel message: OpenSecureChannelResponse
137	5.275475	10.6.3.244	10.6.2.121	Opclia	213	UA Secure Conversation Message: GetEndpointsRequest
138	5.275475	10.6.3.244	10.6.2.121	Opclia	94	UA Secure Conversation Message: GetEndpointsResponse
139	5.275475	10.6.3.244	10.6.2.121	Opclia	95	CloseSecureChannel message: CloseSecureChannelRequest
140	5.275475	10.6.3.244	10.6.2.121	Opclia	123	Hello message
141	5.275475	10.6.3.244	10.6.2.121	Opclia	82	Acknowledge message
142	5.275475	10.6.3.244	10.6.2.121	Opclia	186	OpenSecureChannel message: OpenSecureChannelRequest
143	5.275475	10.6.3.244	10.6.2.121	Opclia	190	OpenSecureChannel message: OpenSecureChannelResponse
144	5.275475	10.6.3.244	10.6.2.121	Opclia	185	UA Secure Conversation Message: CreateSessionRequest
145	5.275475	10.6.3.244	10.6.2.121	Opclia	1879	UA Secure Conversation Message: CreateSessionResponse
146	5.275475	10.6.3.244	10.6.2.121	Opclia	152	UA Secure Conversation Message: ActivateSessionRequest
147	5.275475	10.6.3.244	10.6.2.121	Opclia	158	UA Secure Conversation Message: ActivateSessionResponse
148	5.275475	10.6.3.244	10.6.2.121	Opclia	134	UA Secure Conversation Message: ReadRequest
149	5.275475	10.6.3.244	10.6.2.121	Opclia	122	UA Secure Conversation Message: CreateSubscriptionRequest
150	5.275475	10.6.3.244	10.6.2.121	Opclia	128	UA Secure Conversation Message: ReadResponse
151	5.275475	10.6.3.244	10.6.2.121	Opclia	126	UA Secure Conversation Message: CreateSubscriptionResponse
152	5.275475	10.6.3.244	10.6.2.121	Opclia	215	UA Secure Conversation Message: CreateMonitorItemsRequest
153	5.275475	10.6.3.244	10.6.2.121	Opclia	168	UA Secure Conversation Message: CreateMonitorItemsResponse
154	5.275475	10.6.3.244	10.6.2.121	Opclia	184	UA Secure Conversation Message: PublishRequest
155	5.275475	10.6.3.244	10.6.2.121	Opclia	184	UA Secure Conversation Message: PublishResponse
156	5.275475	10.6.3.244	10.6.2.121	Opclia	184	UA Secure Conversation Message: PublishRequest
157	5.275475	10.6.3.244	10.6.2.121	Opclia	184	UA Secure Conversation Message: PublishResponse
158	5.275475	10.6.3.244	10.6.2.121	Opclia	212	UA Secure Conversation Message: PublishResponse

Figure 14. Tempos de ligação entre o cliente OPC-UA e o LDS Server

VI. CONCLUSÕES E TRABALHO FUTURO

Recorrendo e complementando o protótipo de *SmartBox* que foi desenvolvido no PPS2 do programa PRODUTECH SIF, baseado na plataforma CompactRIO (cRIO) da National Instruments, cujo desenvolvimento foi estendido para este trabalho, implementaram-se e avaliaram-se dois cenários de rede: um cenário de rede sem serviço de descoberta local (LDS), e um cenário de rede com serviço de descoberta local (LDS) para assegurar a ligação e interação entre dispositivos cliente e servidor OPC-UA. Das avaliações, conclui-se que a inclusão de um servidor LDS na rede permite não só automatizar e facilitar os operadores da rede industrial no processo de adição de novos dispositivos com configuração zero, como a penalidade de latência envolvida no processo de adição é relativamente reduzida em comparação com os benefícios oferecidos.

Em resumo, este trabalho demonstrou a operacionalidade do serviço de descoberta em cenários reais de chão de fábrica, muito útil em cenários de múltiplos dispositivos ligados, conforme se objetivava.

Como trabalho futuro, será implementado um Servidor Local de Descoberta com Extensão Multicast (LDS-ME) e avaliadas as reais vantagens e desempenho desta alternativa. Adicionalmente, é importante realizar-se uma análise da robustez da implementação a ataques de negação de serviço, recorrendo, por exemplo, uma *botnet* de dispositivos e uma estrutura de comando e controlo.

AGRADECIMENTOS

This research was supported by the project PRODUTECH SIF - Solutions for the Industry of the Future, financed by the Portuguese National program COMPETE 2020.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] Foundation, O. (2018). OPC Unified Architecture - Interoperability for Industrie 4.0 and the Internet of Things. Consultado em 25 de junho de 2018, disponível em <https://opcfoundation.org/wp-content/uploads/2017/11/OPC-UA-Interoperability-For-Industrie4-and-IoT-EN.pdf>
- [2] Programa Mobilizador PRODUTECH SIF – Soluções para a Indústria de Futuro – Consultado em 24 de janeiro de 2020, disponível em <http://mobilizadores.produtech.org/pt/front-page-platforms/sif>.
- [3] cRIO-9040. (2019). Consultado em 20 de janeiro de 2019, disponível em <https://www.ni.com/pt-pt/shop/select/compactrio-controller?modelId=229876>
- [4] Ascolab - OPC Unified Architecture. (2018). Consultado em 11 de agosto de 2018, disponível em <http://www.ascolab.com/en/technology-unified-architecture/>
- [5] Foundation, A. (2018). Apache. Consultado em 23 de setembro de 2018, disponível em <https://www.apache.org/>
- [6] OPC UA Toolkit Labview. (2019). Consultado em 23 de janeiro de 2019, disponível em <http://www.ni.com/pdf/manuals/376233a.html>
- [7] UaExpert—A Full-Featured OPC UA Client. (2018). Consultado em 16 de junho 2018, disponível em <https://www.unified-automation.com/products/development-tools/uaexpert.html>
- [8] Package LDS, OPC Foundation. Consultado em em 23 de janeiro de 2019, disponível em <https://opcfoundation.org/developer-tools/samples-and-tools-unified-architecture/local-discovery-server-lds/>