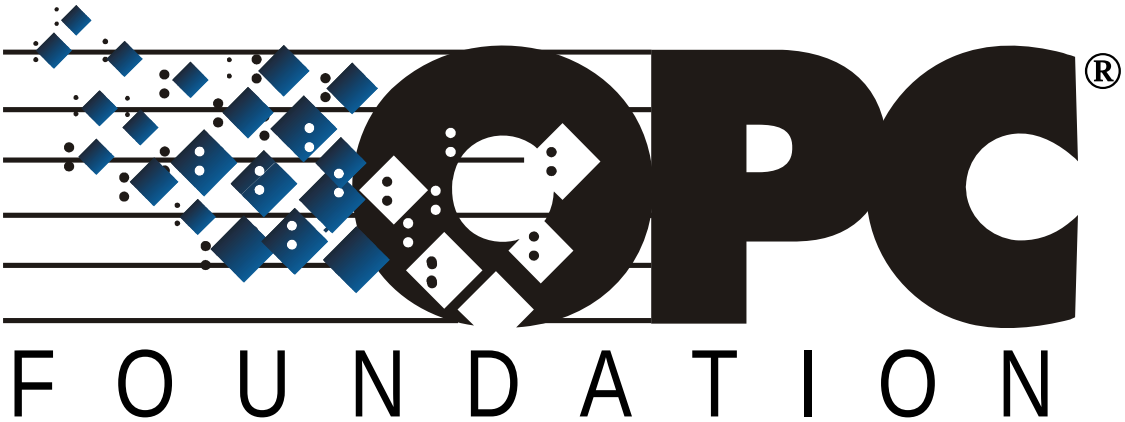


OPC UA のセキュリティ対策と証明書の運用

Whitepaper Version 1.0

March 12, 2018

OPC Council Japan, Technical Committee

はじめに

OPC Foundation は、産業オートメーションの分野で安全に高信頼なデータ交換を実現するための規格を 1996 年に発表しました。現在これは OPC Classic と呼ばれていますが、OPC Classic が提供する相互接続性と相互運用性により、制御ネットワークでの表示装置と制御機器との通信ドライバ開発の効率化と品質向上を実現し、主にプロセスオートメーション市場で広く認知されました。

その後 OPC Foundation は、制御ネットワークより広い領域で、産業オートメーションの分野に限らない汎用的な情報連携基盤となる新たな標準規格を OPC Unified Architecture (OPC UA)として 2008 年に発表しました。この規格はオブジェクトモデルを採用することで多様な情報を表現して伝えることができ、プラットフォーム非依存とすることでどのような環境の相手ともつながることができ、IT 分野で実績のあるセキュリティ技術をとりいれることで安全なコミュニケーションを実現しました。これにより、OPC UA は、現場レベルのセンサーおよび制御機器から、事業所レベルの製造実行管理、さらに上位の経営層レベルでの生産計画または経営資源管理への垂直方向と、インターネット経由で国境をまたがった水平方向への広い領域での使用が可能となりました。

このコンセプトにより OPC UA は Industrial Internet of Things(IIoT)や Industry 4.0 への関心からも広く注目されています。特にセキュリティについては制御システムを狙った Stuxnet の登場やランサムウェアの急増などにより安全性に対する不安が挙がっている背景から高い関心が持たれ、ドイツ政府の情報技術セキュリティ担当省庁である Bundesamt für Sicherheit in der Informationstechnik （BSI）が OPC UA のセキュリティ仕様を評価し、充分な有効性があることをレポートしました。

本稿では OPC UA のセキュリティ仕様の概要と、OPC UA を初めて使用する際にセキュリティに関して必要となるアプリケーション証明書の運用について紹介します。

具体的なセキュリティ対策

本節では盗聴や改ざん、DoS 攻撃、なりすましからの脅威に対して、OPC UA がどのように対策しているかについて紹介します。

盗聴は、ネットワーク上に流れるメッセージを不正に盗み取ることです。経営者は生産データを生産設備からネットワーク経由で取得しています。もし、ネットワークに流れるメッセージが平文であるならば、悪意を持つ者によって生産活動に関わる重要な情報を盗み取られるという脅威があります。この対策として、OPC UA は暗号化したメッセージをネットワーク上に流し、受け取った先で暗号化されたメッセージを複号できる仕様になっています。ネットワークの途中で盗まれたメッセージは暗号化されているので、鍵を持たないものがそれを解読することは困難な作業となります。(図 1)

図 1 メッセージの盗聴防止

改ざんは、ネットワークに流れているメッセージを不正に変更することです。悪意を持つ者により、監視メッセージをプラントのオペレータに気づかれることなくバルブを全開にするような制御メッセージに変更して、生産設備の破壊を企てるような脅威があります。この対策として、OPC UA はデジタル署名による証明書の交換や、メッセージ認証によるメッセージ認証コード付きのメッセージをネットワークに送ることにより、受け取った先でデジタル署名を検証してメッセージの改ざんを検知できるようにしています。(図 2)

図 2 メッセージの改ざん防止

DoS 攻撃とは、ネットワーク上にシステム稼働率を低下させる多量のメッセージを流すことです。悪意を持つ者は、サーバーを過負荷状態にしてシステムの利用を邪魔し、しいてはシステムダウンを引き起こそうとします。また、繰り返し接続を試みてネットワークによる設備の状態監視を妨げようとする脅威もあります。この対策として、OPC UA はセキュアな接続を形成するまでのサービス処理を簡素化し、さらに単位時間で受け付けるメッセージの量を制限します。これにより、攻撃によって使用されるリソースを最小化し、CPU の負荷を軽減します。(図 3)

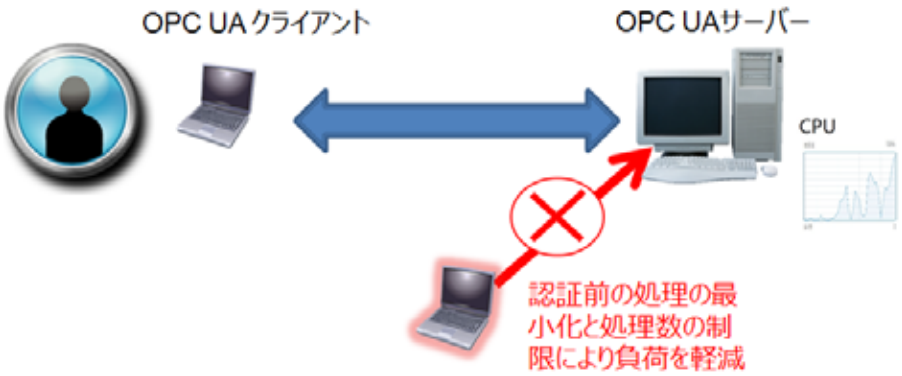


図 3 DoS 攻撃の軽減

なりすましは、接続元を偽って不正に接続することです。悪意を持つ者が不正な OPC UA クライアントを使って OPC UA サーバーに接続し、生産設備に深刻な損害を与えるなどの脅威があります。この対策として、OPC UA は様々な証明書を接続時に交換することで、接続先が信頼できる相手であることを確認することができます。(図 4) アプリケーション証明書は OPC UA アプリケーションの識別情報を持ち、互いの OPC UA アプリケーションが信頼できる接続相手であるかの検証に使われます。ソフトウェア証明書にはその OPC UA アプリケーションが提供する機能を情報に持ち、接続相手が自分の提供するサービスに対応可能であるかを確認することができます。

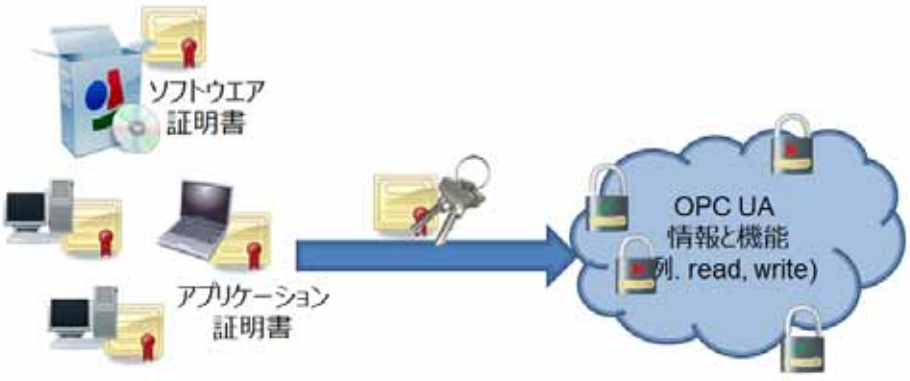


図 4 なりすまし防止

不正な OPC UA クライアントからの接続ではなく、企業内部のインサイダーによって正規の OPC UA クライアントから接続してくる場合への対策として、OPC UA サーバーは利用者のアカウント情報を提供することを求め、そのアカウントに許可された範囲の機能のみ利用できるように制限をかけることができます。アカウント情報にはユーザー名／パスワード入力や、アカウントの証明書などがサポートされています。(図 5)

図 5 ユーザー認証

また、OPC UA では接続や様々なサービスを監査イベントとして通知し、監査ログとして一元管理することが可能です。障害の前に意図的な不正操作が、いつ誰によっておこなわれたかを知る仕組みを備えていることは、インサイダーに対する不正操作の歯止めにも使えます。

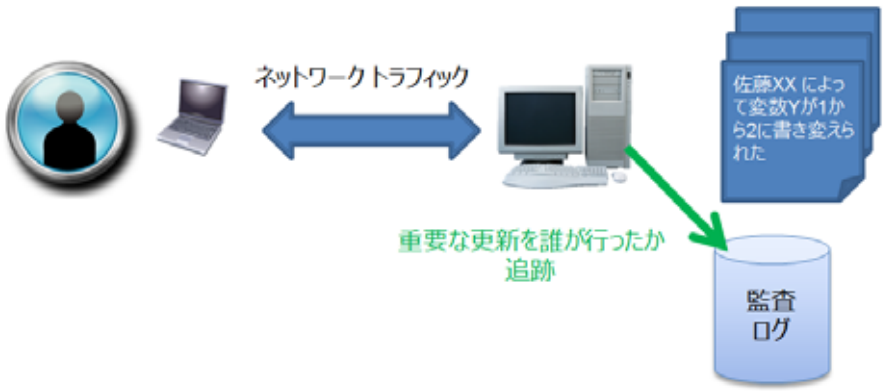


図 6 監査ログ

具体的な脅威を取り上げてその対策を説明してきましたが、OPC UA の仕様書 Part2 にはさらにいくつかの想定される脅威が紹介されてます。

つぎの節ではここまでに述べた脅威への対策を機能要件という観点から整理して、それぞれの機能が OPC UA アプリケーションでどのように実現されるかについて紹介します。

セキュリティ要件

OPC UA アプリケーションの実装に必要で OPC Foundation から提供されるコミュニケーションスタックというライブラリの中には機密性、完全性、可用性の要件を実現する機能が組み込まれています。(図 7)

機密性は通信中のメッセージの内容を他者に漏洩しないことを保証するもので、OPC UA はメッセージの暗号化によって機密性を実現します。機密性を高めることは盗聴の脅威への対策になっています。

完全性は通信中のメッセージの内容を他者から改ざんされないようにするもので、OPC UA ではメッセージ認証によって改ざんされたデータを受信した時点で検知することができます。

可用性はシステムが継続して利用できることです。このために OPC UA のコミュニケーションスタックでは、相手が信頼できるか判断できていない状態の通信処理（接続）を最小化することで可用性を下げないようにしています。また、第三者が接続確立後のメッセージのやり取りを盗み取りそのメッセージを繰り返し送りつけたとしても、メッセージにはシーケンス番号が振られているのでそのような不整合なメッセージを排除しています。

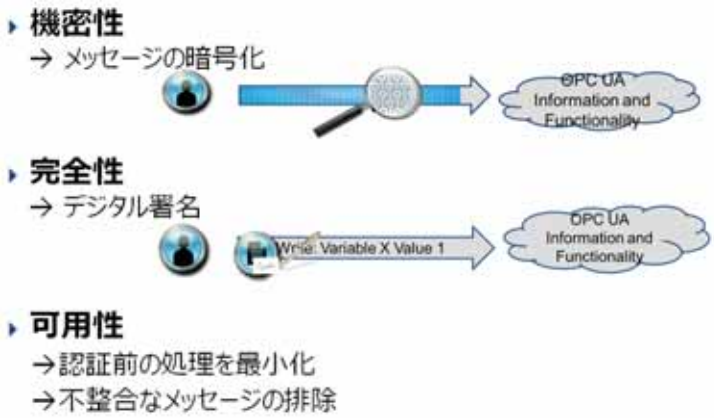


図 7 トランスポート層のセキュリティ要件

コミュニケーションスタックで実現しているもの他に、OPC UA アプリケーションを設計する際に作り込まなければならないセキュリティ要件に、認証、認可、監査があります。(図 8) 認証にはユーザー認証とアプリケーション認証があります。

ユーザー認証は、アプリケーションに対してユーザーを識別するための仕組みです。ユーザーを識別する手段として、ユーザー名とパスワードのペアや証明書を使うことができます。また、アプリケーションを実行するプラットフォームのユーザー認証基盤を流用することもできます。

アプリケーション認証は、接続先の相手を識別する仕組みです。証明書は、アプリケーションのインストール毎に持つ必要があります。個々のインスタンスで作成した証明書や認証局が作成した証明書のどちらも利用できますが、それぞれの認証処理は OPC UA サーバーで作り込む必要があります。

認可は、OPC UA サーバー内の情報モデルを構成する要素ごとにシステムとしてのアクセスレベルとユーザーごとのユーザーアクセスレベルの情報を属性に設定することができるので、OPC UA クライアントは自分のアクセス対象への権限を認識できますし、OPC UA サーバーはその設定内容に従った認可処理を作り込むことになります。

監査は、OPC UA サーバーへの接続や失敗などのセキュリティに関するイベントが生成されるので、それを記録することで実現します。コミュニケーションスタックの内部からは標準的な動作の監査イベントが発行されますが、アプリケーションに特化した情報のイベント通知や、通知イベントをどのように記録に残すかの処理はアプリケーションの作り込みになります。

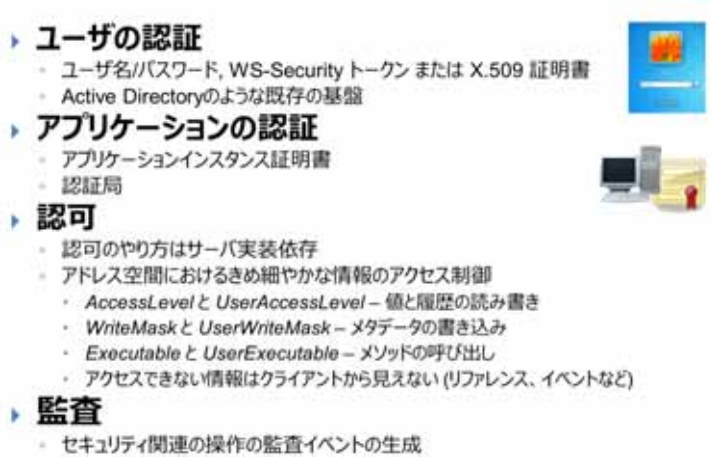


図 8 アプリケーション層のセキュリティ要件

OPC UA の仕様書 Part2 には、ここで説明したセキュリティ要件の項目があります。そして、それぞれの脅威とセキュリティ要件の関連を示しています。詳しく知りたい方は、OPC UA の仕様書 Part2 を参照してください。

OPC UA 仕様のセキュリティの範囲とポリシー

OPC UA 仕様が規定するセキュリティは、OPC UA クライアントと OPC UA サーバー間の通信に関するものでしかありません。標的型メール攻撃による情報漏洩あるいはシステムにひそむ脆弱性に対する対策のために、企業は情報セキュリティポリシーで基本方針、対策基準、実施手順を決める必要があります。そのなかで、個々の認証対象の管理、例えば、ユーザーの認証の管理における、パスワードの規約や有効期間、サーバールームへの入退出管理や、作業者の教育などの組織的な運用がなされることになります。

OPC UA が提供しているセキュリティ仕様は通信に関するものでしかありませんが、一般に使われている IT の機能を利用しているので、これらの情報セキュリティポリシーに親和性よく適用することができます。表 1 に OPC UA 仕様のスコープに入らないセキュリティ運用項目を挙げます。これらは、IEC 62443(ISA 99)、NERC CIP や企業の規定・標準で説明されています。

表 1 OPC UA 仕様で規定されていないセキュリティ運用項目

運用項目	詳細説明
ユーザー管理	ユーザーの追加、削除、ロールの割り当て方法
ユーザー権限の管理	アクセス権限の設定方法
ユーザー認証の管理	認証方法 パスワードの規約 文字の規約(最小文字数、大文字、数字、特殊文字など) パスワードの有効期間

	パスワードの保管方法 ※OPC UA は、ユーザトークンの種別や受け渡しのためのサービスは定義しておりますが、認証方法は規定していません。
証明書の管理	証明書の更新、失効、削除 証明書の暗号化強度や有効期限の設定
組織的な課題	サイトに物理的なアクセス処理の仕方 ゾーン、セキュリティライフサイクルまたはセキュリティポリシー 人材の育成

OPC UA は、デバイスや制御層から情報・基幹システムまで広い領域をカバーします。それぞれの場所で、要求されるセキュリティ ポリシーは異なります。OPC UA はこれらの要件に柔軟に対応できます。OPC UA は、クライアントとサーバーの接続ごとに署名/暗号化をするしない（セキュリティポリシー）を選択できます。そして、暗号化を選択した場合、その暗号化の強度（セキュリティモード）を選ぶことができます。(表 2)

表 2 OPC UA のセキュリティ プロファイル

セキュリティ プロファイル	設定値
セキュリティ ポリシー	SignedAndEncrypt ― メッセージに署名を付け、かつ暗号化 Signed ― メッセージに署名を付けるが暗号化なし None ― セキュリティなし
セキュリティ モード	Basic256Sha256 ― 高いセキュリティ Basic256 ― 中程度から高いセキュリティ Basic128Rsa15 ― 中程度セキュリティ None ― セキュリティなし

証明書の運用方法

本稿の後半では暗号化やデジタル署名の機能を使って OPC UA アプリケーションの通信を行う際に必要な運用について紹介します。

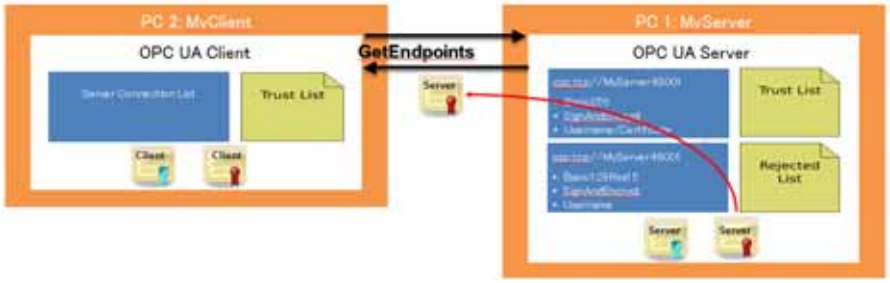
サンプルプログラムなどで OPC UA を試してみたいと考える方々がよく直面する課題があります。それは、クライアント・サーバー間の信頼関係の構築です。OPC UA は、セキュリティを有効化して通信経路を確立する場合、接続に先立ち相互認証を行います。従って、お互いのアプリケーションは、予め信頼できる相手の証明書を知る必要があります。ここでは OPC UA で正しく証明書を運用するための手順を特定のサンプルアプリケーションを使った具体例で紹介します。

初期状態では、OPC UA のクライアントとサーバーは相手を知らない状態になっています。具体的には OPC UA アプリケーションにおいて、接続を許可する相手のアプリケーション証明書を保持するためにある信頼リスト(Trust List)は空になっており、どの相手からの接続も拒否

する状態になっています。はじめに OPC UA クライアントは接続する OPC UA サーバーのエンドポイント情報を取得するために GetEndPoints サービスを呼び出します。戻り値として、OPC UA サーバーのアプリケーション証明書が返されます。(図 9 の手順 1) この時点では OPC UA クライアントの信頼リストには先ほど受け取った OPC UA サーバーのアプリケーション証明書と一致するものが登録されていないので、接続処理はそのままでは続けられないと判断します。OPC UA クライアントが OPC UA サーバーとの接続処理を行うには OPC UA サーバーのアプリケーション証明書を自身の信頼リストに登録しておく必要があるのです。登録するためには予め OPC UA サーバーのアプリケーション証明書を入手できれば良いのですが、それが出来ない場合の処理は OPC UA クライアントの仕様に委ねられます。

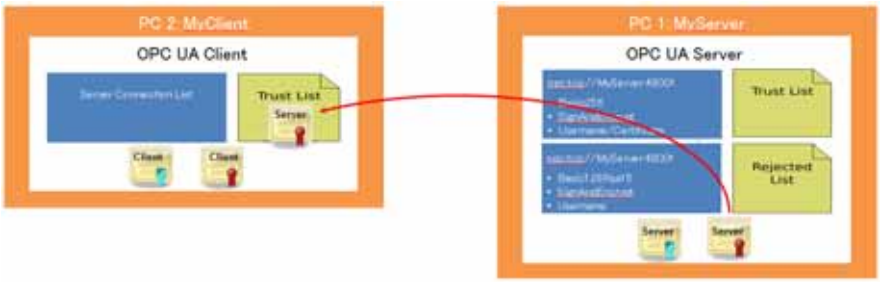
つぎの図で紹介するのは Unified Automation 社の UA Expert という OPC UA クライアントの例ですが、このアプリケーションは取得したアプリケーション証明書が信頼リストに無い場合には、その OPC UA サーバーを信頼するか否かを画面でユーザーに確認を求めます。(図 9 の 2) 信頼するとユーザーが応答すれば OPC UA サーバーのアプリケーション証明書を OPC UA クライアントの信頼リストに配置して、接続処理を継続する仕様になっています。(図 9 の 3)

この他にも、後に紹介する OPC UA サーバー側の運用例と同様な仕組みを提供する OPC UA クライアントもあるでしょう。 OPC UA クライアント側が OPC UA サーバーとの信頼関係を構築する手順は使用する OPC UA クライアントの仕様によるので、事前の確認が必要です。



手順 1：エンドポイントの取得

手順 2：証明書の許可



手順 3：OPC UA サーバーを信頼

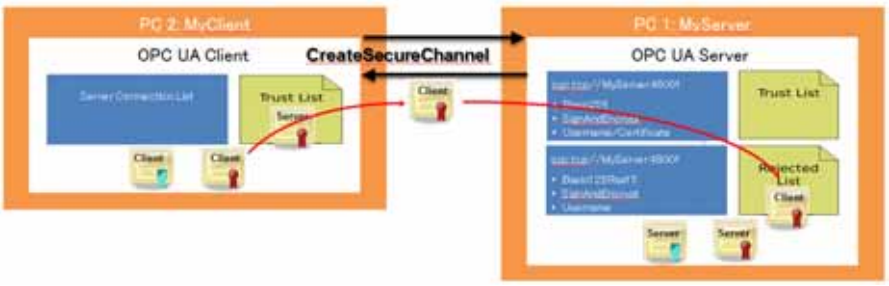
図 9 クライアントの設定

つぎに、OPC UA サーバーが OPC UA クライアントとの信頼関係を構築する手順の例を紹介します。

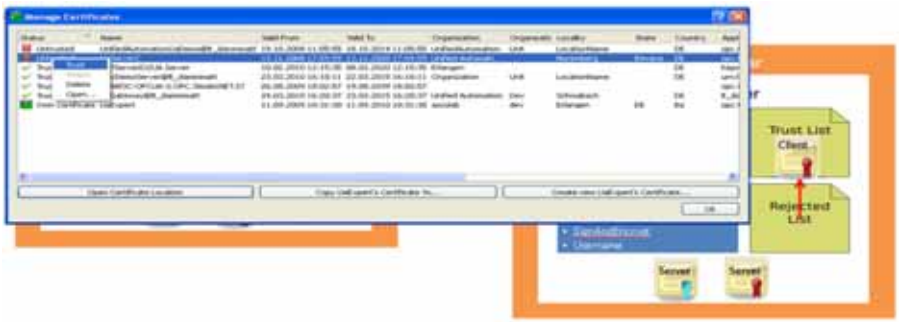
OPC UA クライアントは、OPC UA サーバーとの信頼関係を確認すると、つぎの接続処理として CreateSecureChannel というサービスを呼び出し、自身のアプリケーション証明書を渡します。証明書の扱いは先ほどの OPC UA クライアントのときと同じです。OPC UA サーバーは、自身が管理する信頼リストに OPC UA クライアントのアプリケーション証明書と一致するものが無いのでサービスの処理をエラーと判断し、接続を拒否した OPC UA クライアントのアプリケーション証明書を拒否リスト(Rejected List)に配置します。(図 10 の 1) この状態では、OPC UA クライアントが何度も接続を試みても、OPC UA サーバーは接続を拒否します。

接続を成功させるためには、拒否リストにあるアプリケーション証明書を信頼リストに移動させる必要があります。ここでは Unified Automation 社が提供するユーティリティツールの例を紹介します。この例では、管理者が OPC UA サーバー上でツールを操作して拒否リストにあるアプリケーション証明書を信頼リストに移動する様子を示しています。(図 10 の 2) これで OPC UA サーバーはその OPC UA クライアントとの信頼関係を構築します。

OPC UA サーバー側でアプリケーション証明書を信頼リストに登録する方法も、それぞれの OPC UA サーバーの仕様に委ねられるので、事前の確認が必要です。



手順 1：OPC UA クライアントからの接続を拒否



手順 2：OPC UA クライアントを信頼

図 10 サーバーの設定

参考までに、もしも OPC UA サーバーがセキュリティモード None であるエンドポイントを公開し、OPC UA クライアントがそのエンドポイントに接続要求してきた場合には、ここに紹介した処理は行われません。暗号鍵を用いたセキュリティ機能を必要としない場合には OPC UA サーバーは OPC UA クライアントと信頼関係を構築する手順は不要になります。

ここまでで OPC UA クライアントと OPC UA サーバーはお互いに信頼しあえることになりました。両者の信頼関係が構築された状態で OPC UA クライアントから接続を行うと、CreateSecureChannel サービスも正常に処理されます。ここでアプリケーション証明書の認証が完了しました。次の段階では、ユーザーの認証がおこなわれます。OPC UA クライアントはこの後の接続処理の中で、CreateSession というサービスに続いて呼び出す ActivateSession サービスでユーザー識別情報を渡します。つぎの図ではユーザー識別情報としてユーザー名とパスワードを渡していますが、これ以外にもユーザー証明書やトークンなどでユーザー情報を渡すことも OPC UA の仕様ではサポートされています。ユーザー認証に成功すれば接続処理は完了します。(図 11)

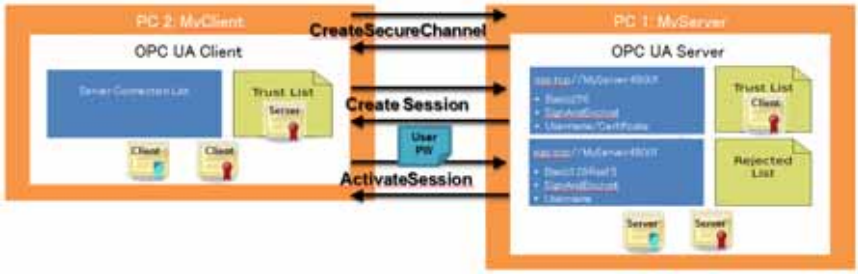


図 11 接続の完了

OPC UA クライアントと OPC UA サーバーのアプリケーション証明書のファイルを事前に入手できるのであれば、それらのアプリケーション証明書をお互いの信頼リストに配置することで、はじめから問題なく接続を成功させることができます。

ここでは Unified Automation 社の提供する OPC UA アプリケーションでの例を紹介しまし

たが、実際に使われる OPC UA アプリケーションの仕様に合わせてお互いの信頼関係を構築するようにしてください。

おわりに

OPC UA が提供するセキュリティ仕様の概要と、OPC UA アプリケーションを使用するためにはお互いの信頼関係を構築する必要があることを紹介しました。

OPC UA のセキュリティ機能の実装は、独自の技術を使用しているのではありません。攻撃者の技術は年々進歩し、これまで安全であったものもそうでなくなります。OPC UA のセキュリティの機能は一般的な IT 技術をつかっているので、時代の流れにしたがい新しい安全な技術を取り入れていくことが可能です。それは、OPC UA のコミュニケーションスタックの中で主に実装されるので、OPC UA アプリケーションのライフサイクルへの影響を及ぼしません。

一方で、これまでの OPC Classic 製品は多く存在します。OPC UA を世に出したからといって、OPC Classic のセキュリティがおざなりにされているのではありません。OPC Classic 製品は UA COM Wrapper/Proxy を使うことによって、OPC UA のセキュリティ機能を利用することができます。UA COM Wrapper/Proxy は OPC Foundation のサイトからダウンロードにより入手して評価することが可能ですし、サードパーティ製品も存在します。ご興味のある方は検討されることをお勧めします。