

# IEEE 1888 over WebSocket 仕様書

作成：2014年12月26日

更新：2016年2月29日

## はじめに

セキュリティ等の理由により通信が制限されたネットワーク内に存在する IEEE1888 コンポーネントに対して、そのネットワークの外部に存在する IEEE1888 コンポーネントからの接続を可能にしたいという要求があるが、従来の IEEE1888 の仕組みではそのような通信制限された環境における IEEE1888 コンポーネント同士による相互の接続や通信を実現することができない。

本仕様書では、そのような通信を制限されたネットワークに設置された IEEE1888 サーバクライアントと、そのネットワークの外部に設置された IEEE1888 サーバクライアントとの双方向性通信を実現する方法を定める。

通信制限のあるネットワーク環境下の機器との通信に関しては、いくつかの実現方法がある。本仕様書で定める方法では、通信制限のあるネットワーク環境下の IEEE1888 サーバの URL を代理する URL をそのネットワークの外部に用意し、かつ外部のネットワークにある IEEE1888 サーバの URL を代理する URL を通信制限のあるネットワーク内に用意することにより、双方のネットワークにある IEEE1888 コンポーネント同士を相互に到達可能にする方法を採用する。

また、代理 URL を提供するコンポーネント同士で WebSocket プロトコルを用いたコネクションを確立することで、通信の双方向性をもたせ、双方のネットワークにある機器からの相互アクセスを実現する。

本仕様書で定める方法は、IEEE1888 サーバの URL の代理、および閉じたネットワーク環境とその外部にあるネットワーク環境の間を結ぶ WebSocket コネクションを構築するための機器を追加するだけであり、IEEE1888 を使用した既存のネットワークに組み込みが容易である。

# 目次

IEEE 1888 over WebSocket 仕様書 .....	1
はじめに .....	1
目次 .....	2
1. 概要 .....	3
1.1. 本仕様書の目的 .....	3
1.2. 本仕様書の対象とする範囲 .....	3
1.3. 本仕様書に対する貢献者 .....	4
2. 用語 .....	4
3. アーキテクチャ .....	5
3.1. システム・アーキテクチャ .....	5
3.2. レイヤ・アーキテクチャ .....	8
4. プロキシの外部インターフェース .....	9
4.1. Global Proxy のプロキシ・サーバ・インターフェース .....	9
4.2. Global Proxy のプロキシ・クライアント・インターフェース .....	10
4.3. Local Proxy のプロキシ・サーバ・インターフェース .....	11
4.4. Local Proxy のプロキシ・クライアント・インターフェース .....	12
5. 通信手順 .....	12
5.1. 通信手順 .....	12
5.1.1. グローバルネットワーク側コンポーネントからローカルネットワーク側 コンポーネントへの通信 .....	13
5.1.2. ローカルネットワーク側コンポーネントからグローバルネットワーク側 コンポーネントへの通信 .....	15
5.1.3. TRAP 手順について .....	16
5.2. WebSocket コネクションの管理 .....	16
5.3. WSDL .....	17
6. プロキシ間のプロトコル .....	17
6.1. WebSocket コネクションの構築 .....	17
6.1.1. WebSocket コネクションを構築する際のエラー .....	17
6.2. WebSocket コネクションによる転送 .....	18
7. セキュリティに関する留意事項 .....	21
7.1. ローカルネットワークのセキュリティ .....	22
7.2. コンポーネント間の通信のセキュリティ .....	22
7.2.1. 限定的なコンポーネント間の通信のセキュリティ .....	22
7.3. 運用ポリシーの異なるネットワーク間の運用 .....	23
7.4. 変換表の整合性 .....	24

# 1. 概要

## 1.1. 本仕様書の目的

本仕様書では IEEE1888規格に対して、通常の方法では相互に通信することが難しいネットワーク環境での双方向通信を可能とすることで、より広い規格の活用を実現することを目的とする。

## 1.2. 本仕様書の対象とする範囲

本仕様書では IEEE1888規格に対して、以下にあげるような、外部ネットワーク側の機器からネットワーク内部の機器に通常の方法では接続できないが、ネットワーク内部から外部ネットワークにむけて HTTP 通信が可能なネットワーク環境において、相互に通信を行うための拡張を提供する。

- ・ プライベートネットワーク + NAT のネットワーク
- ・ ファイアウォールにより、外部から接続を開始できないネットワーク
- ・ 内部から外への通信が HTTP プロキシを介して行われるネットワーク
- ・ イン트라ネット

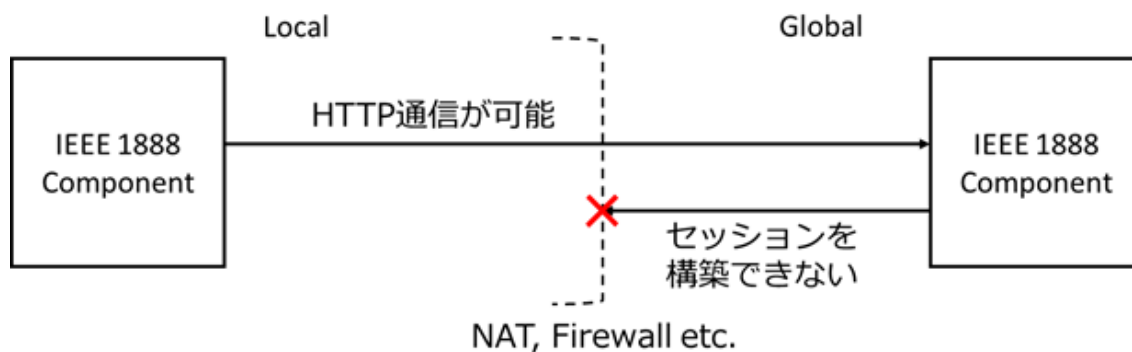


図 1 本仕様書で対象とするネットワークの一例

これらのネットワークを本仕様書では、ローカルネットワークと定義する。本仕様書で対象とする IEEE1888 over WebSocket は、

- (1) ローカルネットワーク内に設置された IEEE1888コンポーネントと、外部ネットワークにある IEEE1888コンポーネントとを相互に通信できるようにする
- (2) 相互通信のために必要な WebSocket 通信のセッションが維持される仕組みを持つ
- (3) IEEE1888システム・アーキテクチャに組み込むことができる

ものである。

本仕様書では、以下のことに関しては対象としない。

- ・ メッセージ圧縮
- ・ 遅延

### 1.3. 本仕様書に対する貢献者

本仕様書は以下に示す各氏の貢献により作成された。(敬称略)

天辰健一 k.amatatsu@comzeit.co.jp コムツアイト  
 池上洋行 ikegam@hongo.wide.ad.jp 東京大学  
 エンヘー テムーレン e-temu@ist.osaka-u.ac.jp 大阪大学  
 落合秀也 jo2lxq@hongo.wide.ad.jp 東京大学  
 坂根昌一 ssakane@cisco.com シスコシステムズ  
 田内康 tauchi@yamaguchi-u.ac.jp 山口大学  
 樽谷優弥 y-tarutn@cmc.osaka-u.ac.jp 大阪大学  
 松田和浩 kazuhiko.matsuda@ane.cmc.osaka-u.ac.jp 大阪大学  
 村田修一郎 mrt@acutus.co.jp 大阪大学  
 (五十音順)

## 2. 用語

ローカルネットワーク	ファイアウォールや NAT で外部ネットワークから切り離されたネットワーク
グローバルネットワーク	ファイアウォールや NAT 等で外部ネットワークから切り離されたネットワークに対する外部ネットワーク
Local Proxy	本仕様書において、ローカルネットワークに配置することで、ローカルネットワーク内にある IEEE1888対応コンポーネントとグローバルネットワーク側にある IEEE1888対応コンポーネントとの相互通信を実現するための機器またはサービス
Global Proxy	本仕様書において、グローバルネットワークに配置することで、グローバルネットワークにある IEEE1888対応コンポーネントとローカルネットワーク内にある IEEE1888対応コンポーネントとの相互通信を実現するための機器またはサービス
コンポーネント・アクセス・インターフェース	IEEE1888対応コンポーネントが提供する機能呼び出すための RPC 受信用インターフェース
プロキシ・クライアント・インターフェース	本仕様書において、Global および Local Proxy が持つ他の IEEE1888対応コンポーネントの機能呼び出すための RPC 送信用インターフェース
プロキシ・サーバ・インターフェース	本仕様書において、Global および Local Proxy が提供する RPC 受信用インターフェース。他の IEEE1888対応コンポー

ネットからはコンポーネント・アクセス・インターフェースと同様に呼び出すことが出来る
---

### 3. アーキテクチャ

#### 3.1. システム・アーキテクチャ

IEEE1888 over WebSocket によるシステムは、最も単純な場合で図 2 のように構成される。図 2 に示すように、IEEE1888 over WebSocket では、ローカルネットワーク環境下の IEEE1888 コンポーネントとグローバルネットワーク側コンポーネントとの相互通信のために、ローカルネットワークとグローバルネットワークとを、WebSocket プロトコルを用いたコネクション（以下 WebSocket コネクション）で接続する。WebSocket コネクションにより、ローカルネットワーク環境下の IEEE1888 コンポーネントに対して、グローバルネットワーク側の IEEE1888 コンポーネントから通信を行うことができる。このグローバルネットワーク側とローカルネットワークの間に WebSocket コネクションを構築する機器またはサービスをそれぞれ Global Proxy、Local Proxy と定める。

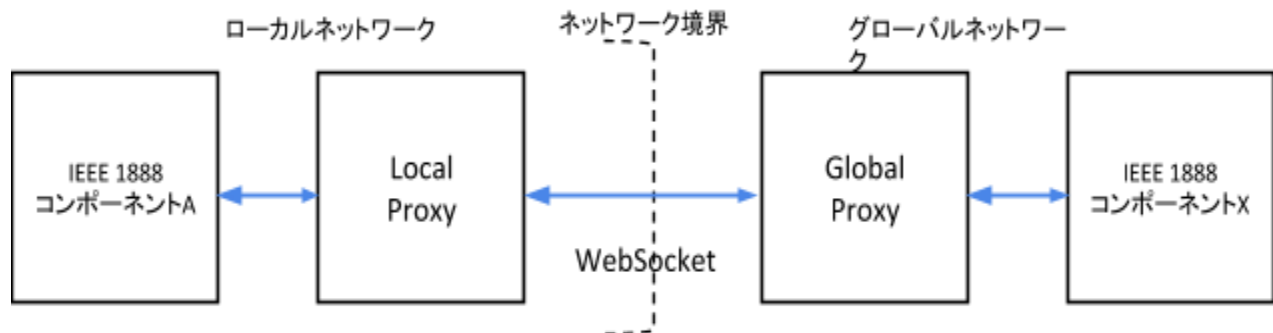


図 2 IEEE1888 over WebSocket で構成される単純なシステム

図 3 はより一般的な IEEE1888 over WebSocket で構成されるシステムである。Global Proxy はグローバルネットワーク側に設置され、WebSocket 用の URL を持っている。それとは別に、ローカルネットワーク環境下の各 IEEE1888 コンポーネントに対応する複数の Proxy 用の URL を持っている。各 URL 宛に送られた通信は WebSocket コネクションを通じて Local Proxy に転送され、対応する IEEE1888 コンポーネントに送信される。このように Global Proxy がローカルネットワーク環境下の IEEE1888 コンポーネントの代理をすることによって、図 4 のようにグローバルネットワーク側から、擬似的にローカルネットワーク環境下の IEEE1888 コンポーネントを見ることが出来る状態にする。グローバルネットワーク側にあるコンポーネントから各コンポーネントが図 4 のように見える。図中の実線と点線で表されたコンポーネントは、それぞれ直接アクセスが可能なコンポーネント、Global Proxy を介してアクセスすべきコンポーネントである。

Local Proxy は外部から閉じたネットワーク環境下に設置される。Local Proxy は Global Proxy との間に WebSocket コネクションを構築し、Global Proxy から転送された通信を、送られたプロキシ用の URL に対応する NAT 配下の IEEE1888 コンポーネントに送信する。

Local Proxy は Global Proxy と同様に Proxy 用 URL を提供する機能を持つ。図 3において Local Proxy1が提供している「URL(X用)」「URL(D用)」、および Local Proxy2が提供している「URL(Y用)」がそれぞれである。Local Proxy1や Local Proxy2が設置されているネットワークの各コンポーネントからは他のコンポーネントがそれぞれ図 5, 図 6のように見える。実線で表されたコンポーネントは直接アクセスできるコンポーネント、点線で表されたコンポーネントは Local Proxy を介してアクセスできるコンポーネントである。

互いにアクセス可能な状態にあるコンポーネント同士は、Global Proxy, Local Proxy の組み合わせを介さずに直接アクセスしてもよい。

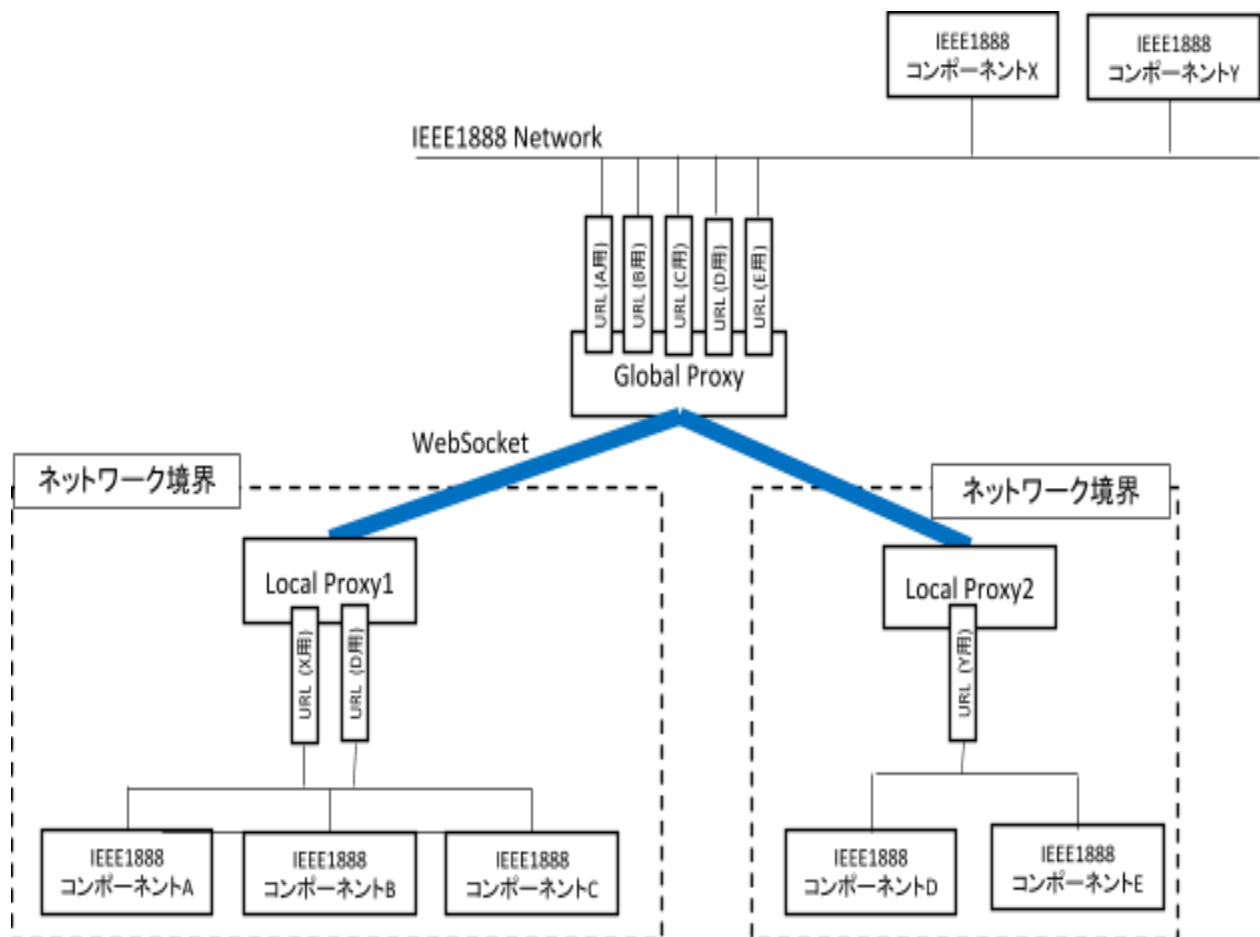


図 3 IEEE1888 over WebSocket で構成されるシステムのアーキテクチャの例

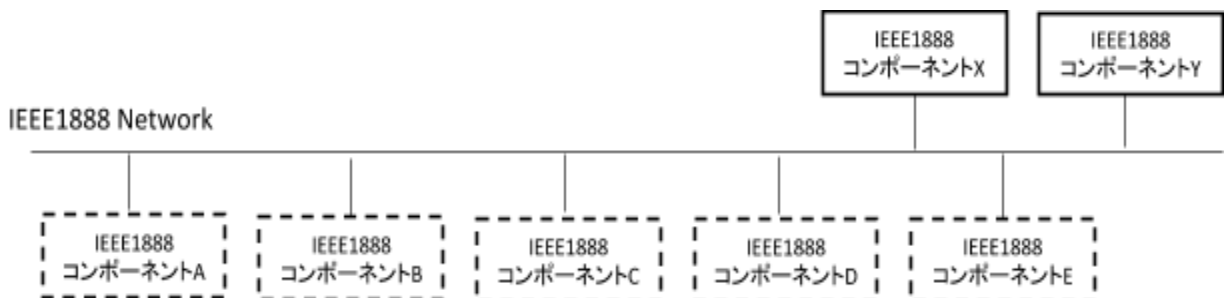


図 4 グローバルネットワーク側コンポーネントからみた図 3の IEEE1888通信網

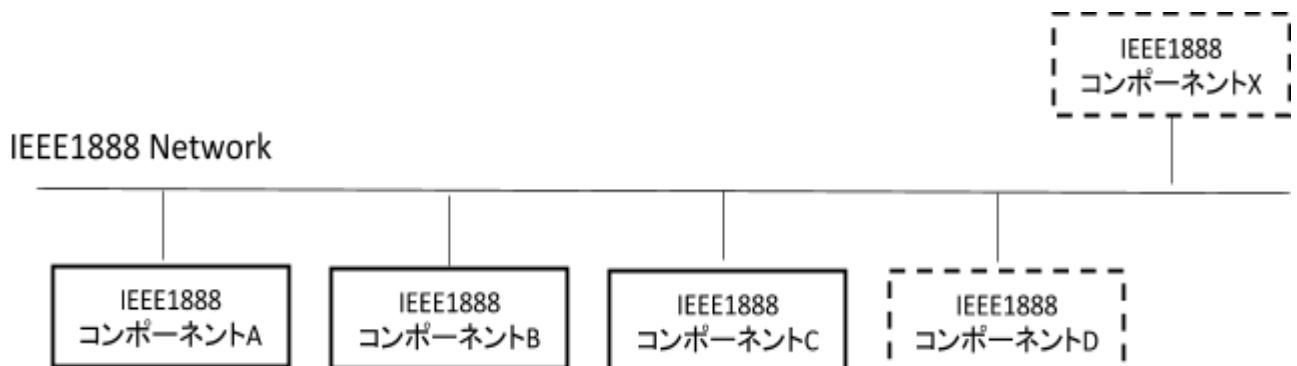


図 5 Local Proxy1のネットワークにあるコンポーネントからみた図 3の IEEE1888通信網

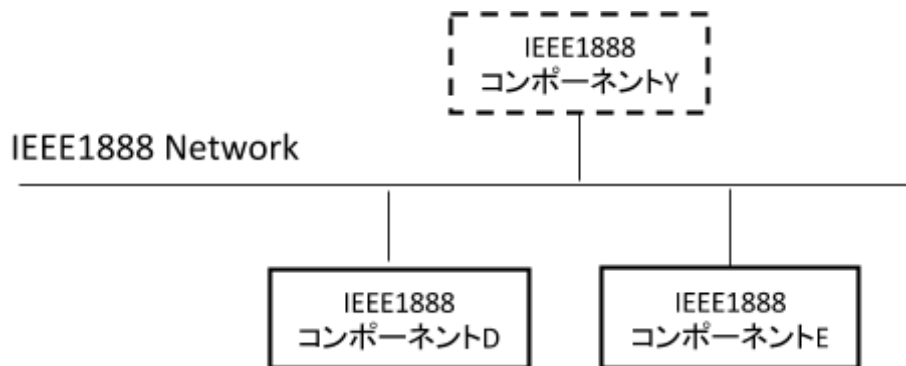


図 6 Local Proxy2のネットワークにあるコンポーネントからみた図 3の IEEE1888通信網

### 3.2. レイヤ・アーキテクチャ

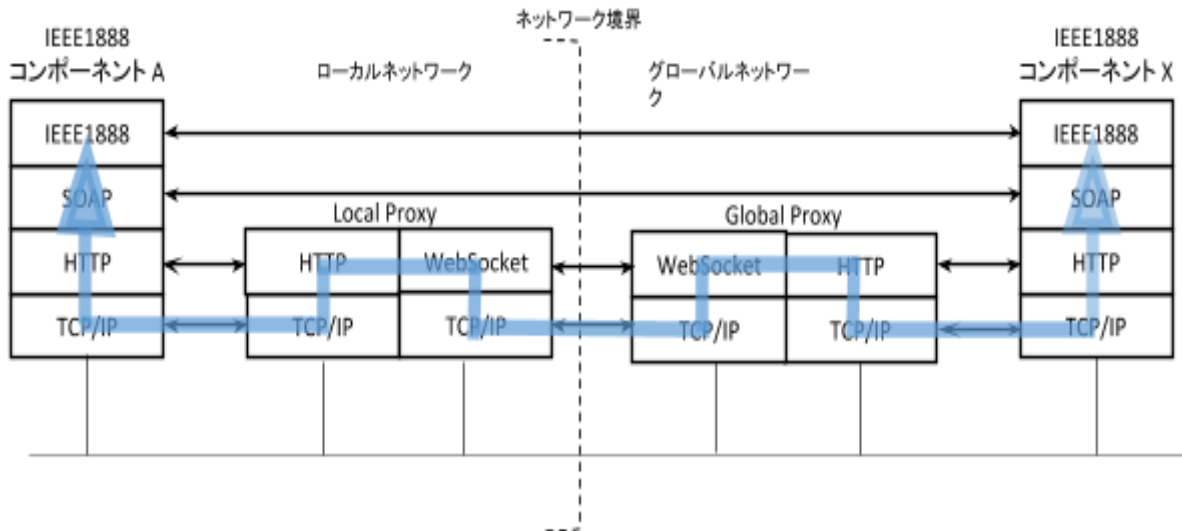


図 7 IEEE1888 over WebSocket のレイヤ・アーキテクチャ

IEEE1888 over WebSocket ではコンポーネント-プロキシ間と Global Proxy-Local Proxy 間(Proxy 間)で異なる通信方法が用いられる。コンポーネント-プロキシ間では通常の IEEE1888 におけるコンポーネント間の通信を行う。Proxy 間では、TCP/IP のセッションを構築し、HTTP リクエストによって接続の UPGRADE 要求を行い、Proxy 間の TCP/IP のセッション上に WebSocket コネクションを構築する。

WebSocket コネクションの構築後、各コンポーネントから受信した HTTP メッセージを Proxy 間の接続を通して転送することにより通信を実現する。

本仕様書における IEEE1888 over WebSocket のレイヤ・アーキテクチャを図 7 に示す。図 7 に示すように IEEE1888、SOAP のレイヤでは、通信の方向にかかわらず、各コンポーネントから送られてきたメッセージはプロキシを通じて宛先のコンポーネントに送られる。通常、各プロキシでは、これらのレイヤのメッセージは処理を行うことなく、透過的に宛先のコンポーネントへ送信される。

プロキシは HTTP ヘッダーを参照し、どのプロキシやコンポーネントにメッセージを転送するかを決定する。図 7 において、コンポーネント X からコンポーネント A への通信の場合、宛先の URL は Global Proxy のもつ URL であり、Global Proxy はそれを参照して、Global Proxy がどの Local Proxy に転送するか、また Local Proxy がどのコンポーネントに送信するかを決定する。Local Proxy も Global Proxy と同様の転送用 URL を提供し、宛先を実際のコンポーネントの URL に変換したのち、Global Proxy に転送する。

WebSocket レイヤでは、コンポーネントから受け取った HTTP メッセージを WebSocket の Data フレームに格納して対象のプロキシに転送する。



## 4. プロキシの外部インターフェース

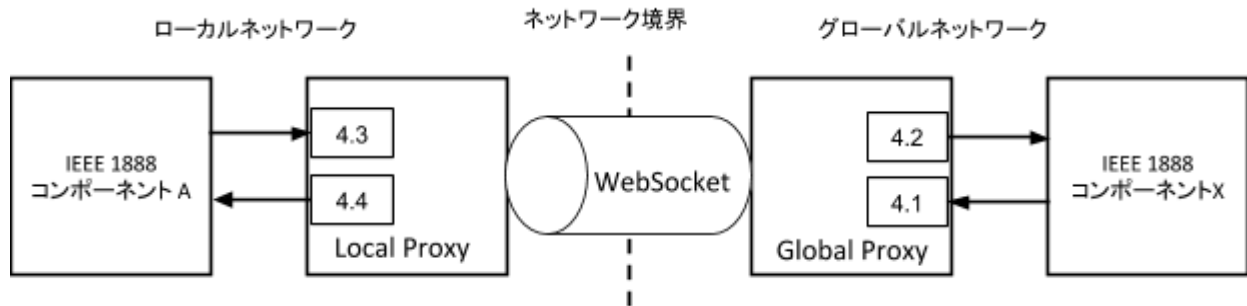


図 8 プロキシの外部インターフェース

本章ではプロキシの外部インターフェースについて記述する。図 8は各節がそれぞれの部分のインターフェース仕様を記述しているかを示している。本仕様書では「Global Proxy のプロキシ・サーバ・インターフェース Global Proxy のプロキシ・サーバ・インターフェース」と「Local Proxy のプロキシ・サーバ・インターフェース Local Proxy のプロキシ・サーバ・インターフェース」と、また「Global Proxy のプロキシ・クライアント・インターフェース Global Proxy のプロキシ・クライアント・インターフェース」と「Local Proxy のプロキシ・クライアント・インターフェース Local Proxy のプロキシ・クライアント・インターフェース」とはそれぞれ同一の仕様である。

### 4.1. Global Proxy のプロキシ・サーバ・インターフェース

Global Proxy のプロキシ・サーバ・インターフェースは、グローバルネットワーク側の IEEE1888コンポーネントからローカルネットワーク環境下にある複数の IEEE1888コンポーネント・アクセス・インターフェースへの接続を仲介する。Global Proxy はローカルネットワーク環境下の IEEE1888コンポーネント・アクセス・インターフェースを代理するための複数の URL を持っている。各 URL はローカルネットワーク環境に複数ある IEEE1888コンポーネント・アクセス・インターフェースに一対一で対応している(表 1)。それらの IEEE1888サービス URL に対して送信される HTTP メッセージは、Local Proxy を通じて、その URL に対応するローカルネットワーク環境下の IEEE1888コンポーネント・アクセス・インターフェースに透過的に転送される。そして、ローカルネットワーク環境下の IEEE1888コンポーネント・アクセス・インターフェースから応答される IEEE1888通信(SOAP メッセージを含む、HTTP メッセージ)が、返答される。

Global Proxy が提供するプロキシ・サーバ・インターフェースのエンドポイント	実際の接続先 IEEE1888コンポーネント・アクセス・インターフェースのエンドポイント URL
http://globalproxy.example.com/A	http://local-ieee1888.exmaple.com/A
http://globalproxy.example.com/B-8888	http://local-ieee1888.example.com:8888/B

http://globalproxy.example.com/192-C	http://192.168.0.10/C
--------------------------------------	-----------------------

表 1 Global Proxy のエンドポイント変換表の例

Global Proxy のプロキシ・サーバ・インターフェースはグローバルネットワーク側の IEEE1888クライアントから送信された HTTP メッセージを受け取り、受け取ったメッセージを WebSocket の Data フレームに格納し、WebSocket を用いて Local Proxy に転送する。

Global Proxy は Local Proxy へメッセージを転送する際、HTTP メソッドのパス部分および Host ヘッダーを実際の送信先に書き換える。

また、Global Proxy のプロキシ・サーバ・インターフェースへのリクエスト中に発生したエラーは、次の形で IEEE1888クライアントに対して返答される。エラーコードは特記のない場合 HTTP 1.1のステータスコード(RFC2616)である。

- ・ 要求メッセージを送信しようとしたときに、Local Proxy が利用できない場合
  - WebSocket コネクションが切断されている場合: 503 (Service Unavailable)
  - WebSocket コネクションは切断されていないが、Local Proxy がリクエストを処理できない : 502 (Bad Gateway):
  - WebSocket コネクションは切断されておらず、Local Proxy に要求メッセージは届くが、応答がなく接続要求のタイムアウトが発生 : 504 (Gateway Timeout)
  - 要求メッセージまたは宛先コンポーネントからの返信のサイズが大きすぎて転送出来ない場合: 413 (Payload too large)
- ・ 要求メッセージを送信しようとしたときに、宛先のコンポーネントが利用できない場合
  - 宛先コンポーネントが HTTP のステータスコードを返した場合: そのステータスコード
  - 宛先コンポーネントの返答がなかった(タイムアウトした)場合: 504 (Gateway Timeout)
  - 宛先コンポーネントが HTTP ではないレスポンスを返した場合:502 (Bad Gateway)

また、グローバルネットワーク側から、ローカルネットワーク環境下 IEEE1888サービス URL に対する WSDL 要求は、Global Proxy が IEEE1888の WSDL の<soap:address>の location 属性を Global Proxy が提供するエンドポイント URL へ書き換えたうえで返答する。

## 4.2. Global Proxy のプロキシ・クライアント・インターフェース

Global Proxy のプロキシ・クライアント・インターフェースはローカルネットワーク側の IEEE1888クライアントを代理し、Local Proxy に対する通信をきっかけとし、この Global Proxy のプロキシ・クライアント・インターフェースから対象となる

IEEE1888コンポーネント・アクセス・インターフェースに対しての通信を開始する。Local Proxy に書き込まれた HTTP メッセージを、通信対象となっている IEEE1888 コンポーネント・アクセス・インターフェースに送り、その応答を受信する。なお、受信した HTTP 応答メッセージは Local Proxy からローカルネットワーク側の IEEE1888クライアントに返される(Local Proxy のプロキシ・サーバ・インターフェース節参照)。

Global Proxy はリクエストの送出時に Forwarded:ヘッダー(RFC7239)を追加することができる。

### 4.3. Local Proxy のプロキシ・サーバ・インターフェース

Local Proxy のプロキシ・サーバ・インターフェースは Global Proxy のプロキシ・サーバ・インターフェースと同様、グローバルネットワーク側にある IEEE1888コンポーネント・アクセス・インターフェースを代理するための URL を提供する(図 3 「URL(X 用)」)。代理となる URL と実際の接続先との変換表は Global Proxy の場合と同様に表 2 のようになる。この HTTP プロキシのクライアント側のインターフェースは、Global Proxy のプロキシ・クライアント・インターフェースである。

Local Proxy が提供するプロキシ・サーバ・インターフェースのエンドポイント	実際の接続先 IEEE1888コンポーネント・アクセス・インターフェースのエンドポイント URL
http://localproxy.example.com/A	http://global-ieee1888.example.com/A
http://localproxy.example.com/B-8888	http://global-ieee1888.example.com:8888/B
http://localproxy.example.com/1-C	http://global-ieee1888-1.example.com/C

表 2 Local Proxy のエンドポイント変換表の例

Local Proxy のプロキシ・サーバ・インターフェースはローカルネットワーク側の IEEE クライアントから送信された HTTP メッセージを受け取り、受け取ったメッセージを WebSocket の Data フレームに格納し、WebSocket を用いて Global Proxy に転送する。

Local Proxy のプロキシ・サーバ・インターフェースへのリクエスト中に発生したエラーは、次の形で IEEE1888クライアントに対して返答される。エラーコードは特記のない場合 HTTP 1.1のステータスコード(RFC2616)である。

- ・ 要求メッセージを送信しようとしたときに、Global Proxy が利用できない場合
  - WebSocket コネクションが切断されている場合: 503 (Service Unavailable)
  - WebSocket コネクションは切断されていないが、Global Proxy がリクエストを処理できない: 502 (Bad Gateway)

- WebSocket コネクションは切断されておらず、Global Proxy に要求メッセージが届くが、応答がなく接続要求のタイムアウトが発生：504 (Gateway Timeout)
- 要求メッセージまたは宛先コンポーネントからの返信のサイズが大きすぎて転送出来ない場合：413 (Payload too large)
- 要求メッセージを送信しようとしたときに、宛先のコンポーネントが利用できない場合
  - 宛先コンポーネントが HTTP のステータスコードを返した場合：そのステータスコード
  - 宛先コンポーネントの返答がなかった(タイムアウトした)場合：504 (Gateway Timeout)
  - 宛先コンポーネントが HTTP ではないレスポンスを返した場合：502 (Bad Gateway)

#### 4.4. Local Proxy のプロキシ・クライアント・インターフェース

Local Proxy のプロキシ・クライアント・インターフェースはグローバルネットワーク側の IEEE1888クライアントを代理し、Global Proxy に対する通信をきっかけとし、この Local Proxy のプロキシ・クライアント・インターフェースから対象となる IEEE1888コンポーネント・アクセス・インターフェースに対しての通信を開始する。Global Proxy に書き込まれた HTTP メッセージを、通信対象となっている IEEE1888コンポーネント・アクセス・インターフェースに送り、その応答を受信する。なお、受信した HTTP 応答メッセージは Global Proxy からグローバルネットワーク側の IEEE1888クライアントに返される(Global Proxy のプロキシ・サーバ・インターフェース節参照)。

Local Proxy はリクエストの送出時に Forwarded:ヘッダー(RFC7239)を追加することができる。

## 5. 通信手順

### 5.1. 通信手順

本節では、図 9に示す IEEE1888 over WebSocket のネットワークにおいて、各コンポーネントがどのように通信を行うかを解説する。

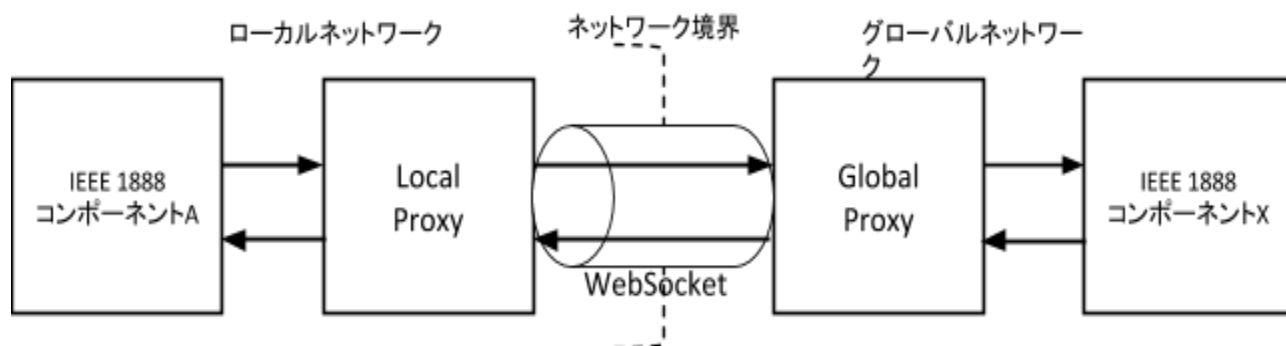


図 9 IEEE1888 over WebSocket の通信

### 5.1.1. グローバルネットワーク側コンポーネントからローカルネットワーク側コンポーネントへの通信

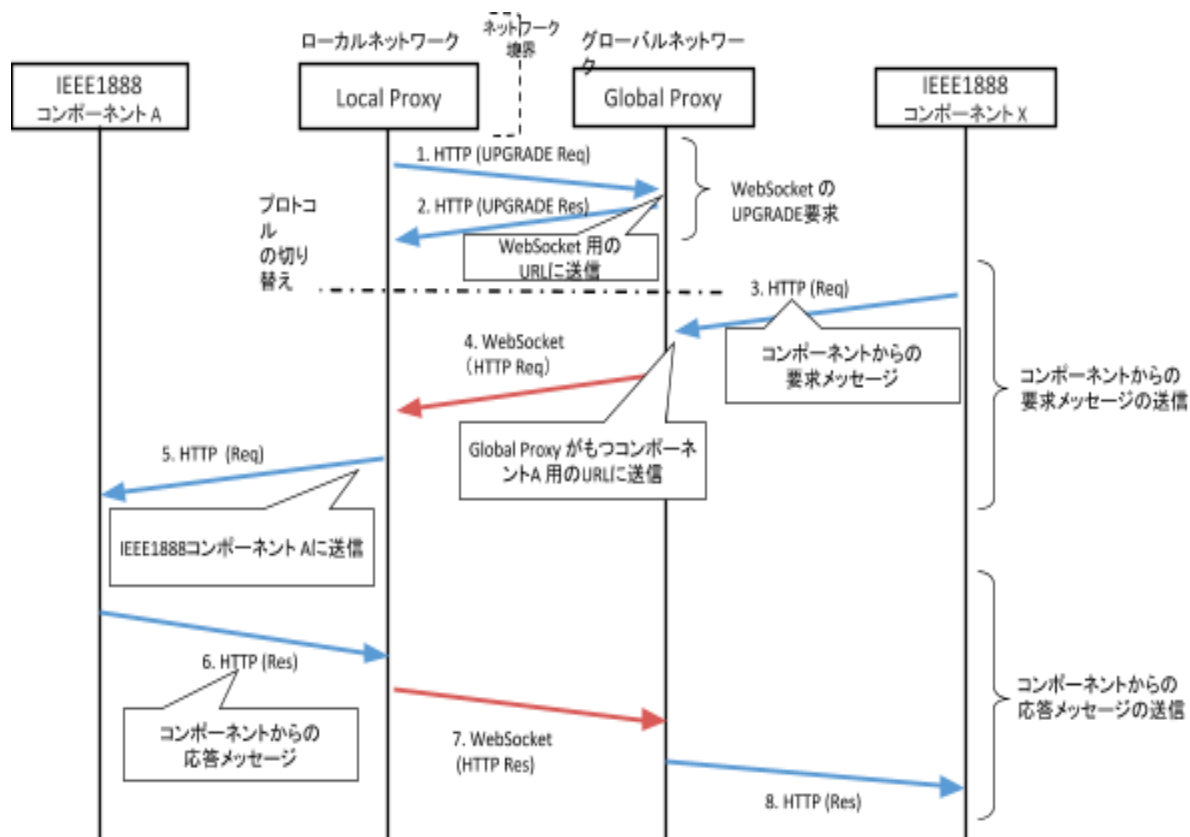


図 10 IEEE1888 over WebSocket の通信手順の例(グローバルネットワーク側からの場合)

図 10はグローバルネットワーク側のコンポーネント X からローカルネットワーク側のコンポーネント A への通信手順である。以下に各手順の具体的な動作内容を示す。

1. Local Proxy が Global Proxy に対して、WebSocket コネクションの接続要求 (UPGRADE Req) を送信する。

2. Global Proxy は Local Proxy に UPGRADE Res を返す。結果として、Local Proxy、Global Proxy 間に WebSocket コネクションが確立される。
3. IEEE1888コンポーネント X は Global Proxy が持つコンポーネント A 用の URL 宛に要求メッセージを送信する。
4. Global Proxy は受け取った要求メッセージを WebSocket のコネクションを用いて Local Proxy に転送する。
5. Local Proxy は IEEE1888コンポーネント A に対して、要求メッセージを送信する。
6. IEEE1888コンポーネント A は受信した要求メッセージに対しての応答メッセージを、Local Proxy に返す。
7. Local Proxy は WebSocket コネクションを通じて応答メッセージを Global Proxy に転送する。
8. Global Proxy は受け取った応答メッセージを IEEE1888コンポーネント X に対して送信する。

### 5.1.2. ローカルネットワーク側コンポーネントからグローバルネットワーク側コンポーネントへの通信

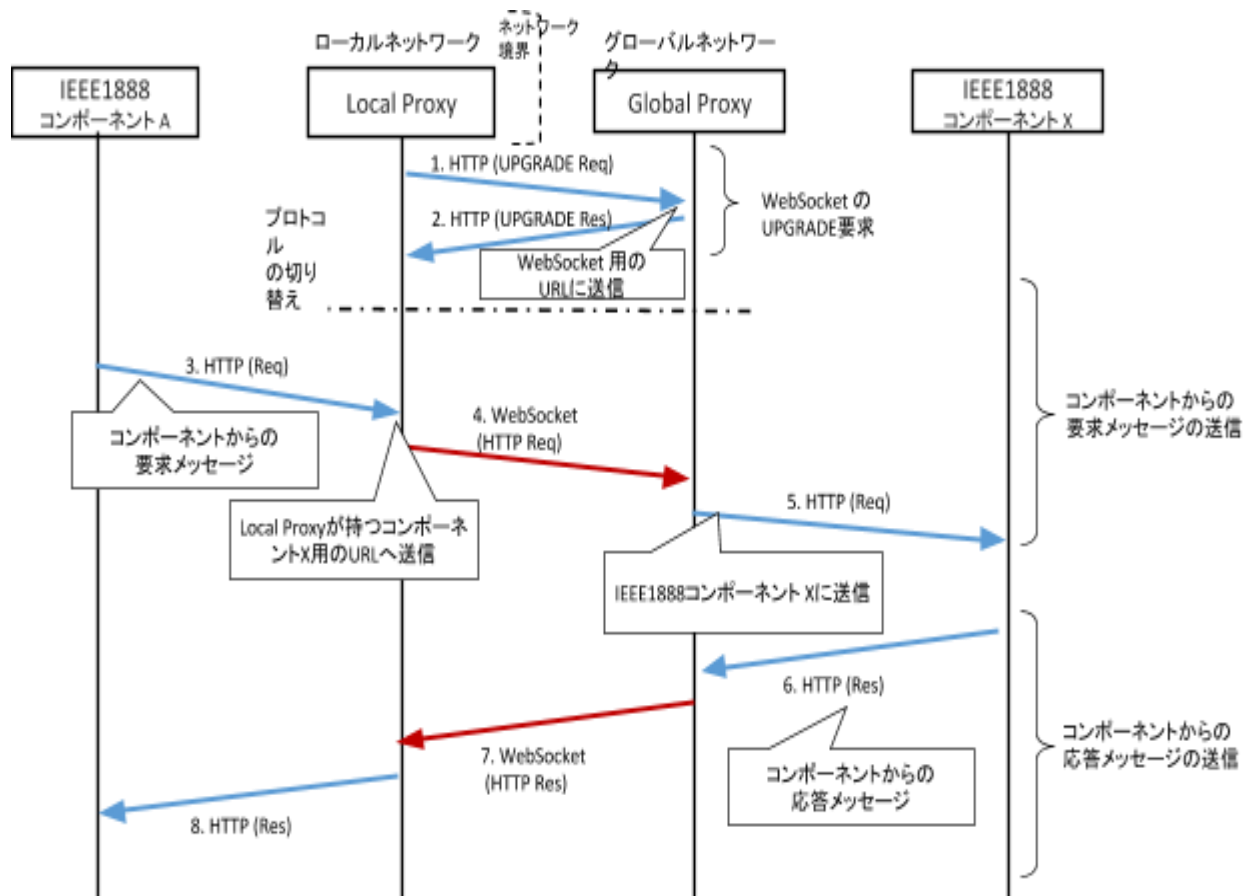


図 11 IEEE1888 over WebSocket の通信手順の例(ローカルネットワーク側からの場合)

図 11はローカルネットワーク側のコンポーネント A からグローバルネットワーク側のコンポーネント X への通信手順である。各手順の具体的な動作内容は以下の通りである。

1. Local Proxy が Global Proxy に対して、WebSocket コネクションの接続要求 (UPGRADE Req) を送信する。
2. Global Proxy は Local Proxy に UPGRADE Res を返す。結果として、Local Proxy、Global Proxy 間に WebSocket コネクションを確立する。
3. IEEE1888コンポーネント A は Local Proxy がもつコンポーネント X 用の URL 宛に要求メッセージを送信する。
4. Local Proxy は WebSocket を通して Global Proxy に要求メッセージを転送する。
5. Global Proxy は転送された要求メッセージを IEEE1888コンポーネント X に送信する。

6. 要求メッセージを受け取った IEEE1888コンポーネント X は Global Proxy に対して応答メッセージを返す。
7. Global Proxy は Local Proxy に対して応答メッセージを転送する。
8. Local Proxy は転送された応答メッセージを IEEE1888コンポーネント A に対して返送する。

### 5.1.3. TRAP 手順について

グローバルネットワーク側コンポーネントからローカルネットワーク側コンポーネントへの通信節およびローカルネットワーク側コンポーネントからグローバルネットワーク側コンポーネントへの通信節に示す通信手順によって、コンポーネント A とコンポーネント X の双方向通信が実現される。コンポーネントが TRAP 要求を出す場合は、あらかじめサーバインターフェイスが公開している Callback 用の URL を使用する。

## 5.2. WebSocket コネクションの管理

WebSocket コネクションの生成は、図 10 および図 11 に示しているように Local Proxy から Global Proxy の WebSocket 用 URL に対して、コネクション接続要求を行うことで生成される。WebSocket コネクションはそれぞれのプロキシによって以下のような管理がされている。

Global Proxy :

- 接続している Local Proxy の管理
- コネクションの削除

Local Proxy :

- コネクションの構築
- コネクションの維持・再構築
- コネクションの削除

Local Proxy は任意のタイミングで Global Proxy に対し コネクション要求を行う。Global Proxy は WebSocket コネクションの構築要求に対して、認証を行うよう実装しても良いが、本仕様書では対象外とする。コネクション切断時の再構築は、Local Proxy が適切なタイミングで行う。コネクションの維持は、Local Proxy が Global Proxy に対し WebSocket の Ping フレームを送信することで行う。



### 5.3. WSDL

Global および Local Proxy を問わず、Proxy 用 URL への WSDL 要求は、その Proxy 用 URL を提供する Proxy が返答する。各 Proxy は、返すべき WSDL を、IEEE1888 の WSDL と、みずから提供している Proxy 用 URL とを組み合わせで生成する。

## 6. プロキシ間のプロトコル

本章では、プロキシ間の WebSocket コネクションにおけるプロトコルおよびデータ形式について記述する。

### 6.1. WebSocket コネクションの構築

各プロキシは以下の手順で WebSocket コネクションを構築する。

1. Local Proxy は起動後、Global Proxy の WebSocket 用 URL に UPGRADE 要求を送信
2. Global Proxy は Local Proxy の UPGRADE 要求に対し、応答を返す
3. Local Proxy、Global Proxy 間に WebSocket を構築する

Global Proxy の WebSocket 用 URL が `ws://<global proxy hostname>/example` とした場合、実際に送られるリクエスト、レスポンスは以下のようなになる。

#### \* REQUEST

```
GET /example HTTP/1.1
Host: <global proxy host name>
Upgrade: websocket
Connection: Upgrade
Origin: <local proxy name>
```

#### \* RESPONSE

```
HTTP/1.1 101 Switching Protocols
Upgrade: websocket
Connection: Upgrade
```

ヘッダー `Origin` は接続する Local Proxy の名前を表し、Global Proxy はこの名前をもとに、リクエストを転送する先を判断する。Origin の内容は URL 形式とし、Global Proxy の管理者が Local Proxy 名の競合が起きないように管理する。

#### 6.1.1. WebSocket コネクションを構築する際のエラー

Local Proxy が Global Proxy に対して WebSocket コネクションを構築する際に発生するエラーを以下に示す。エラーコードは特記のない場合 HTTP 1.1 のステータスコード (RFC2616) である。

- 通常の WebSocket のエラーに準拠する
- Global Proxy が存在していないかまたはポートが閉じており、UPGRADE 要求を送信できない：コネクションタイムアウト等が発生する。このエラーは HTTP のステータスコードではない
- Global Proxy が WebSocket に対応していない：501 (Not Implemented)
- なんらかの理由（例：WS コネクション構築数の制限）で、Global Proxy が UPGRADE リクエストを拒否：503 (Service Unavailable)
- （認証がある場合）Global Proxy における認証失敗：401(Unauthorized)

## 6.2. WebSocket コネクションによる転送

各プロキシはコンポーネントから送信された HTTP メッセージを対応する適切なプロキシに WebSocket コネクションのテキストフレームを用いて転送する。転送するメッセージはコンポーネントから送られてきた HTTP メッセージに対し転送する複数のメッセージの区別のためメッセージ管理部が付加される。ここで、「HTTP メッセージ」とは「メソッド」「ステータスライン」「ヘッダー」「ボディ」を含むメッセージ全体のことである。IEEE1888コンポーネントからの HTTP メッセージが chunk 形式であった場合は、Proxy による受信時にまとめられ、転送される。このとき、Content-Length ヘッダーが付与されたものになる。

メッセージ管理部には ID (TransactionID) およびその TransactionID を生成したプロキシの名前(TransactionOrigin)が含まれる。

TransactionID は、リクエストを転送する Proxy が、転送されるリクエストひとつごとに異なる文字列として付与し、Proxy がリクエスト/レスポンスの対応関係を識別するために使われる。

TransactionOrigin はリクエストを転送する側の Proxy が、Proxy ごとに決めた一定の文字列を付与し、受信したメッセージがリクエストなのかレスポンスなのかを識別するために使われる。つまり、受け取ったメッセージの TransactionOrigin が自らが付与する TransactionOrigin の文字列と異なる場合は、そのメッセージは他の Proxy から転送されたリクエストである。また、受け取ったメッセージの TransactionOrigin が自らの付与する TransactionOrigin の文字列である場合は、そのメッセージは他の Proxy から返送されたレスポンスである。

そのため転送されたリクエストに対してレスポンスを返送する場合、もとのリクエストに付与されていたのと同じの TransactionID および TransactionOrigin を付与する。各プロキシは、どの TransactionOrigin を持ったメッセージがどのコンポーネントから送られてきたメッセージで、どの WebSocket コネクションを用いて転送されたものかの対応が取れるように管理する。

TransactionID は36文字までの任意のテキストである。

TransactionOrigin は URL 形式とし、Local Proxy の TransactionOrigin は WebSocket コネクション構築時の Origin と同一の文字列とする。Global Proxy の TransactionOrigin は各 Local Proxy の TransactionOrigin と異なる文字列を付与するものとする。

メッセージ管理部は各ヘッダ行を”\r\n”(キャリッジリターンおよびラインフィード)で区切って構成され、最後は”\r\n”の繰り返しで終わる。(図 12, 図 13, 図 14) ヘッダの出現順序は任意とする。(ヘッダ行の構造については RFC7230 3.2 節に従う) 定義のないヘッダ行は無視してよい。

```
TransactionOrigin: <Name of Transaction Originator>\r\n
TransactionID: <ID>\r\n
\r\n
```

図 12 メッセージ管理部

実際に WebSocket コネクション 上を流れるメッセージは図 13 および図 14 のようになる。

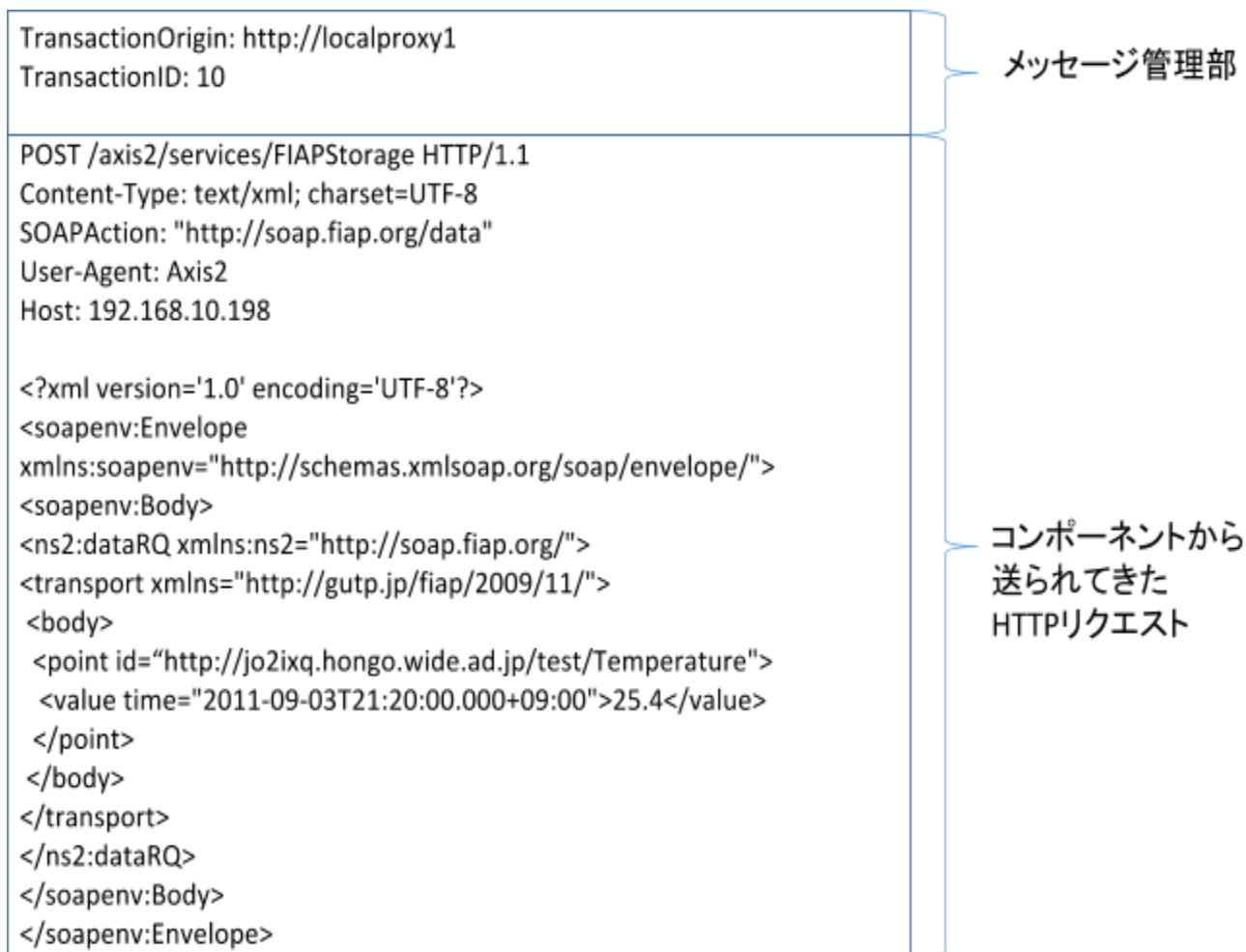


図 13 WebSocket コネクション上を流れるメッセージ例(リクエスト転送)

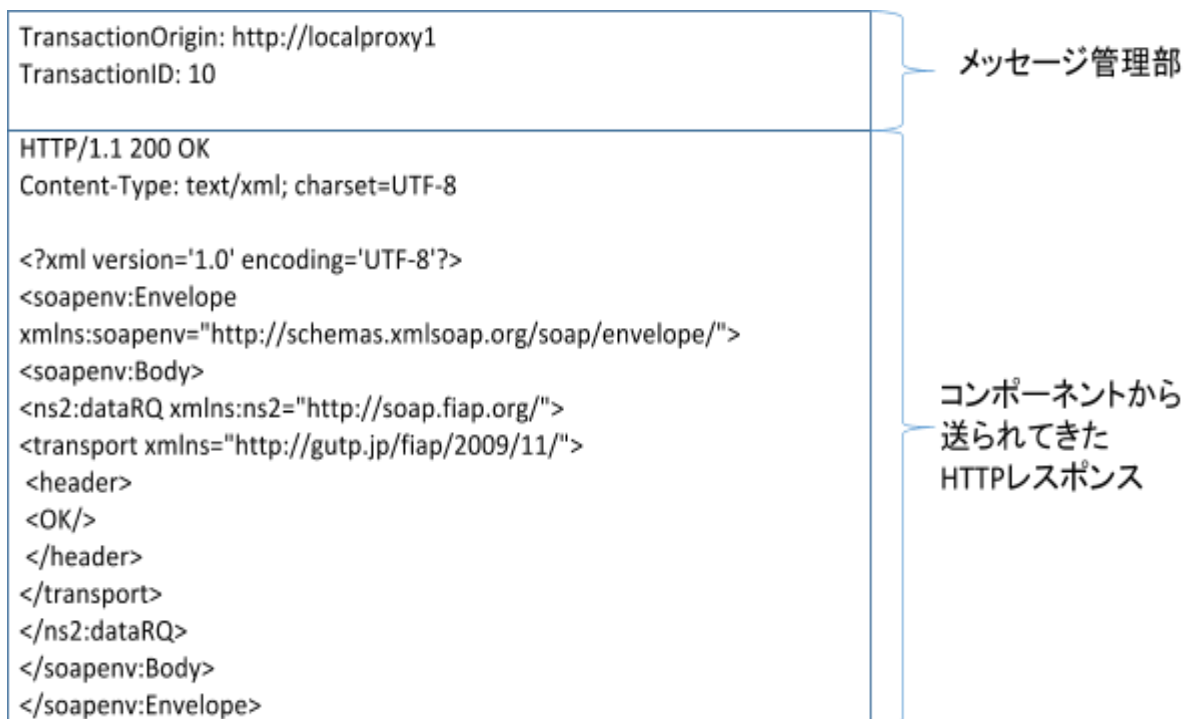


図 14 WebSocket コネクション上を流れるメッセージ例(リプライの転送)

WebSocket コネクション上を通じて転送されたメッセージは、メッセージ管理部を外して、宛先のコンポーネントに送られる。これらの一連の動作において、一部の例外を除いて任意の HTTP メッセージは書き換えられることなく送受信される。メッセージ転送の際に発生するエラーは Global Proxy のプロキシ・サーバ・インターフェース節および Local Proxy のプロキシ・サーバ・インターフェース節のとおりである。

## 7. セキュリティに関する留意事項

この章では、本仕様書で定める技術を用いた場合のセキュリティに関する留意事項について、図 15を参照しながら述べる。

図 15は、ローカルネットワークとグローバルネットワークにあるコンポーネント間の通信を本仕様書で定める技術を用いて実現した場合の典型的なシステム構成である。ローカルネットワークの IEEE1888コンポーネント A がクライアントまたはサーバとなり Local Proxy と通信を行う部分を X とする。グローバルネットワークの IEEE1888コンポーネント B がクライアントまたはサーバとなり Global Proxy と通信を行う部分を Y とする。さらに、Local Proxy と Global Proxy が通信を行う部分を Z とする。

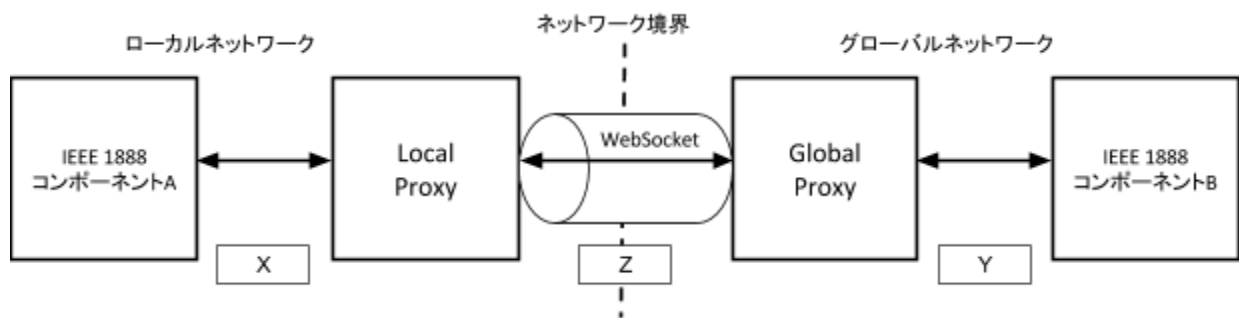


図 15 IEEE1888 over WebSocket を用いた典型的なシステム構成

## 7.1. ローカルネットワークのセキュリティ

本仕様書で定める技術を利用すると、グローバルネットワークの任意の IEEE1888 コンポーネントが、FW 等で保護されたローカルネットワークのコンポーネント A に自由にアクセスできるようになる。また、Global Proxy のプロキシ・サーバ・インターフェースが公開する URL にグローバルネットワークにある通常の HTTP クライアントがアクセスすることにより、ローカルネットワークに対して何らかの悪影響を与える可能性がある。さらに、図 15 には示されていないが、別のローカルネットワークが Global Proxy を経由して、図中のローカルネットワークに接続された場合にも、同様のことが起きる。

したがって、本仕様書で定める技術を導入する場合は、関係しているネットワーク、及びシステムの管理者は事前に十分に協議を行いリスクを共有すると共に、適切な変換表を Proxy に設定しなければならない。また、ローカルネットワークを保護するため、Proxy は最低限アクセスコントロール機能を提供しなければならない。ただし、通信相手を認証しないアクセスコントロールは限定された効果しかないことを考慮すべきである。

## 7.2. コンポーネント間の通信のセキュリティ

本仕様書で定める技術では、IEEE1888.3 を用いたコンポーネント間の安全な通信は実現できない。これは、図 15 におけるコンポーネント A とコンポーネント B 間の通信を Local Proxy と Global Proxy が中継していて、各区間で別々の TLS セッションを構築するからである。ただし、下記に述べる方法である程度の安全な通信を実現することができる。

### 7.2.1. 限定的なコンポーネント間の通信のセキュリティ

限定的にコンポーネント間のセキュリティを実現するには、2つの方法が考えられる。

1つは、図 16 の様にコンポーネントと Proxy を1つのノードに実装して、かつ、Proxy 間の通信に相互認証可能な WebSocket Security (WSS) を使った場合である。本仕様書では、WSS における通信相手の認証方法や相互認証方法については述べないが、この方法を採用する場合は必ず検討しなければならない事項である。

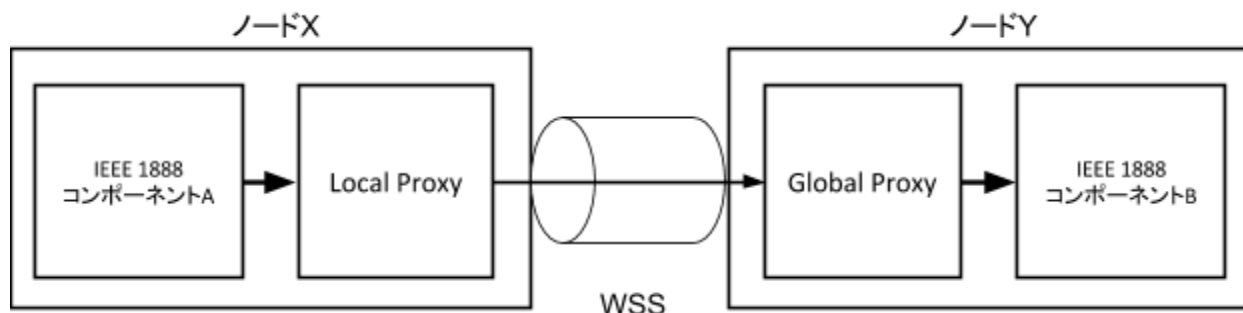


図 16 コンポーネントと Proxy を同じノードに実装したセキュアな通信の例

もう1つの方法は、図 15における A,Z,Y のそれぞれの通信に TLS を用いて、かつ Local Proxy, Global Proxy を完全に信頼できる場合である。この場合、Proxy が物理的にも安全に運用されている必要がある。図 17に、コンポーネント A がクライアントとなる場合の例を示す。まずコンポーネント A に対するコンポーネント B の URL PE-B、及びコンポーネント B の実際の URL E-B のスキーマは https とする。コンポーネント A は、IEEE1888.3を用いて Local Proxy の PE-B に接続する。Local Proxy は変換表に従い PE-B を E-B に変換して、Global Proxy との WSS コネクションにその通信を転送する。最後に Global Proxy はコンポーネント B の E-B と IEEE1888.3で通信を行う。

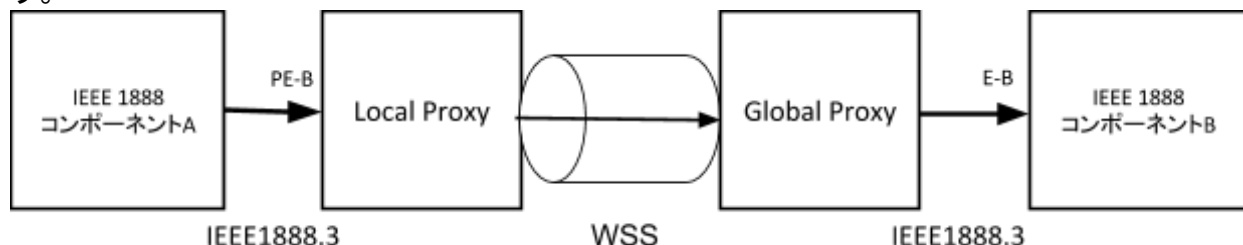


図 17 信頼できる Proxy を用いたセキュアな通信の例

### 7.3. 運用ポリシーの異なるネットワーク間の運用

ローカルネットワークとグローバルネットワークのセキュリティに関する運用ポリシーが異なる場合でも、IEEE1888の通信を利用したいという要求がある。例えば、グローバルネットワークでは IEEE1888.3を採用しているが、ローカルネットワークの一部の機器の計算リソースが少ないために全ての機器に IEEE1888.3を採用できない時に、グローバルネットワークからローカルネットワークの IEEE1888.3を採用していない機器の情報を取得したり操作したい場合である。この場合は、Proxy 間に通常の WebSocket と WSS の二つのセッションを構築して、Global Proxy が提供する URL のスキーマに応じてセキュアなパスとそうではないパスの2つを提供することが考えられる。(図 18)

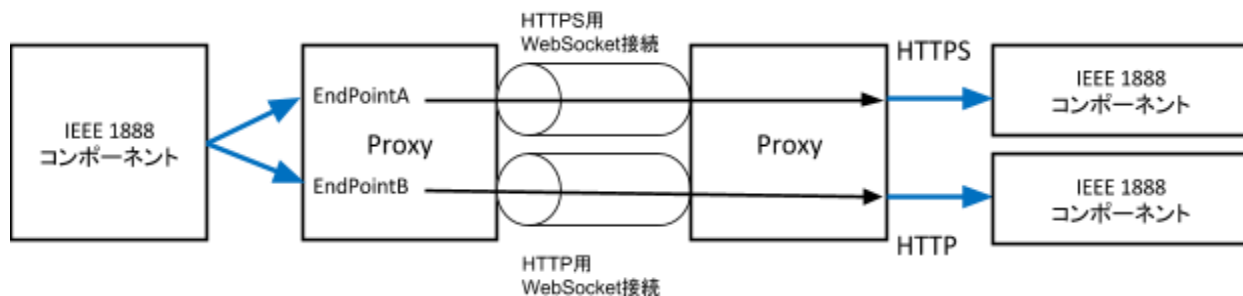


図 18 プロトコル別 WebSocket エンドポイントの利用

#### 7.4. 変換表の整合性

例えば、グローバルネットワークもローカルネットワークも IEEE1888.3を採用していたとする。ここで、プロキシ・サーバ・インターフェースが提供する URL のスキーマは HTTPS にもかかわらず、変換後のスキーマとして HTTP を誤って設定してしまうと、クライアントは安全な通信を期待していたにも関わらず、Proxy からサーバまでは TLS が使われなくなってしまう。一方で、各通信区間（図 15における X,Y,Z）において、HTTP、HTTPS を選択的に使用したいという運用ポリシーがあるかもしれない。そのような場合にも適切なスキーマの変換が必要になる。したがって、Proxy のプロキシ・サーバ・インターフェースの変換表の設定には特に注意すべきである。