QlikView のセキュリティについて

クリックテック・ジャパン株式会社
グローバルサービス部
2011年7月

QlikView

本セッションの目的

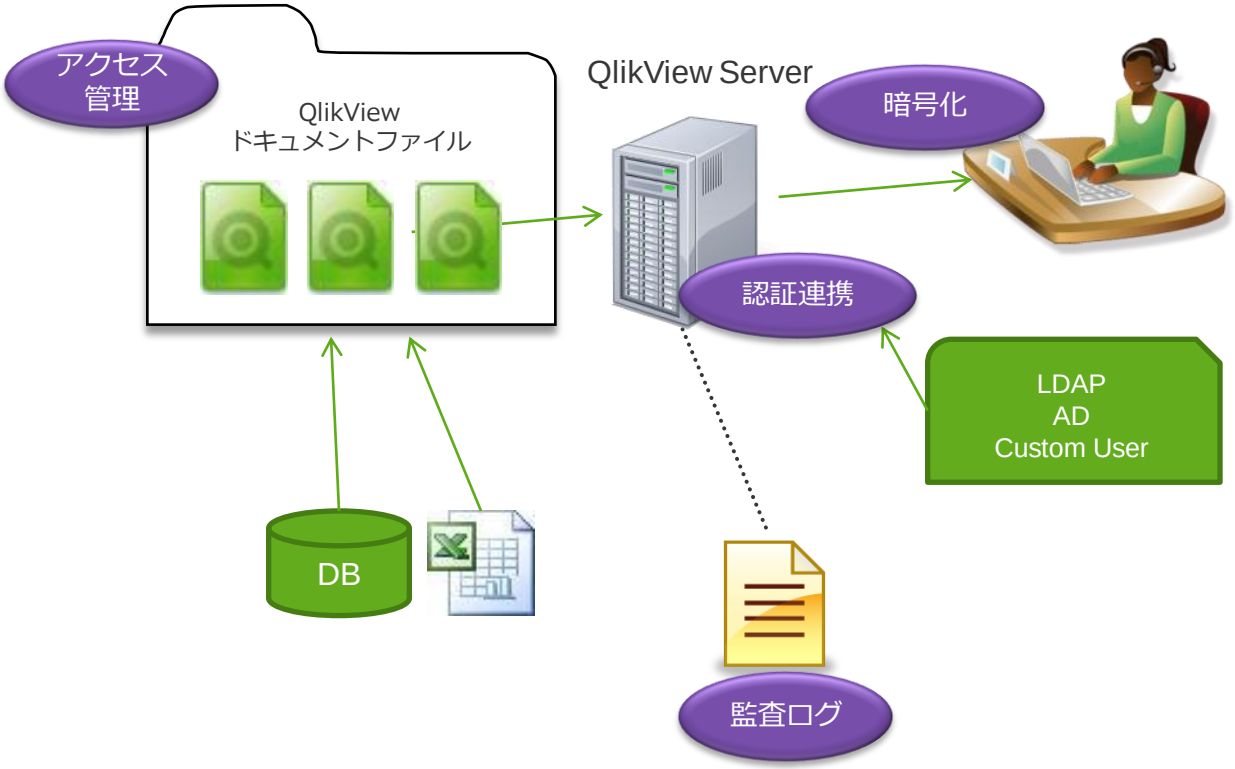
本セッションの目的は以下の通りです

- QlikView のセキュリティについて理解（おさらい）する
- QlikView へのアクセス時にどのように認証が行われるかを理解する
- QlikView ドキュメントに対するアクセス許可がどのように行われるかを理解する
- QlikView ドキュメントのデータに対するアクセス許可がどのように行われるかを理解する

QlikView のセキュリティ管理

情報活用基盤を支えるセキュリティ機能

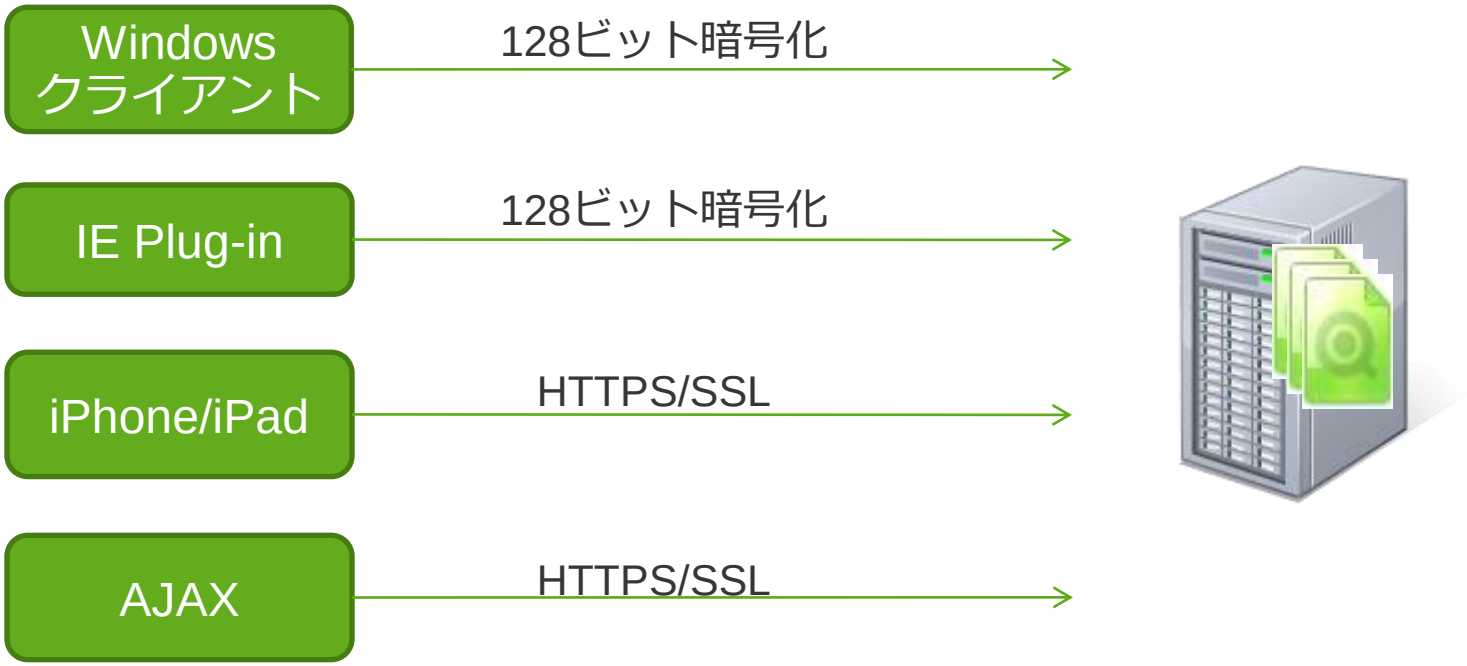
- LDAP, カスタム認証など
- フォルダ単位、グループ単位での容易なアクセス権管理
- レポートレベル、行レベル、列レベルでの詳細なアクセス権制御
- 利用情報の監査、監査用サンプルレポート



通信の暗号化

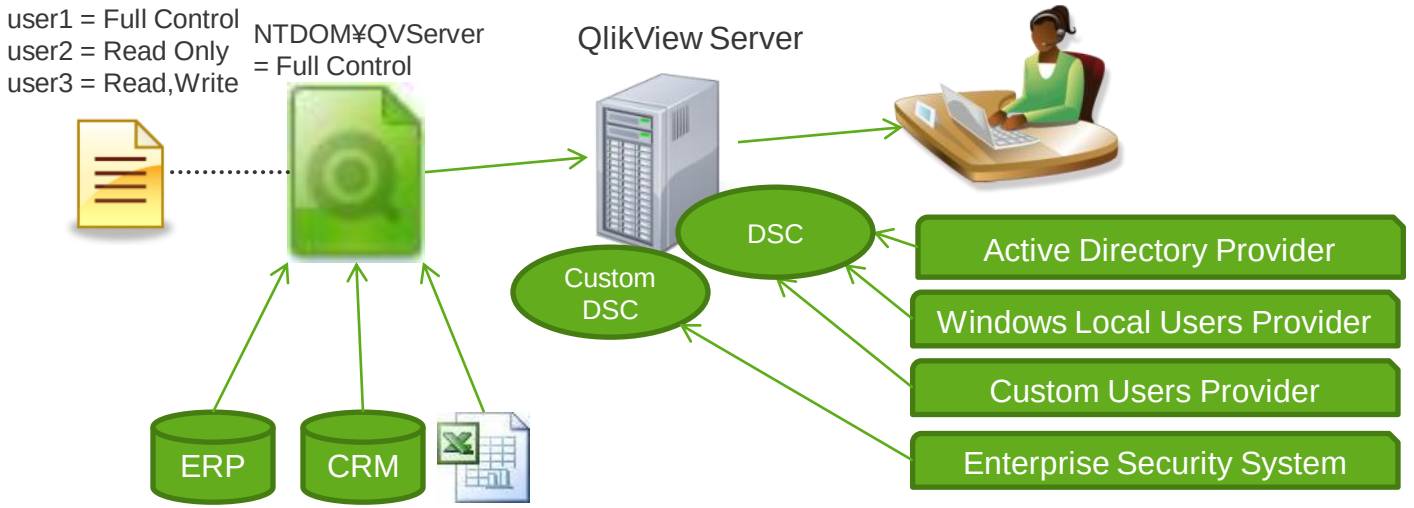
暗号化

- QlikView Server と Windows ベースのクライアントの間の通信はすべて暗号化
- QlikView Server はクライアントが接続するとき、RSA アルゴリズムに基づいた 128 ビットの暗号を確立
- Ajaxクライアントの場合は、SSLを利用可能



認証連携

- 認証（= Authentication）:”Who is this user?”
  - このユーザーは誰か?
- ユーザー認証方法
  - OSへのログオン (シングルサインオン)
  - ディレクトリ サービスを使用したログオン



アクセス管理

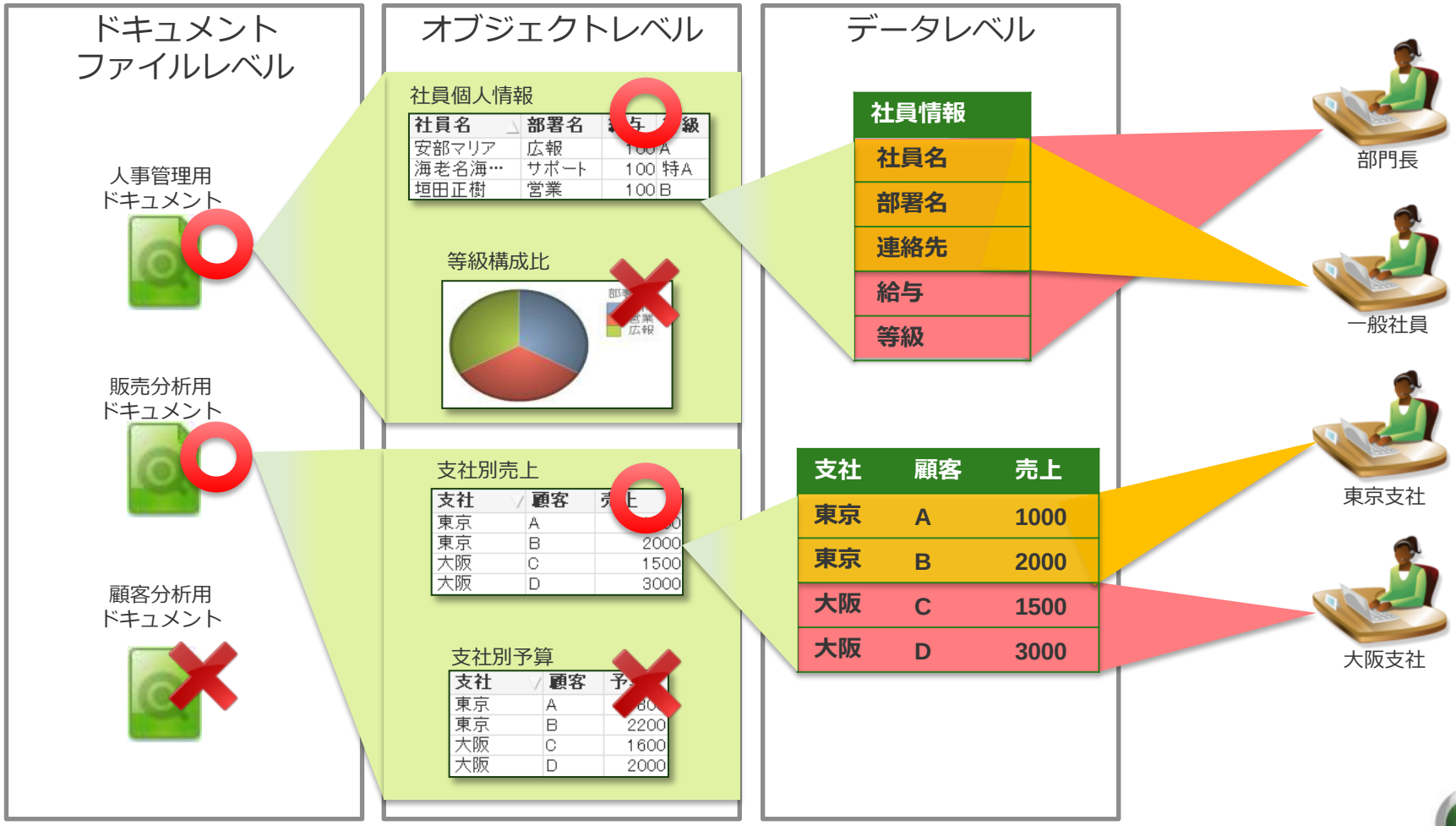
QlikViewドキュメントファイルに対するアクセス権の制御

- WindowsのNTFSを利用したアクセス管理（NTFSモード）
  - 許可は Windows NTFS ファイル システムによって処理される。
  - Windows を通して認証が行われる必要がある
- QlikView 独自のアクセス管理（DMSモード）
  - QlikView Server 上の DMS(Document Metadata Service) 設定によりアクセス許可

データへのセキュリティ制御

アクセス管理

- QlikViewでは様々なレベルでユーザー/グループごとにきめ細やかなセキュリティ制御をかけることが出来ます

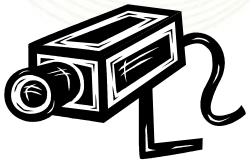


※NTFSアクセス権で設定

※オブジェクト毎のプロパティで設定

※SectionAccess機能により設定

監査ログ



- ユーザーの操作をログファイルに記録
- 誰が、いつ、どのデータを参照し、どんな操作を行ったかをトラッキング

ログに記録される操作:

- ✓ シートの選択
- ✓ 値の選択
- ✓ ブックマークの使用
- ✓ レポートの使用
- ✓ オブジェクトのクリア
- ✓ すべてクリア

Audit Log Details

DocName	User	SessionStart	Timestamp	Action	Object	Value
QV10_DEMO.qvw	QTSEL\how	2010-12-06 11:56:02	2010-12-06 11:56:42	Selection	営業員名	哀川 崇史, 安斎 知世
QV10_DEMO.qvw	QTSEL\how	2010-12-06 11:56:02	2010-12-06 11:56:34	Selection	顧客名	会田電化
QV10_DEMO.qvw	QTSEL\how	2010-12-06 11:56:02	2010-12-06 11:56:30	Selection	都道府県	埼玉県
QV10_DEMO.qvw	QTSEL\how	2010-12-06 11:56:02	2010-12-06 11:56:23	Selection	支店名	関東支店
QV10_DEMO.qvw	QTSEL\how	2010-12-06 11:56:02	2010-12-06 11:56:19	Activated sheet D...	SH01	-
QV10_DEMO.qvw	QTSEL\how	2010-12-06 11:56:02	2010-12-06 11:56:13	Selection	年度	2007 年度
QV10_DEMO.qvw	QTSEL\how	2010-12-06 11:56:02	2010-12-06 11:56:11	Selection	製品中分類名	ブルーレイレコーダー
QV10_DEMO.qvw	QTSEL\how	2010-12-06 11:56:02	2010-12-06 11:56:09	Selection	製品大分類名	レコーダー
Data Visualizatio...	JPTOK-HOW\how	2010-12-06 11:04:36	2010-12-06 11:05:26	Selection	Region	JAPAN
Data Visualizatio...	JPTOK-HOW\how	2010-12-06 11:04:36	2010-12-06 11:05:20	Activated sheet D...	SH15	-
Data Visualizatio...	JPTOK-HOW\how	2010-12-06 11:04:36	2010-12-06 11:05:17	Activated sheet D...	SH12	-
Data Visualizatio...	JPTOK-HOW\how	2010-12-06 11:04:36	2010-12-06 11:05:16	Activated sheet D...	SH11	-

セキュリティ関連のキーワード

- ドメイン／Active Directory

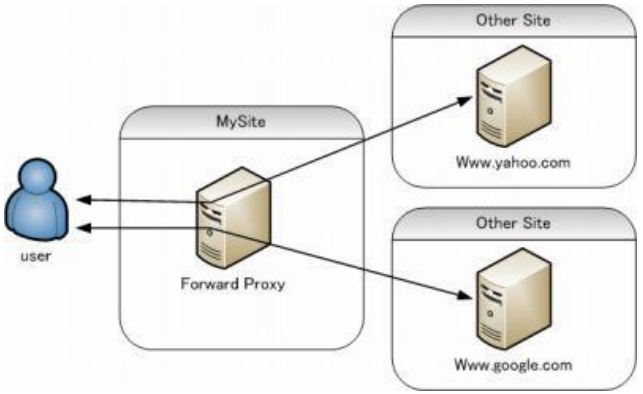
- SSO
  - IDとパスワードの入力は一度だけで済むという考え方が一般的
  - 毎回入力はするものの、同じIDとパスワードを利用可能という意味に使う場合もある

- Firewall
  - 外部との境界を流れるデータを監視し、不正なアクセスを検出・遮断

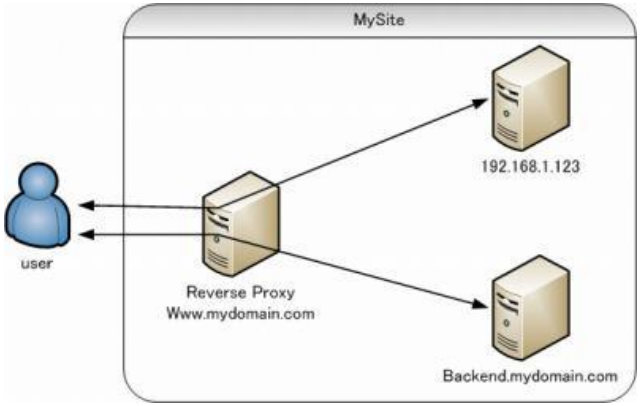
- DMZ
  - インターネットに接続されたネットワークにおいて、ファイアウォールによって外部ネットワーク(インターネット)からも内部ネットワーク(組織内のネットワーク)からも隔離された区域

セキュリティ関連のキーワード

- フォワードプロキシ
  - 内部ネットワークの「代理」としてインターネットとの接続を行うサーバー。
  - 内部から特定の種類の接続のみを許可したり、外部からの不正なアクセスを遮断



- リバースプロキシ
  - 特定のサーバの代理として、そのサーバへの要求を中継。
  - ユーザは全てリバースプロキシを経由することになるため、サーバが直接アクセスを受けることはなくなる。
  - また、コンテンツをキャッシュに保存することで高速化したり、パスワード認証によってアクセス制限をするなどの機能を持つ。
  - 社内LANなどの内部ネットワークとインターネットとの接点に置かれ、外部からのアクセスを中継することもあるが、その様子が通常のプロキシ(フォワードプロキシ)の「内部から外部へのアクセスを中継する」動作と反対であることが「リバース」の由来であると言われている。



「IT用語辞典」より抜粋

セキュリティ関連のキーワード

- 暗号化
  - データそのもの
  - ユーザー ⇔ サーバー間
  - 各サービス間

- 認証(Authentication) vs 許可(Authorisation)
  - 認証 = 誰であるかを確認すること
    - ✓「正当性を検証する作業。例えば、ユーザ名とパスワードの組み合わせを使って、コンピュータを利用しようとしている人にその権利があるかどうかや、その人が名乗っている本人かどうかなどを確認すること。」（「IT用語辞典」より）
  - 許可 = アクセス許可を与えること
    - ✓「認証（Authentication）によって確認された利用者を識別して、アクセス権限の制御を行い、利用者ごとに固有のサービスを提供すること。」（@IT 「セキュリティ用語辞典」より）

QlikViewのセキュリティアプローチ

- 基本的な３つのアプローチ

  - ユーザーが適切にサーバーアクセスできるようなインフラ構成
    - ✓ 3rd Party SSO製品や、各種認証システム、VPN等、QlikViewの外側で構築
  - QlikViewに対し認証されたユーザー名を渡す
    - ✓ QlikView Server自体が直接ユーザー認証を行うわけではないことに注意
  - アクセス許可をコントロール
    - ✓ あるユーザーがどのドキュメントを参照する権限を持つかを判断
    - ✓ ドキュメントを開くと、今度はそのユーザーがどのデータを参照することが出来るか判断

QlikView における「認証」について

QlikViewにおいてユーザーを識別する方法としては、以下の4つがあげられます

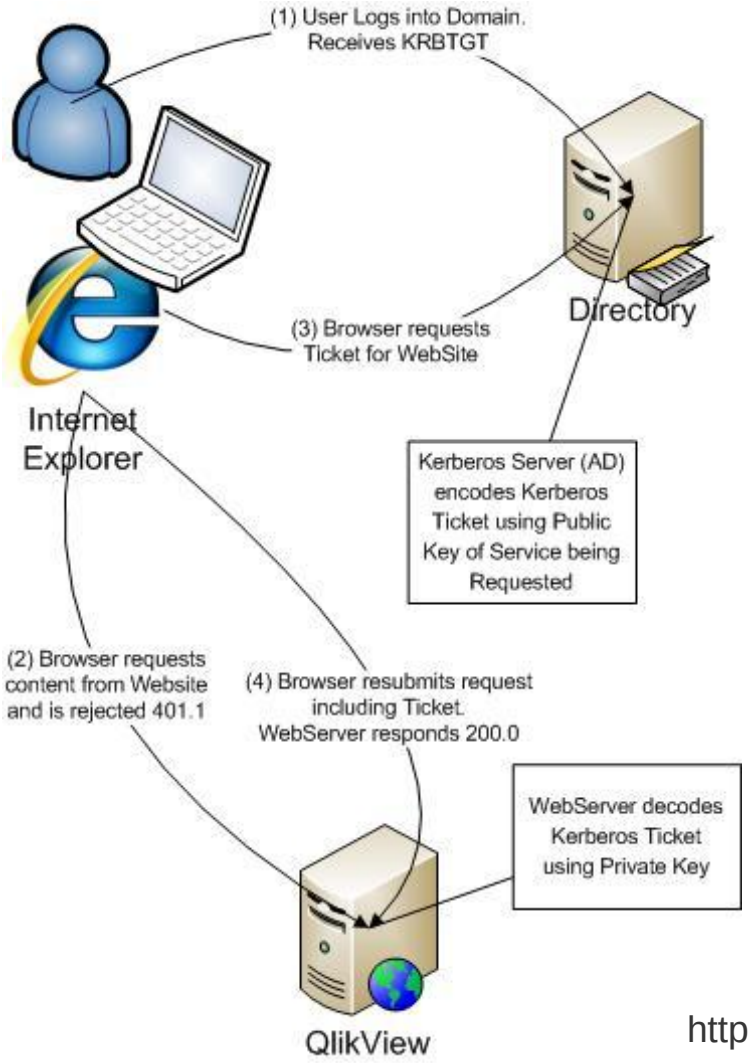
1. QlikViewデフォルト
   - QlikView Desktop： Windows認証
   - QlikView WebServer： NTLM認証によるシングルサインオン
   - IIS： 統合Windows認証 （Kerberos/NTLM）
2. HTTP Header （要設定）
   - AccessPointを経由する
   - 認証はリバースプロキシ等で事前に行われる
   - または、ISAPIフィルタが暗号化されたセッションクッキー等から判断
3. Ticket認証 （要開発）
   - 単一のドキュメントにのみ対応し、ドキュメント連携には非対応
   - AccessPointの機能は提供しない
   - 他のWebアプリケーションから直接ドキュメントを起動するとき等に適した方法
4. Custom Directory (フォーム認証)
   - QlikView側で認証するための唯一の方法

認証① QlikViewデフォルト

ユーザー識別方法

- PCへログイン
- ユーザー識別情報をActive Directory等の認証ディレクトリに問い合わせて認証
- AccessPointを参照する場合は、WebServerがブラウザに対し、識別情報を渡すよう要求
  - 統合Windows認証
    - ✓ Kerberos
    - ✓ NTLM
  - HTTP ベーシック認証（ユーザーID/パスワード入力用ポップアップ）
- AccessPointは、WebServerが差し出す識別情報を信頼

統合Windows認証



1. ユーザーはクライアントPCからドメインにログイン。ADからTGT(Ticket Granting Ticket)を取得
2. ブラウザからQlikViewにアクセスすると、WebServerによりアクセス拒否される（HTTP 401.1）
3. ブラウザはADに対しサービス プリンシパル名 (SPN：[例 HTTP/website.qliktech.com]) を取得するためにKerberosチケットを要求
4. ブラウザは、WebServerに対し、ユーザー名を含むKerberosチケットを持って再度アクセス
5. WebServerはチケットをデコードしてIDを判別。ブラウザに成功応答を返す（HTTP 200）

http://keicode.com/windows/kerberos-basic.php より

参考：統合Windows認証

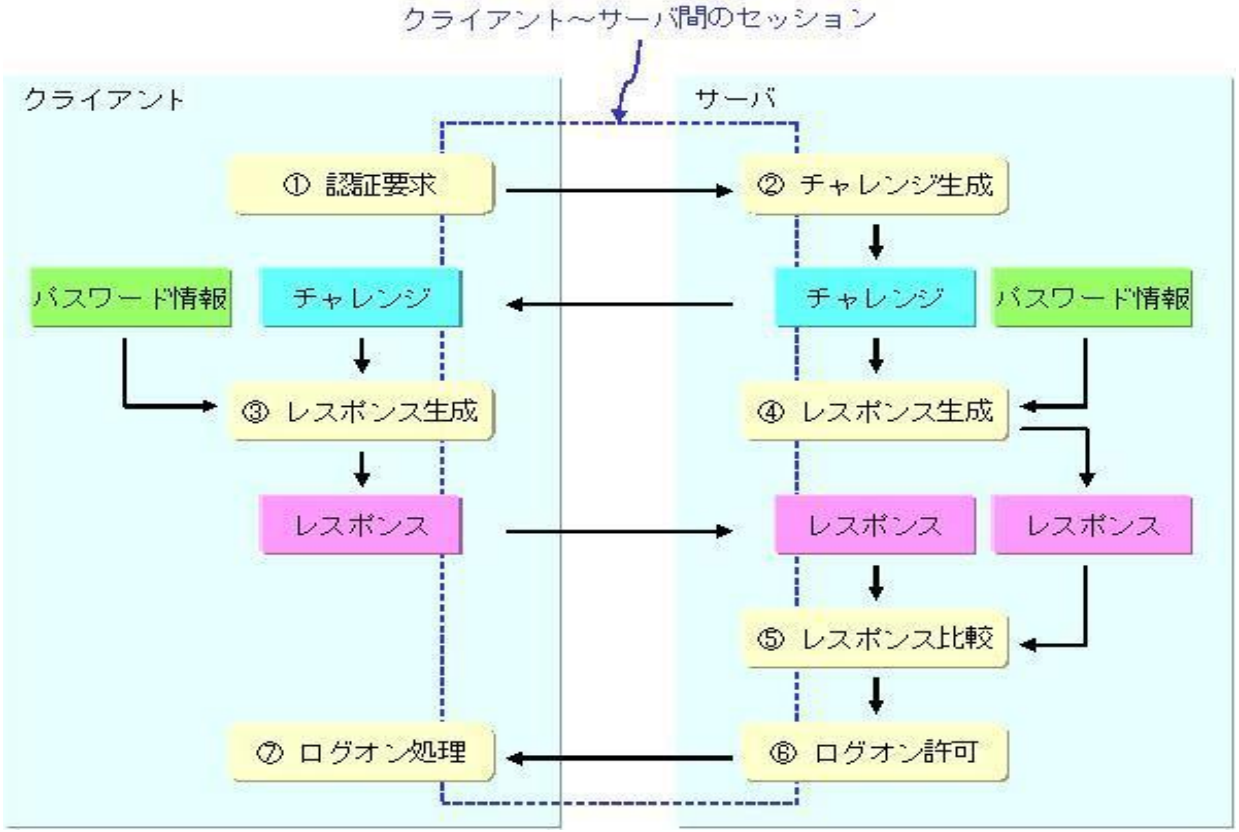
クライアント コンピュータ上にある最新の Windows ユーザー情報が使用されるため、基本認証と異なり、ユーザー名とパスワードの入力を要求せず、ユーザー名とパスワードがネットワークで転送されない安全な認証形式です。

- 統合 Windows 認証を有効にした場合
  - ユーザーのブラウザは、ハッシングを使用して Web サーバーと暗号化された情報を交換し、パスワードを知っていることを証明します。
  - 統合 Windows 認証では、Kerberos Version 5 認証プロトコルと、独自のチャレンジ/レスポンス認証プロトコルの両方を使用できます。サーバーにディレクトリ サービスがインストールされていて、ブラウザに Kerberos v5 認証プロトコルとの互換性がある場合、Kerberos v5 認証プロトコルとチャレンジ/レスポンス プロトコルの両方が使用されます。
    その他の場合、チャレンジ/レスポンス プロトコルのみが使用されます。
- 統合 Windows 認証の流れ
  - クライアント コンピュータ上にある最新の Windows ユーザー情報が使用されるため、ユーザー名とパスワードの入力を要求しない。
  - 最初に認証処理がユーザーの識別に失敗した場合、ブラウザはユーザーに対して Windows ユーザー アカウントのユーザー名とパスワードの入力を要求します。入力された情報は、統合 Windows 認証によって処理されます。 Internet Explorer は、ユーザーが有効なユーザー名とパスワードを入力するか、またはダイアログ ボックスを閉じるまで、ユーザーに繰り返し入力を要求します。

※以前は、“NTLM”、“Windows NT チャレンジ/レスポンス認証” と呼ばれていた。

http://blog.chsing.net/castle/archive/2004/07/21/244.aspxより

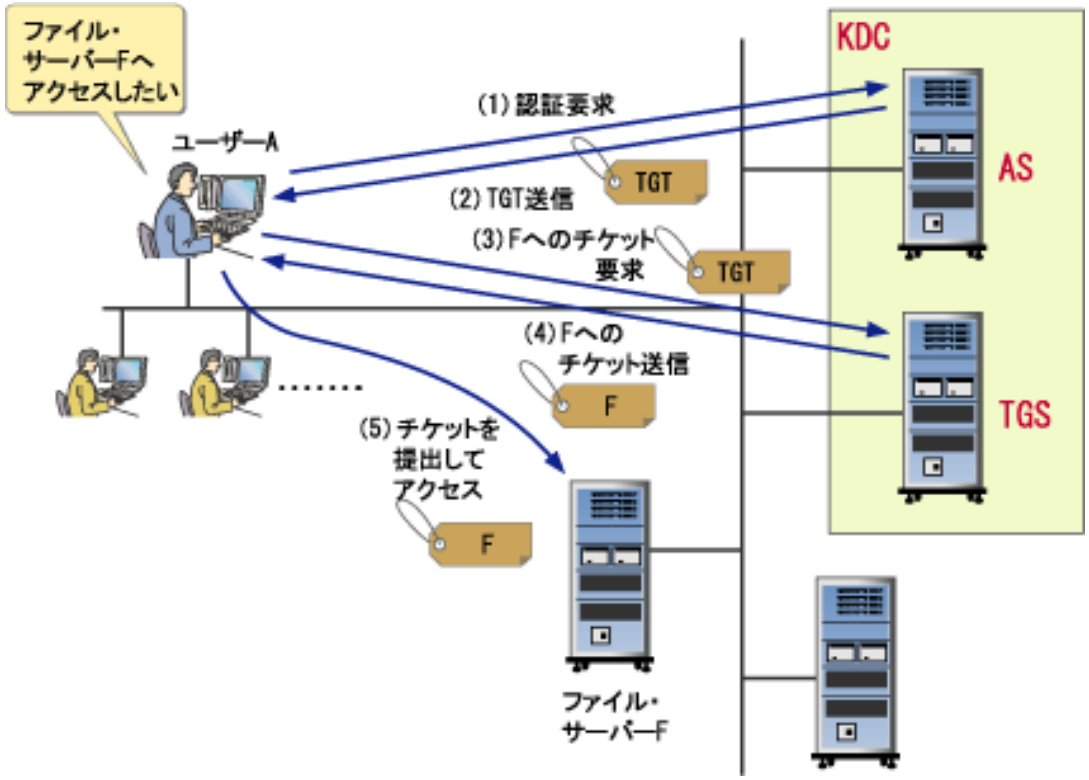
参考：NTLM認証の流れ



① クライアントがサーバに対し、ユーザ認証の要求を発行する
② サーバは認証要求を受け、ランダムなバイト列「チャレンジ」を送り返す
③ クライアントは、チャレンジとパスワード情報に基づいて「レスポンス」を生成し、サーバに送る
④ サーバ側でも先ほど送ったチャレンジとパスワード情報を基にレスポンスを生成する
⑤ クライアントから送られたレスポンスと、自ら生成したレスポンスを比較することにより、クライアント側とサーバ側両方のパスワード情報が同一であることを確認する
⑥ パスワード情報の同一性が確認できた場合、クライアントにログオン許可を与える
⑦ クライアント側ではログオン許可を受け、ログオン処理を実行する

http://www.st.rim.or.jp/~shio/csm/ntlm/より

参考：Kerberos認証の流れ



① ユーザーAは，KDC （Key Distribution Center）のAS（Authentication Server）に認証してもらう
② 正しいIDとパスワードで認証に成功すると，ユーザーに対してTGT（Ticket Granting Ticket：チケット発行のための大もとのチケット）が発行される
③ 次に，ユーザーはKDCのTGS（Ticket Granting Server）に対して，実際にアクセスしたいサーバーへの利用権を請求する
④ TGSはユーザーに対して，チケットと呼ばれるものを発行
⑤ ユーザーはこのチケットをアクセス先のサーバーへ提出して，これを受け取ったサーバーはユーザーを認識し，そのユーザーに合ったアクセスを 許可

http://itpro.nikkeibp.co.jp/article/COLUMN/20060518/238303/より

参考：HTTP認証方式について

Microsoftサイトより引用：
http://msdn.microsoft.com/ja-jp/library/ms789031.aspx

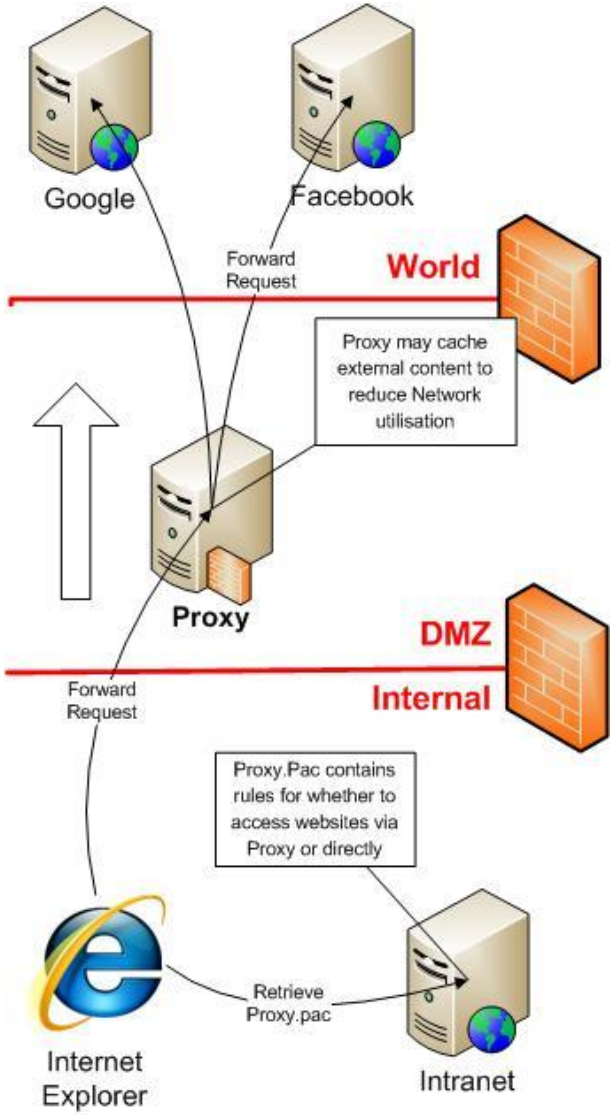
認証方式	説明
匿名	匿名要求は、認証情報を含みません。これは、リソースへのアクセス権をすべてのユーザーに付与することを意味します。
基本	基本認証では、クライアントのユーザー名とパスワードを含む Base64 エンコード文字列が送信されます。Base64 は暗号化の形式ではありません。クリア テキストでのユーザー名とパスワードの送信と同じであると考えてください。リソースを保護する必要がある場合は、基本認証以外の認証方式の使用を検討して ください。
ダイジェスト	ダイジェスト認証は、基本認証の代わりに使用できるチャレンジ レスポンス方式の認証です。サーバーは、nonce と呼ばれるランダムな文字列データをチャレンジとしてクライアントに送信します。クライアントは、ユーザー名、パスワード、nonce およびその他の追加情報を含むハッシュを使用して応答します。この認証方式では、このようなデータの交換によってもたらされる複雑さとデータのハッシュに より、ユーザーの資格情報を盗んで再使用することがより困難になります。 ダイジェスト認証では、Windows ドメイン アカウントを使用する必要があります。ダイジェストのレルムは Windows ドメイン名です。したがって、Windows ドメインをサポートしていないオペレーティング システム (Windows XP Home Edition など) で実行されているサーバーでは、ダイジェスト認証を使用できません。逆に、Windows ドメインをサポートしていないオペレーティング システムでクライアントが実行されている場合は、認証時にドメイン アカウントを明示的に指定する必要があります。
NTLM	NTLM (NT LAN Manager) 認証もチャレンジ レスポンス方式の認証ですが、ダイジェスト認証よりもセキュリティが強化されています。NTLM 認証では、エンコードされていないユーザー名とパスワードではなく、Windows 資格情報を使用してチャレンジ データが変換されます。NTLM 認証では、クライアントとサーバー間で複数のメッセージ交換を行う必要があります。認証を正常に完了するため、サーバーおよび介在するすべてのプロキシが 永続的な接続をサポートしている必要があります。
Negotiate	ネゴシエート認証では、可用性に応じて、Kerberos プロトコルと NTLM 認証のいずれかが自動的に選択されます。Kerberos プロトコルを使用できる場合は Kerberos プロトコルが使用され、それ以外の場合は NTLM が使用されます。Kerberos 認証は、NTLM 認証を大幅に強化した認証方式です。Kerberos 認証は NTLM 認証よりも高速であるだけでなく、相互認証およびリモート コンピューターへの資格情報の委任を使用できます。

認証② HTTP Header

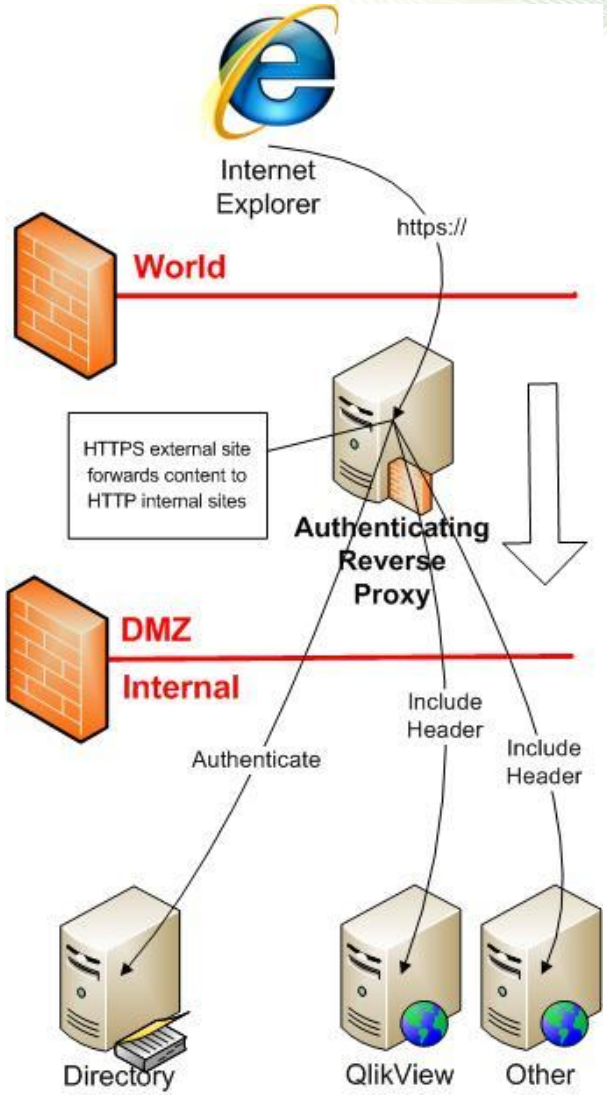
ユーザー識別方法

- 認証用リバースプロキシ経由でQlikViewへアクセス
  - プロキシまたはリバースプロキシとは * (次スライド参照)
  - プロキシが識別したユーザーをブラウザのリクエストのヘッダーに埋め込む
- ISAPI フィルタは、クッキーを元にIDを判別
  - ユーザーが外部アプリケーション/ポータルにログイン
  - アプリケーション/ポータルは、ID情報またはセッション情報を含むクッキーをセット（暗号化）
  - ISAPIフィルタは、QlikViewに送られたリクエストを傍受し、クッキーからID情報を識別。ブラウザからのリクエストにヘッダーを追加
- AccessPointは、WebServerから渡されたIDを常に信頼

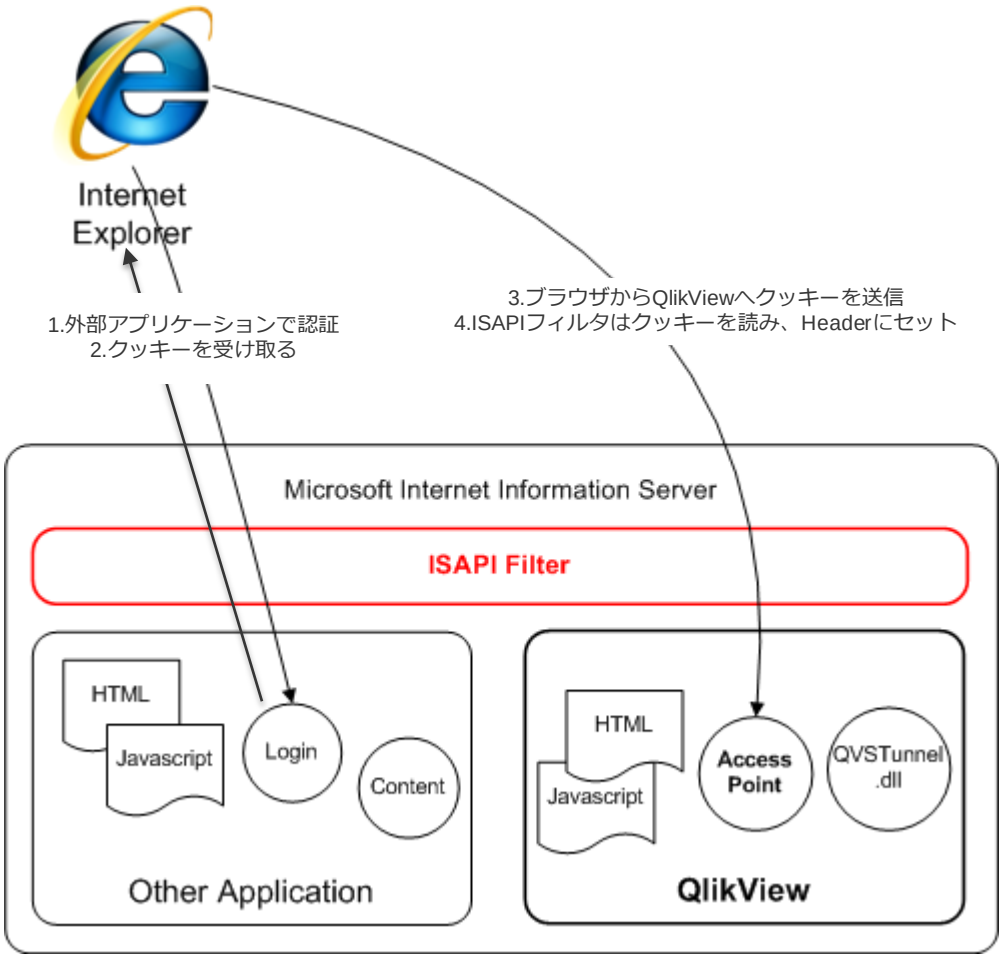
プロキシサーバー



- リバースプロキシは、外部から社内のWebServerへアクセス可能にするために有効
- リバースプロキシはSSLを利用可能だが、非SSLのWebServerへ接続を転送することも可能。これらの接続は”ジャンクション”と呼ばれる
- 認証はLDAPやその他のDBで行われ、HTTPヘッダーをセットすることでQlikViewが識別情報を読み込むことが可能となる
- AccessPoint は、ユーザー名を含むヘッダーの名称を認識し、ユーザーを判別



ISAPI フィルタ



1. まず、ユーザーはポータルなどのQlikView外のアプリケーションで認証
2. 認証により、ブラウザにクッキーが送信される
3. ブラウザはクッキーを持ってQlikViewにアクセス
4. ISAPI フィルタはクッキーからIDを識別、HTTP Headerにセット
5. AccessPoint は、ユーザー名を含むヘッダーの名称を認識し、ユーザーを判別

認証③ Ticket認証

ユーザー識別方法

AccessPointからドキュメントを開くときは常に、ユーザー識別情報を表すチケット情報がURLに付加される

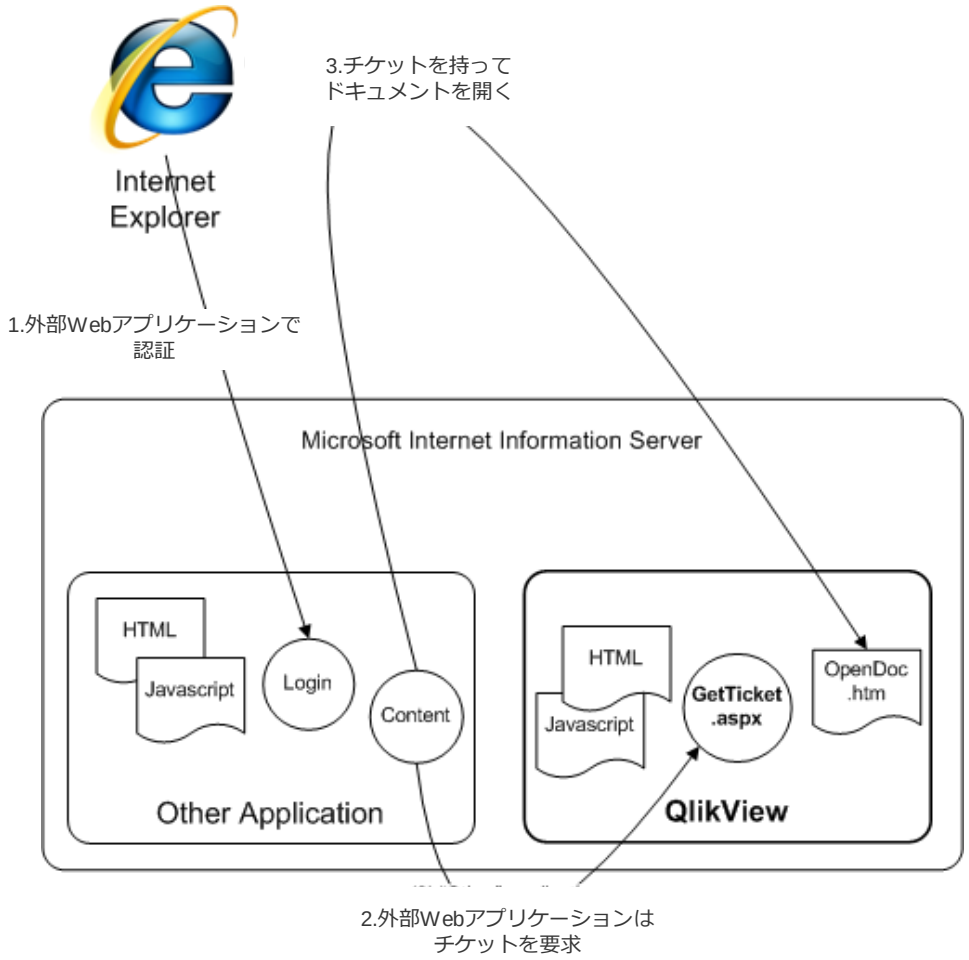
以下の方法でプログラム的にチケットをリクエスト可能

- 特定のQvS
  - QvsNetRemote.dll
  - QvsComRemote.dll
- WebServerに対してリクエストをPOST
  - v10: http://localhost/QvAJAXZfc/GetTicket.aspx?admin=
  - v9: http://localhost/QvAJAXZfc/QvsClient.asp?admin=

最終的にチケット情報を含むURLが生成

- http://localhost/**QvAJAXZfc**/opendoc.htm ?document=Sales/NL.qvw &**ticket=ABCXYZABCXYZ**

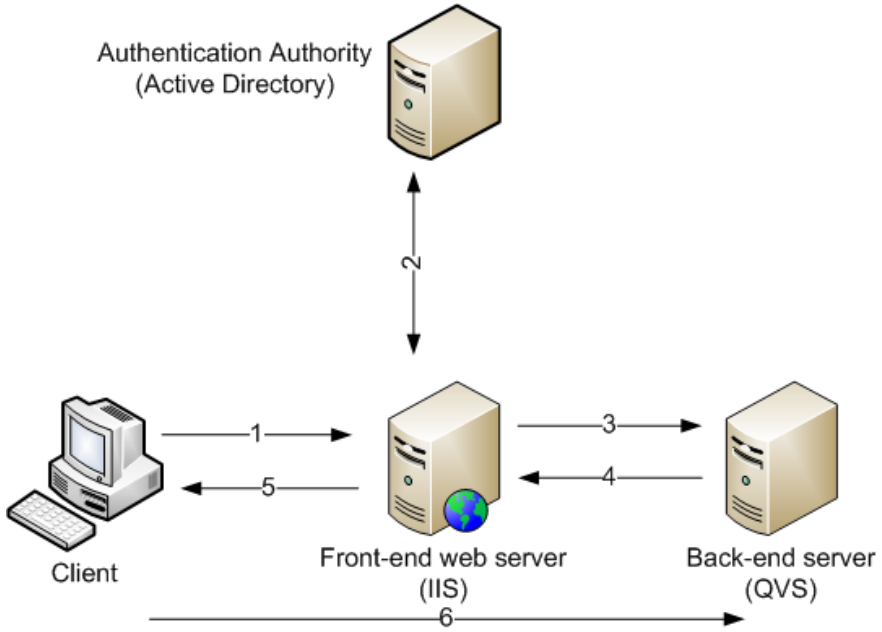
認証③ Ticket認証（基本的な流れ）



1. まず、ユーザーはポータルなどのQlikView外のアプリケーションで認証
2. QlikView以外のアプリケーションは、“GetTicket.aspx” またはQvsNetRemote.dll APIを使ってQVSに対しチケットをリクエスト
3. ユーザーは、“OpenDoc.htm” にリダイレクトされ、開くドキュメントおよびチケットが特定される

認証③ Ticket認証（パターン1）

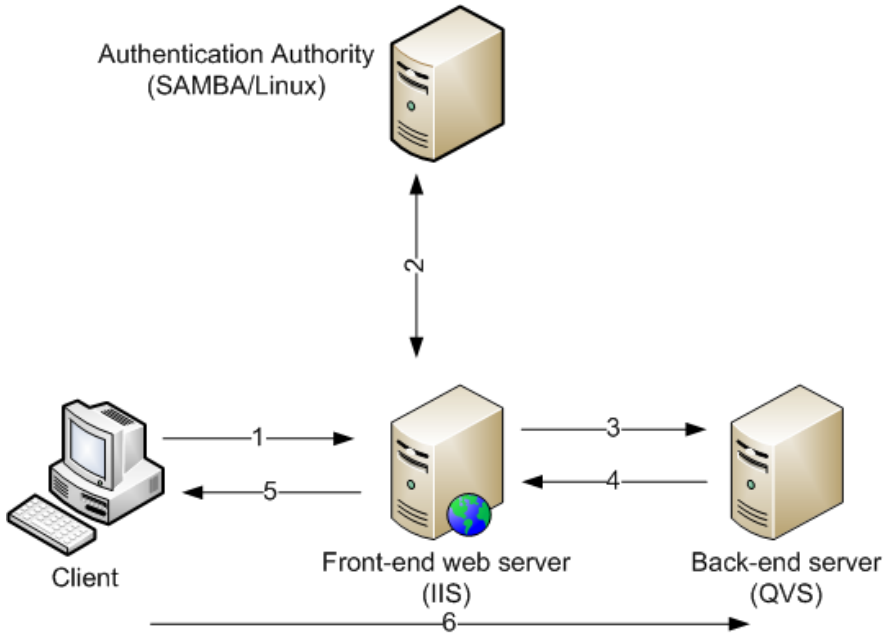
GetTicketForMe
（Windowsユーザーの場合）



1. クライアントがWebサーバーへのアクセスをリクエスト
2. クライアントがActive Directoryで認証される
3. Webサーバーが認証されたWindowsユーザーに基づくチケットの発行を依頼
4. QlikView Server がWebサーバーに対しチケットを発行
5. Webサーバーがチケットをクライアントに転送
6. ドキュメントを開くために、クライアントがQlikView Serverにチケットを送信
7. クライアント と QlikView Server の間で認証が確立される

認証③ Ticket認証（パターン2）

GetTicket(user)
（非Windowsユーザーの場合）



1. クライアントがWebサーバーへのアクセスをリクエスト
2. クライアントが非Windows認証サービスにより認証される
3. Webサーバーが認証された非Windowsユーザーに基づくチケットの発行を依頼
(リクエストをするユーザーは、QlikView Server 上のLocal QlikView Administrators グループのアカウントに成りすます)
4. QlikView Server がWebサーバーに対しチケットを発行
5. Webサーバーがチケットをクライアントに転送
6. ドキュメントを開くために、クライアントがQlikView Serverにチケットを送信
7. クライアント と QlikView Server の間で認証が確立される

注: QlikView Serverは、DMSモードに設定されている必要あり。

Ticket認証（ASPでの実装例）

- ‘Use MS XML Object for interacting with QVPX
  Dim xmlHttp 'used for connectivity to QVPX
  Set xmlHttp = Server.CreateObject("MSXML2.ServerXMLHTTP")

- ‘ Construct XML Request
  Dim METHOD
  METHOD = "<Global method=""GetTicket"">“
  METHOD = METHOD & "<UserId>" & USERID & "</UserId>“
  METHOD = METHOD & "</Global>“

- ' Submit request to the QVPX via HTTP
  xmlHttp.open "POST",QVPX,false,Username,Password
  xmlHttp.setRequestHeader "Content-Type","text/xml“
  xmlHttp.setRequestHeader "Content-Length", Len(METHOD)
  xmlHttp.Send METHOD

- Select case xmlHttp.Status
  Case 200:
  Dim TICKET
  TICKET = xmlHttp.responseXML.documentElement.text
  If Len(TICKET) = 0 then
  response.write(" The credentials for QVPX (" & QVPX & ") are not a member of QlikView Administrators.")
  else
  '--
  ' Successful response from GetTicket
  response.write("<a target=""_new"" href=""/QvAJAXZfc/opendoc.htm?document=" + DOCUMENT _
  + "&ticket=" + TICKET + """>Open " + DOCUMENT + " as " + USERID + " using ZeroFootprint Client")
  '--
  End If

- Case 401:
  response.write(" The credentials for QVPX (" & QVPX & ") are incorrect. “)
  Case 403:
  response.write(" The credentials for QVPX (" & QVPX & ") are incorrect. ")
  Case 404:
  response.write(" The URL for QVPX (" & QVPX & ") is incorrect. ")
  Case 503:
  response.write(" The QVPX (" & QVPX & ") is unavailable. ")
  Case Other
  response.write(" An error (" & xmlHttp.Status & ") occurred trying to access QVPX ")
  End Select

Request a Time Limited Ticket:

```
<Global method="GetTicket">
   <UserId>mrw</UserId>
<Global>
```

The Answer should look something like:

```
<Global>
  <_retval_>BDF5D96AB86(40ch)11DA99460383</_retval_>
</Global>
```

認証④ Custom Directory

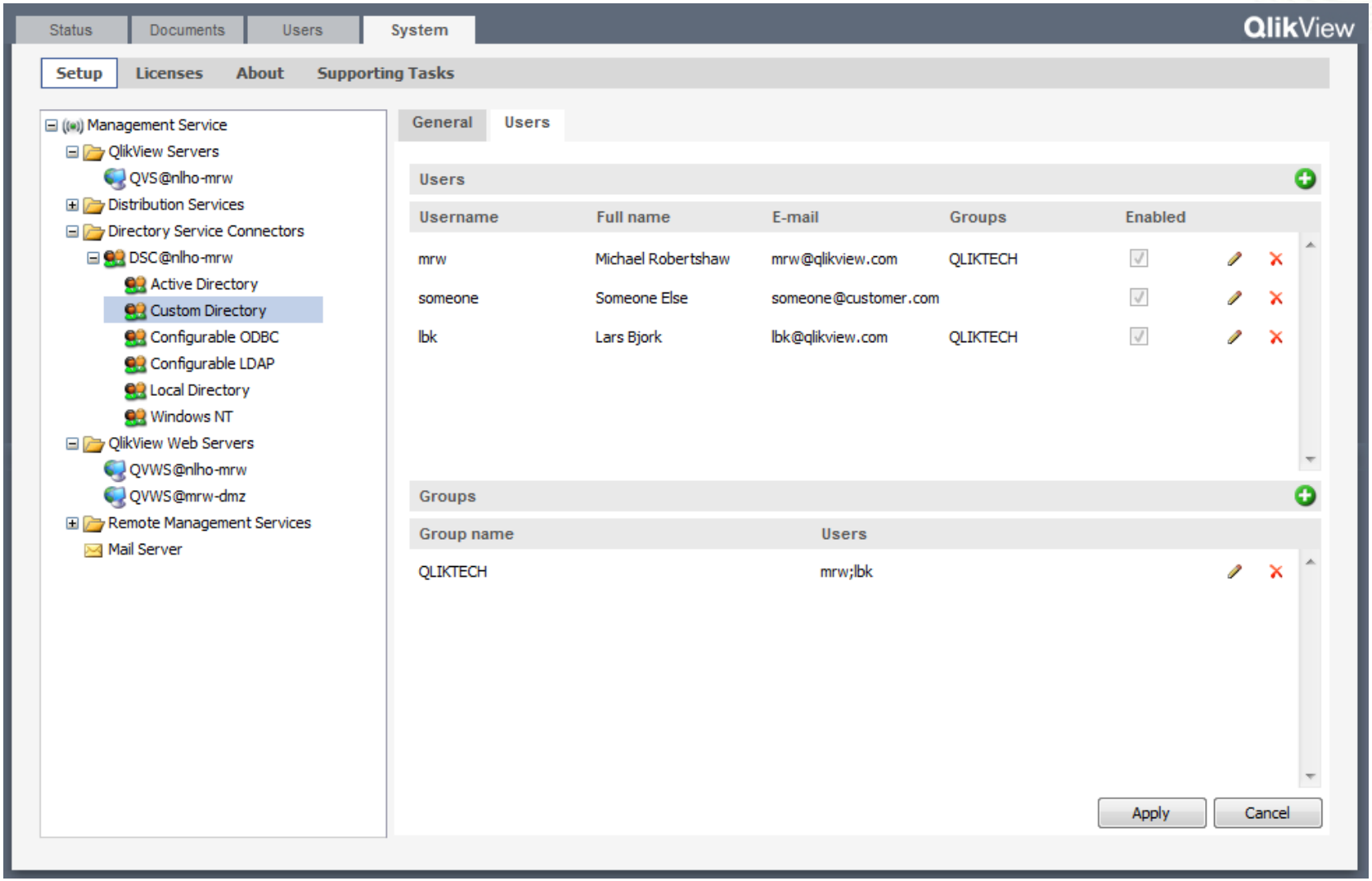
ユーザー識別方法

Custom Directoryでは、ユーザーおよびグループをXMLリポジトリとして格納

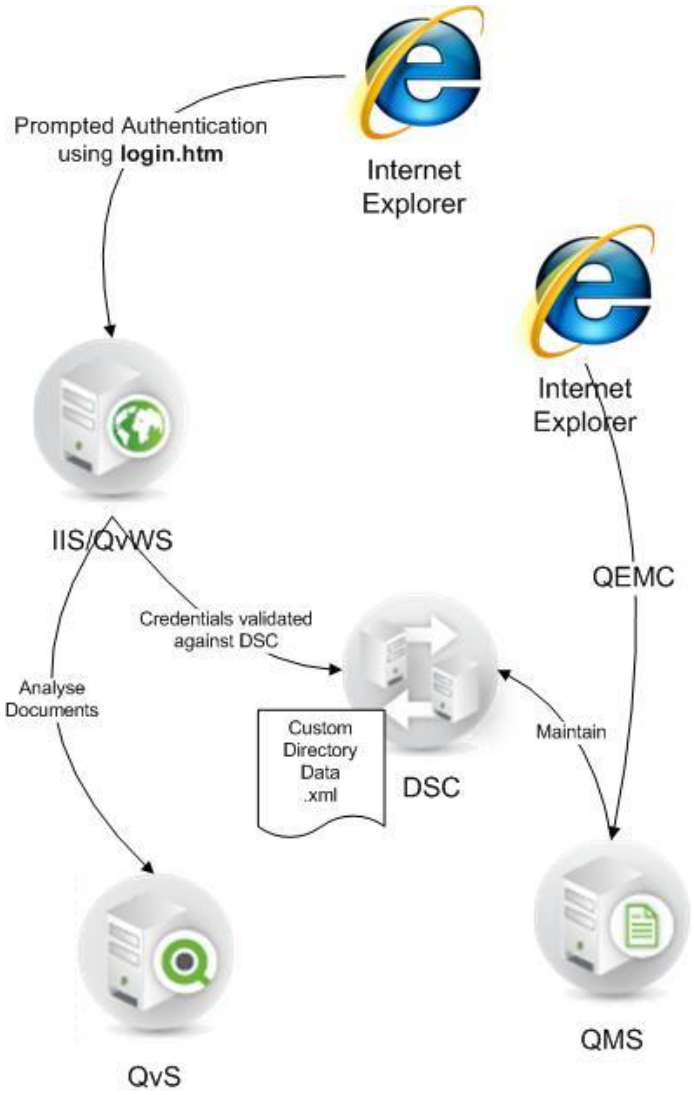
- QEMC または WSDL APIを用いて管理
- Directory Service Connector (DSC)経由でのアクセス
- %ProgramData%¥QlikTech¥DirectoryServiceConnector¥CustomDirectoryData.xmlに格納

http://localhost/qlikview/login.htmにて、認証情報を入力し、DSC経由で評価後、AccessPointが表示される

Custom Directory

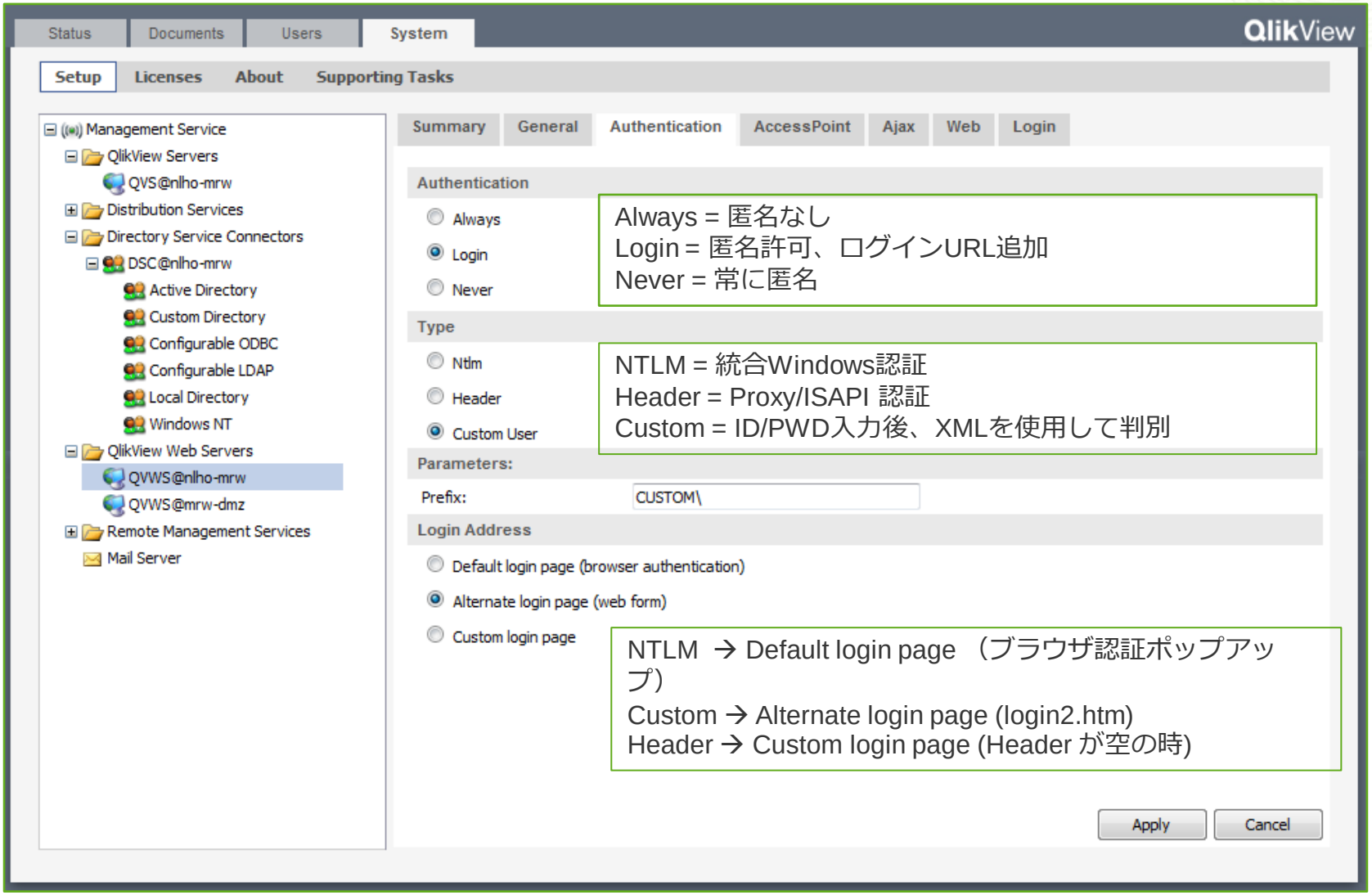


Custom Directory

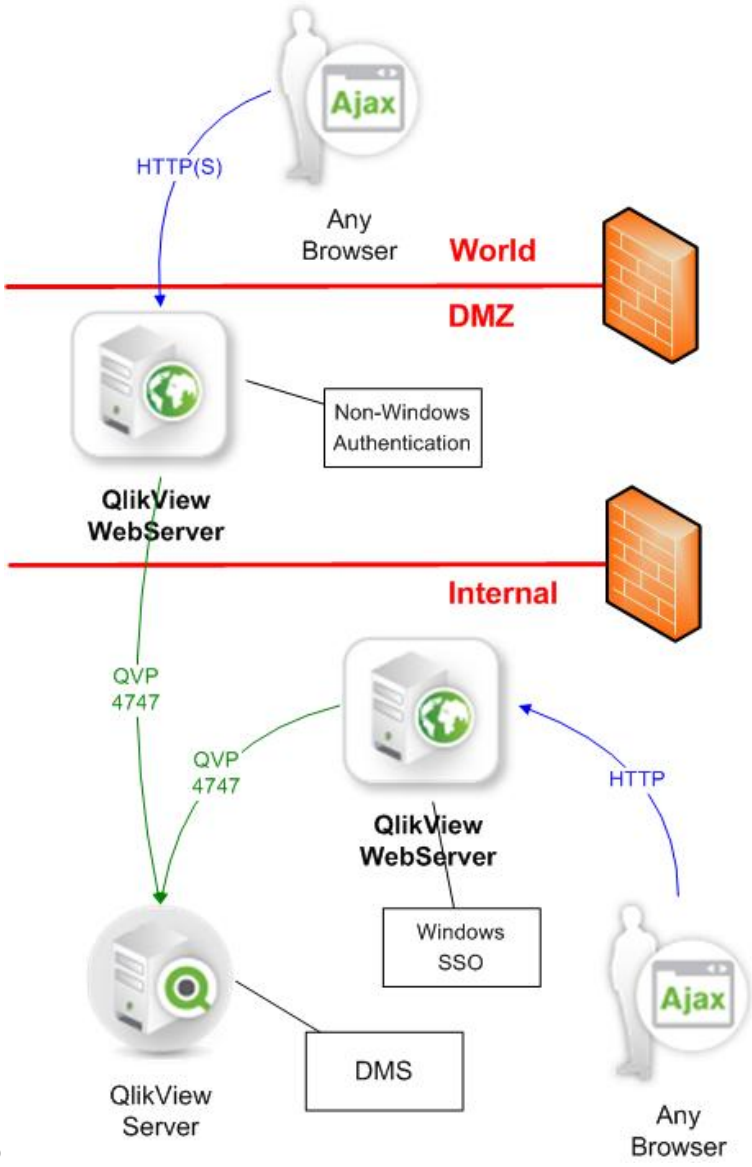


- ユーザーは、QEMCまたはDSC(4735)に送信されるWebServiceリクエスト経由で管理される
- WebServerは、 “login.htm” ページでID/パスワードの入力を促し、DSCへのクエリを用いて識別する
- カスタムユーザーは、“Custom¥”という名前の疑似ドメイン名で管理される(変更可能)
- Custom Directory を利用するにはDMS モードが必須 (SBEでは利用不可)
- Custom Directoryは、Publisher Distribution Task (email lookup)でも利用可能

認証 – QEMCの設定


Status
Documents
Users
System
QlikView
Setup
Licenses
About
Supporting Tasks
Management Service
QlikView Servers
QVS@nlho-mrw
Distribution Services
Directory Service Connectors
DSC@nlho-mrw
Active Directory
Custom Directory
Configurable ODBC
Configurable LDAP
Local Directory
Windows NT
QlikView Web Servers
QVWS@nlho-mrw
QVWS@mrw-dmz
Remote Management Services
Mail Server
Summary
General
Authentication
AccessPoint
Ajax
Web
Login
Authentication
Always
Login
Never
Always = 匿名なし
Login = 匿名許可、ログインURL追加
Never = 常に匿名
Type
Ntlm
Header
Custom User
NTLM = 統合Windows認証
Header = Proxy/ISAPI 認証
Custom = ID/PWD入力後、XMLを使用して判別
Parameters:
Prefix:
CUSTOM\
Login Address
Default login page (browser authentication)
Alternate login page (web form)
Custom login page
NTLM → Default login page （ブラウザ認証ポップアップ）
Custom → Alternate login page (login2.htm)
Header → Custom login page (Header が空の時)
Apply
Cancel

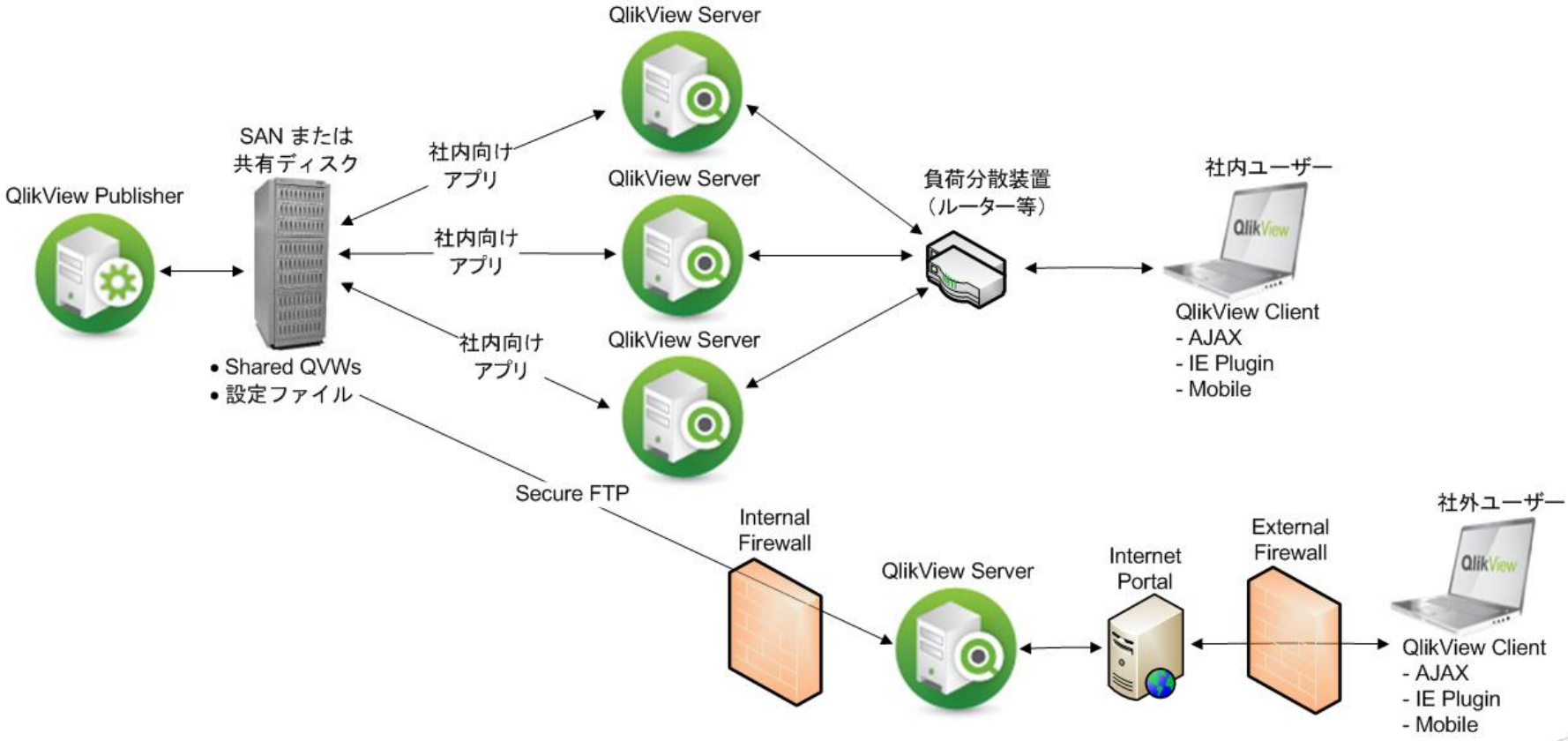
複数Web Tier構成例



- 社外ユーザー、社内ユーザー用にそれぞれWebServerをたてる例
- 社外向けサイトにはSSLを利用、Custom Directory 認証とする
- 社内向けサイトではSSLは使わず、Windows統合認証とする
- 双方のWebServerは、QVSおよびDSCに接続できる必要あり
- QMSから双方の WebServerに接続できる必要あり
- WebServerは、IIS、QVWSいずれも利用可

DMZサーバー構成例

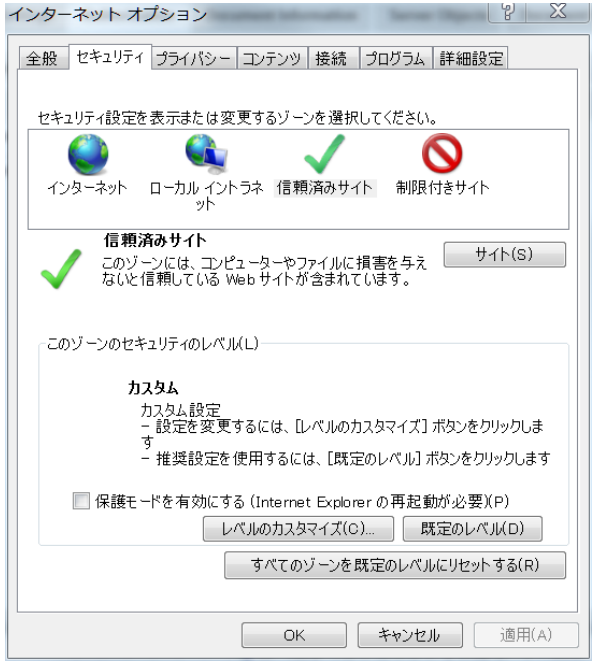
社外のユーザーおよび複数部門に渡るユーザーが多数存在する場合に適した構成例。
社内用アプリケーションは社内の QlikView Server クラスタ環境にロードされて利用されますが、社外ユーザーはDMZの QlikView Server を利用します。QVWは毎晩、社内のSANから社外向け QlikView Server へセキュアなFTPを使って転送されます。よって外部サーバーから社内向けに通信が発生することはありません。



Internet Explorer のセキュリティゾーン

Internet Explorer のセキュリティゾーンにより、クライアントからWebサーバー（IIS）に対してどのように認証を行うかが定義される。

- クライアントが接続しようとするWebサイトがどのゾーンに存在するかにより、認証の手順は異なる
  - インターネットゾーン
    - ✓ IEは常にユーザーに認証ポップアップを表示してWebサーバーへ接続
    - ✓ チケット発行のためのKDCは接続不能と判断され、NTLM認証がデフォルトで使用される
  - イントラネットゾーン
    - ✓ Windows統合認証を使って、ドメインにログオンしているユーザーならそのIDで自動的にWebサイトにログオンしようとする。ユーザー認証が必要な場合や認証失敗した場合、Webサーバーは401ステータスコード(Unauthorized)を返す
    - ✓ KDCが利用可能であればKerberos認証が使われ、失敗すれば代わりにNTLM認証が使用される
    - ✓ 認証に失敗するとIEはIDとパスワード入力用にポップアップを表示する



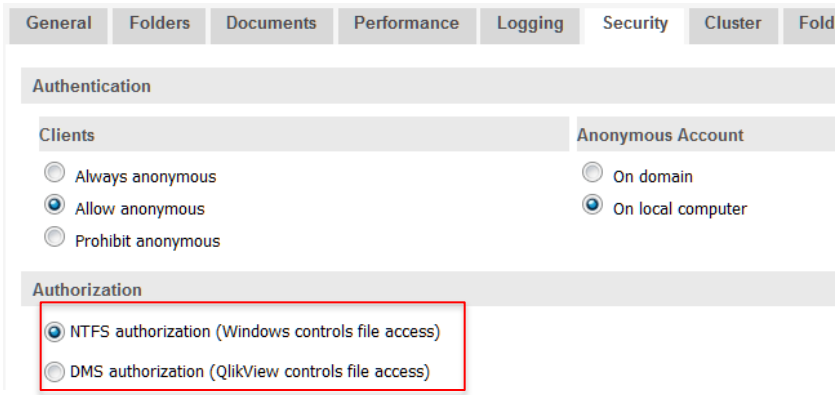
QlikView における「許可」について

認証が済んだら、次に「何を見る/操作する」ことが許可されているのかを判断します

QlikView のアクセス許可レベルは以下の2通りがあります

- **ドキュメントに対するアクセス許可**（=ドキュメント参照権限）
  - NTFS　　Windows によるファイルアクセス権管理
  - DMS　　QlikView によるファイルアクセス権管理
- **データに対するアクセス許可** (=行/列レベルセキュリティ)
  - Section Access
  - QlikView Publisher タスクを使ったデータ削除

ドキュメントに対するアクセス許可
NTFSモードとDMSモード



許可（= Authorization）

- NTFS Authorization
  - 許可は Windows NTFS ファイル システムによって処理される。そのためには、Windows を通して認証が行われる必要がある
- DMS(Document Metadata Service) Authorization
  - QlikView Server 上の DMS 設定によりアクセス許可

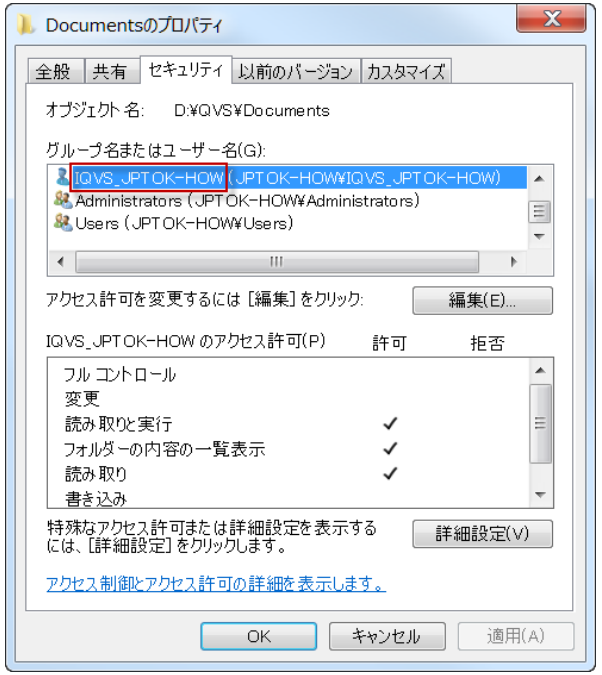
注意:

- QlikView Server は、DMS モード、NTFS モードのいずれか一方のみ使用可能。
- NTFSとDMSの切り替えにはQVSの再起動が必要

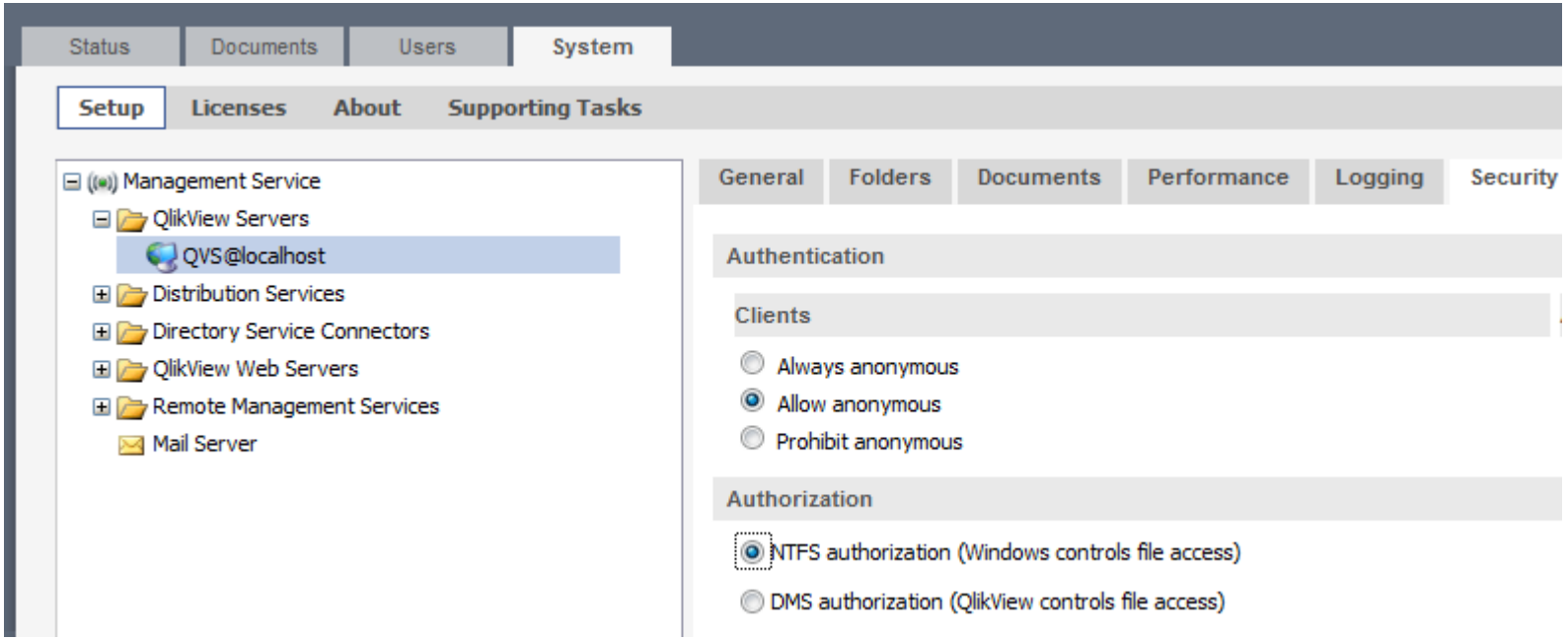
ドキュメントに対するアクセス許可 - NTFS

- QlikView Server がWindows上のQVWに対するアクセス許可をチェックし、キャッシュに保持する（15分間）
- アクセス許可の設定単位
  - フォルダおよび継承
  - QVWファイル毎

  ※Windows Explorerのドキュメントプロパティ＞セキュリティタブで設定
- ファイル許可は、PublisherのDistribution Taskでも設定可能
- 匿名アクセスは、“IQVS_ホスト名”というアカウント（インストール時に自動的に作成）として権限付与される

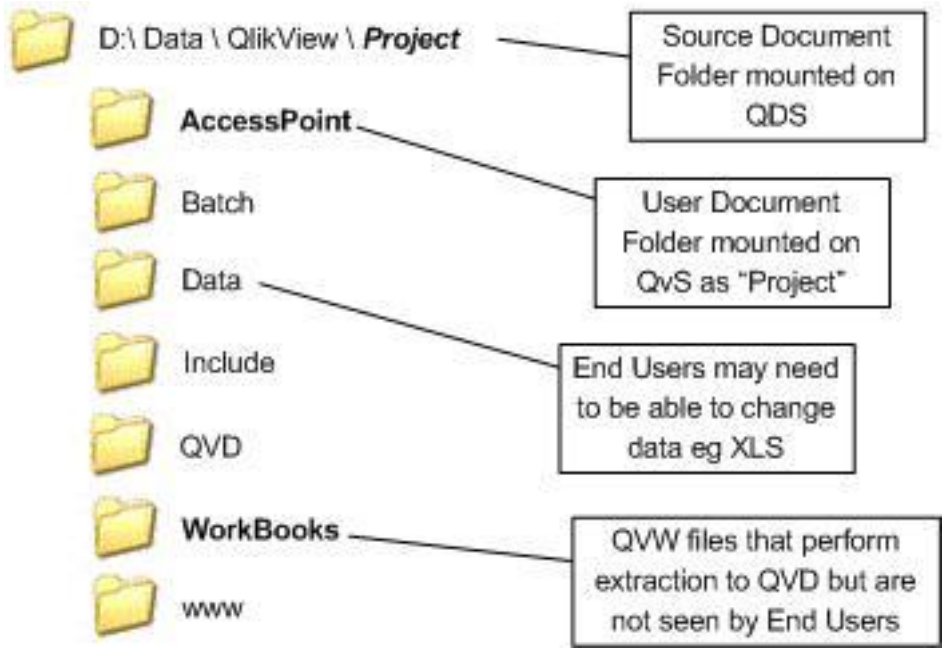


ドキュメントに対するアクセス許可 - NTFS



注）Small Business Edition (SBE) では、NTFSのみ使用可能

ドキュメントに対するアクセス許可 - NTFS



QlikView開発プロジェクトにおけるフォルダ構成例

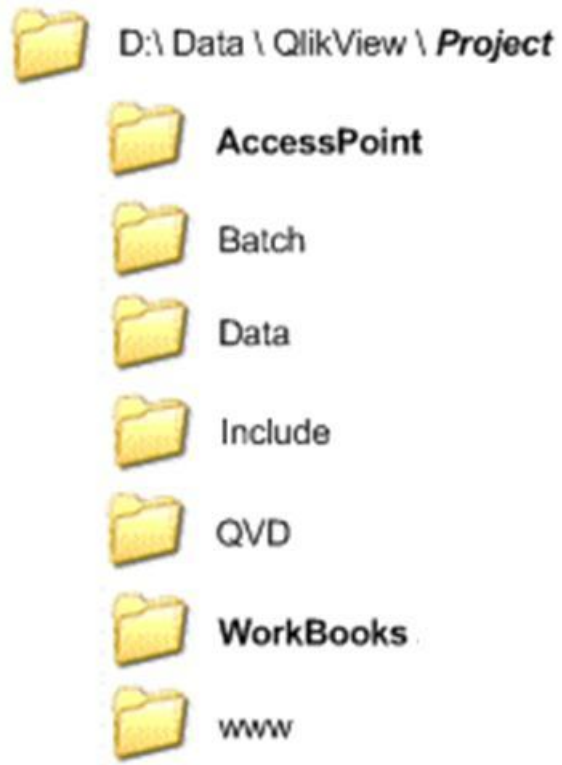
- プロジェクト、顧客、環境ごとに同様のフォルダ構成を保持するようにする
- ファイルの指定は相対パスを使用
  例：XLS, QVD, $(include=..¥include¥global.txt)

- QlikView Distribution Service (QDS)は、すべてのファイルに対して read/write アクセス権が必要
- QlikView Server (QvS) は、AccessPoint フォルダのアクセス権に基づき、ドキュメントをエンドユーザーへ公開

1. D:¥Data¥QlikView の所有者を “QlikView Administrators”に変更
2. D:¥Data¥QlikViewから継承権限を削除
3. 各***Project*** のオーナーや管理者にフルコントロール権限を付与
4. （必要に応じて）AccessPoint フォルダにRead権限を付与
5. Dataフォルダに対し適切なアクセス権を付与し、¥¥fileserver¥***Project*** を共有
6. wwwフォルダに対し、Everyone Read アクセス権を付与

- SBEを使用している場合は、***Project*** フォルダをUser Document Folderとしてマウント

ドキュメントに対するアクセス許可 - NTFS



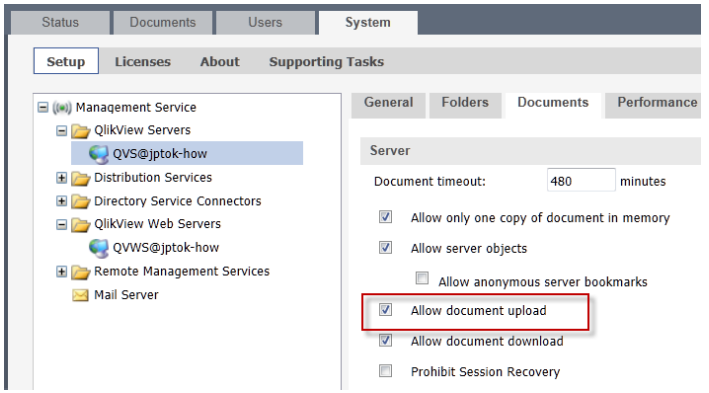
各フォルダの説明

- **Project**：
  - ✓ 環境やプロジェクト名（SiB,Dev,UAT,Prod等）
- **AccessPoint**：
  - ✓ ユーザードキュメントフォルダ。AccessPointから参照可能なQVWファイルを格納。ソースフォルダからPublisher配信されたドキュメントは主にここに格納される。
- **Data**：
  - ✓ CSVやExcel等のデータソース
- **Include**：
  - ✓ グローバル変数や多言語対応用の定義、サブルーチン等、共通化可能なスクリプトをテキストファイルとして格納。QVWの中では$(include=..¥include¥xxx-global.txt);のようにコールする。
- **QVD**：
  - ✓ ソースドキュメントにより生成/利用されるQVDファイルを格納。
- **WorkBooks**：
  - ✓ データ抽出用のQVWを格納。QDSによりリロードされるが、エンドユーザーからはアクセスできない。
- **www**：
  - ✓ HTMLや画像ファイル、その他マッシュアップ用のファイル

ドキュメントに対するアクセス許可 - DMS

- QlikView Server は、QVWに紐づいたMetaファイルからドキュメントアクセス権を読み込み、15分間キャッシュに保持
- アクセス許可は、QVWファイル毎に個別に記録される（.meta）
- QEMC > User Document > Authorisation タブにて設定（このタブはDMSモードの場合のみ表示される）

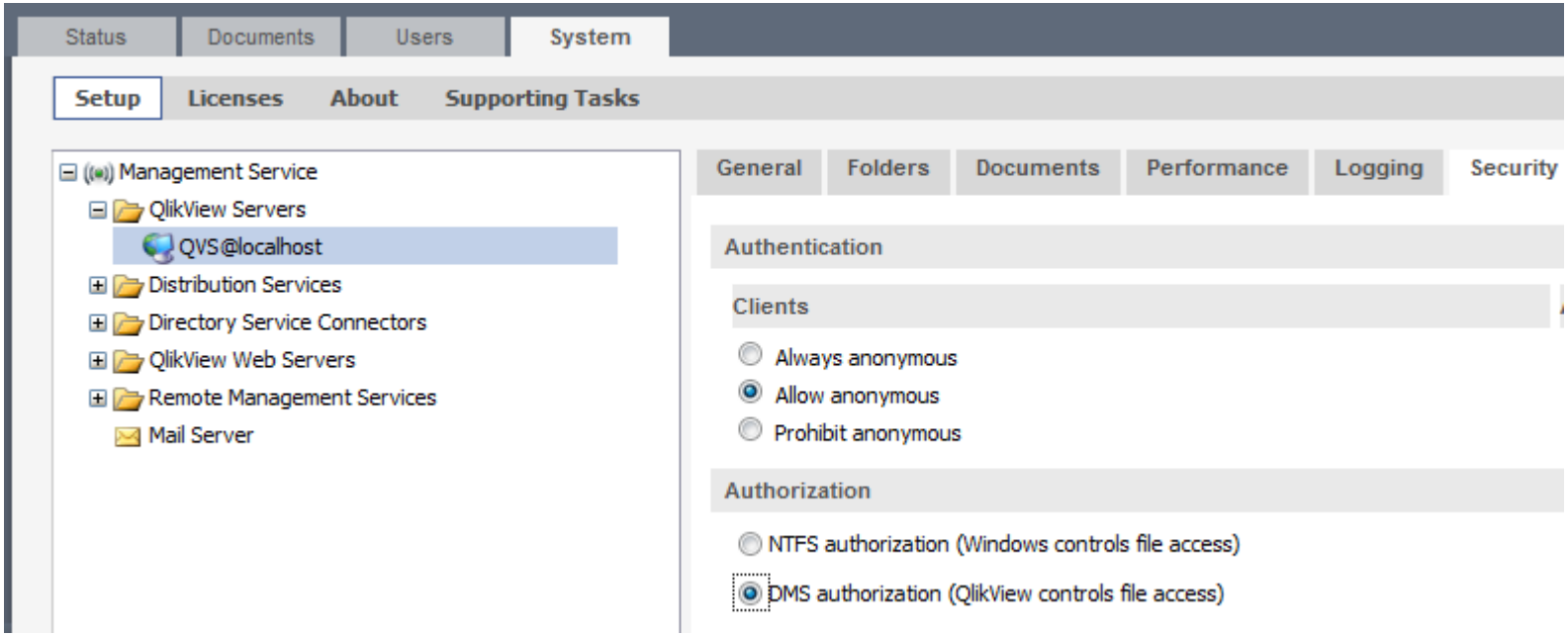
- アクセス許可は、PublisherのDistribution Taskでも設定可能
  - Publisher ライセンスが必要
  - QVSへの配信タスクを設定する場合にのみ、Metaファイルへの書き込みが可能
  - QVSにて “Allow document upload” が有効になっている必要あり



- DMSでは、非Windowsユーザーにも対応
  - 例：Headerまたはチケットの認証情報で識別
- グループに対するアクセス権も付与可能。DSCがグループメンバーシップリポジトリにアクセスできるよう正しく設定されている必要あり

注）SBEでは、DMS利用不可

ドキュメントに対するアクセス許可 - DMS



DMSに変更するとAccess Pointからドキュメントが見えなくなります。
これはなぜでしょう？

データに対するアクセス許可 – Section Access

- SectionAccess実装時には以下の2つのセクションに分かれる
  - Section Access
    - ✓ このセクション以下で、SectionAccessの定義を明示的に宣言
    - ✓ ドキュメントに対する認証
    - ✓ ユーザーと項目値の組み合わせを定義
  - Section Application
    - ✓ このセクション以下で実際のデータモデルを記述

データに対するアクセス許可 – Section Access – 項目

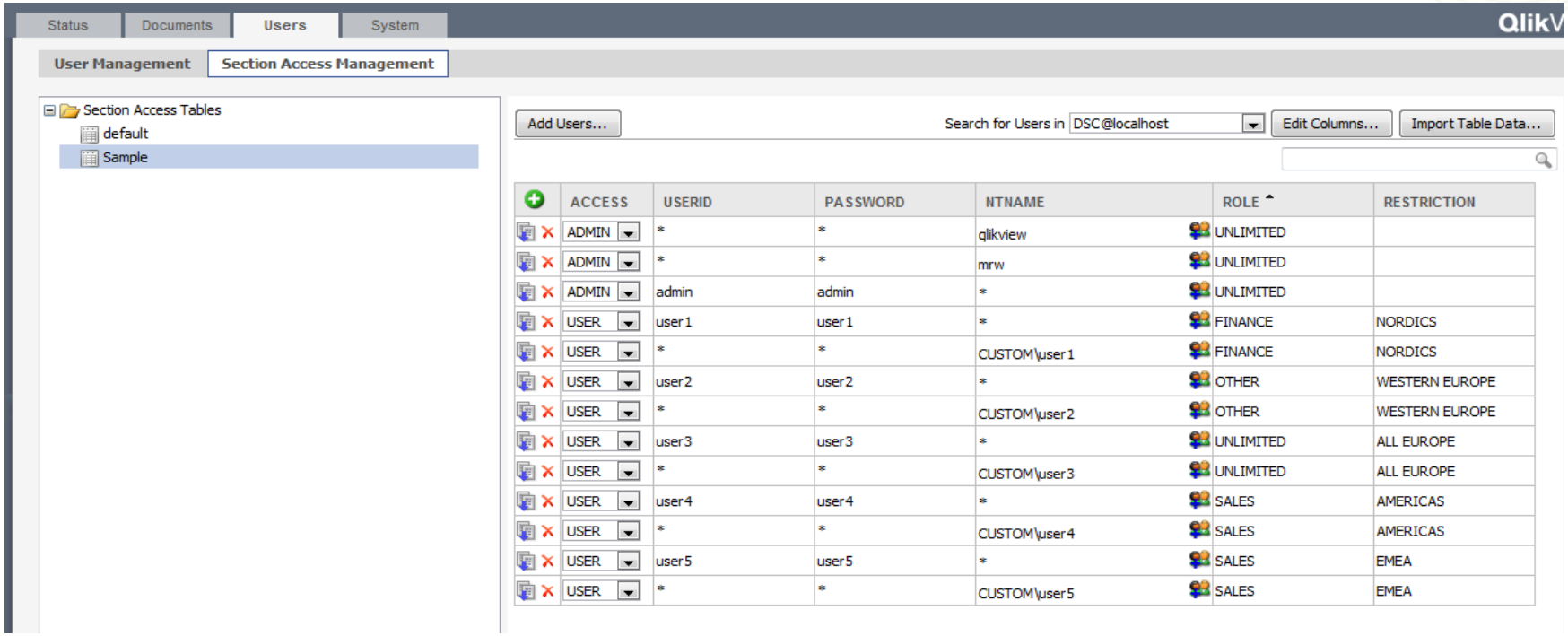
Section Access項目	
ACCESS	USER または ADMIN 対応するユーザーに与えられるアクセス権を定義する項目。QlikView Desktopのドキュメントプロパティに対するアクセス権に影響
USERID	許可されるユーザー ID を含む項目。 QlikView がユーザー ID の入力を要求し、この項目の値と比較します。 このユーザー ID は、Windows のユーザー ID と同じではありません。
PASSWORD	許可されるパスワードを含む項目。 QlikView がパスワードの入力を要求し、この項目の値と比較します。 このパスワードは、Windows のパスワードと同じではありません。
SERIAL	QlikView のシリアル番号に対応する番号を含む項目。 例: 4900 2394 7113 7304 QlikView がユーザーのシリアル番号を確認し、それをこの項目の値と比較します。
NTNAME	Windows NT ドメインのユーザー名またはグループ名に対応する文字列を含む項目。 QlikView が OS からログオン情報を取得し、それをこの項目の値と比較します。
NTDOMAINSID	Windows NT ドメインの SID に対応する文字列を含む項目。 例: S-1-5-21-125976590-4672381061092489882 QlikView が OS からログオン情報を取得し、それをこの項目の値と比較します。
NTSID	Windows NT の SID を含む項目。 例： S-15-21-125976590-467238106-1092489882-1378 QlikView が OS からログオン情報を取得し、それをこの項目の値と比較します。
OMIT	特定のユーザーに対して項目自体を表示にする。 ワイルドカードを使用したり、項目を空にしたりできます

データに対するアクセス許可 – Section Access – ウィザード

スクリプトエディタのメニューから、
挿入 > Section Access > インラインロード

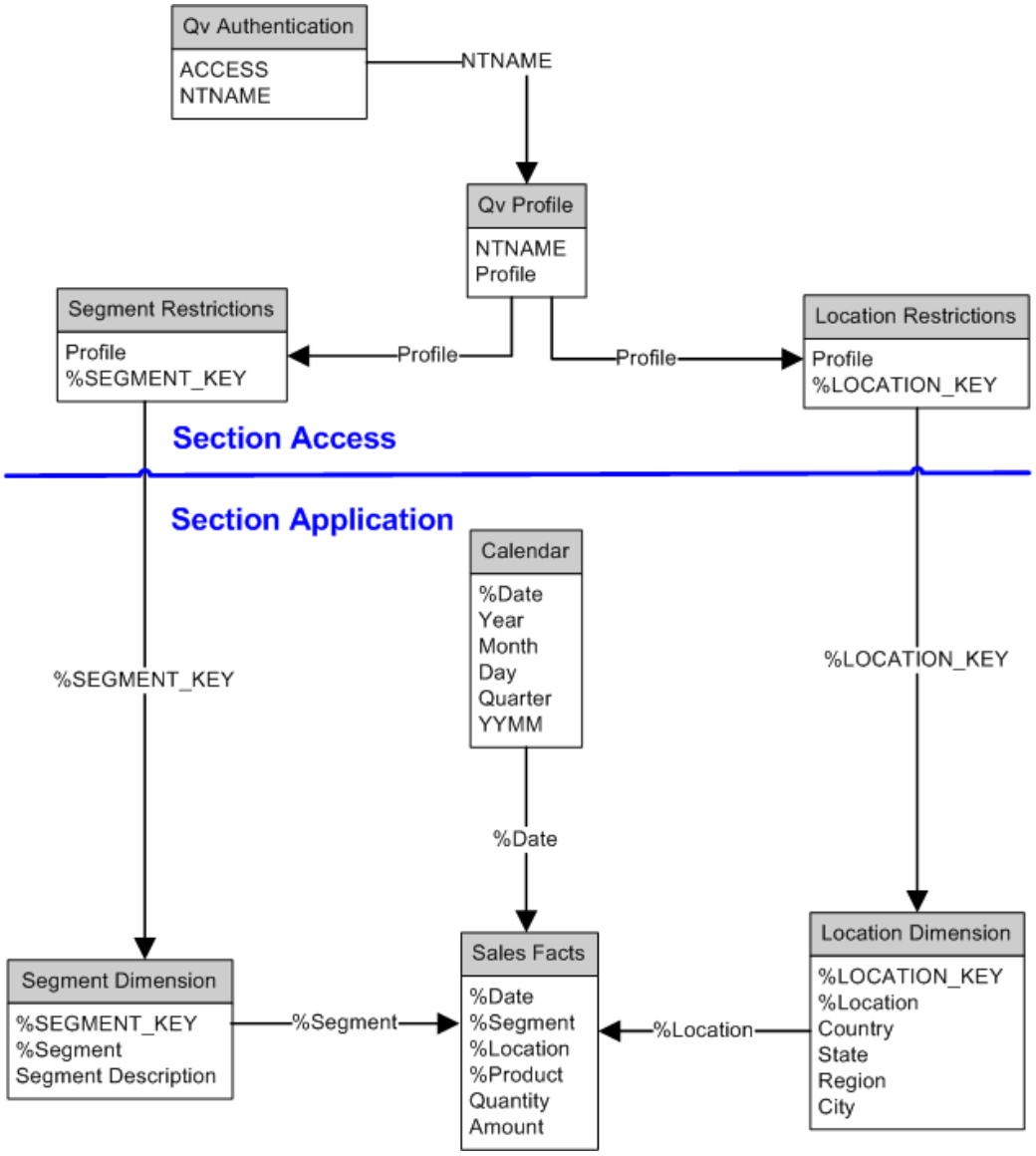
- “隠しロードスクリプト”タブを作成し、SectionAccessを記述。
- 隠しロードスクリプトは常に最初のタブの前に配置
- USERID & PASSWORDを使うより、NTNAMEを使う方が安全
- ログインユーザーの確認
  - ✓NTNAME：OSUser()
  - ✓USERID：QvUser()

データに対するアクセス許可 – Section Access - QMS



- SectionAccessテーブルは、QEMC上で一元管理可能
- QEMC上のSectionAccessテーブルへは以下のURLからアクセス
  http://server:4780/QMS/AuthTable
- 要 Publisher ライセンス

データに対するアクセス許可 – Section Access – データ削除



- Section Access内の項目名が大文字あるいはダブルバイトの場合のみ、 Section Applicationから参照可能
- %SEGMENT_KEY 項目内の値も大文字である必要あり
- Section Access部分にも連想データモデルが適用される。また、記述するテーブル数は複数でも可能
- この例ではSection Application部分で循環参照は発生しない

データに対するアクセス許可 – Section Access – 応用

```
// Load all Cost Centers "as is" from Source Data
[Cost Center Restrictions]:
LOAD DISTINCT
    UPPER([Cost center]) as [%COST CENTER RULE]
   ,[Cost center]       as [%COST CENTER]
FROM Data.xlsx
    (ooxml, embedded labels, table is [Chart of Accounts]);

// Find all Cost Center Rules that contain wildcards
[Patterns]:
LOAD DISTINCT [%COST CENTER RULE] as Pattern
FROM [http://localhost:4780/QMS/AuthTable]
    (html, utf8, embedded labels, table is [Cost Center Rules])
WHERE Index([%COST CENTER RULE],'*') > 0;

// Loop through all Patterns, appending matching Cost Centers
LET NumberOfPatterns = NoOfRows('Patterns');
FOR i = 0 to $(NumberOfPatterns) -1
    LET Pattern = Peek('Pattern',$(i),Patterns);

    CONCATENATE ([Cost Center Restrictions])
    LOAD '$(Pattern)'   as [%COST CENTER RULE]
       ,[Cost center]  as [%COST CENTER]
    FROM Data.xlsx
        (ooxml, embedded labels, table is [Chart of Accounts])
    WHERE WildMatch([Cost center],'$(Pattern)');

NEXT;
DROP TABLE [Patterns];
```

- SectionAccessのROLE列のワイルドカード(*)は、SectionAccess内にリストされている全ROLEを参照
- SectionApplicationの実データより少ない可能性があるため要注意
- ワイルドカードを使用するときは上記の点に注意し、意図した結果が得られるか確認すること

データに対するアクセス許可 – Publisher タスクによるデータ削除

例1:

X社の各営業は、商談に向かう際に担当顧客の購買傾向を分析したい。必要なのは自分の担当顧客の購買履歴のみで、ネットワークアクセスは不可能な場合が多い。

例2:

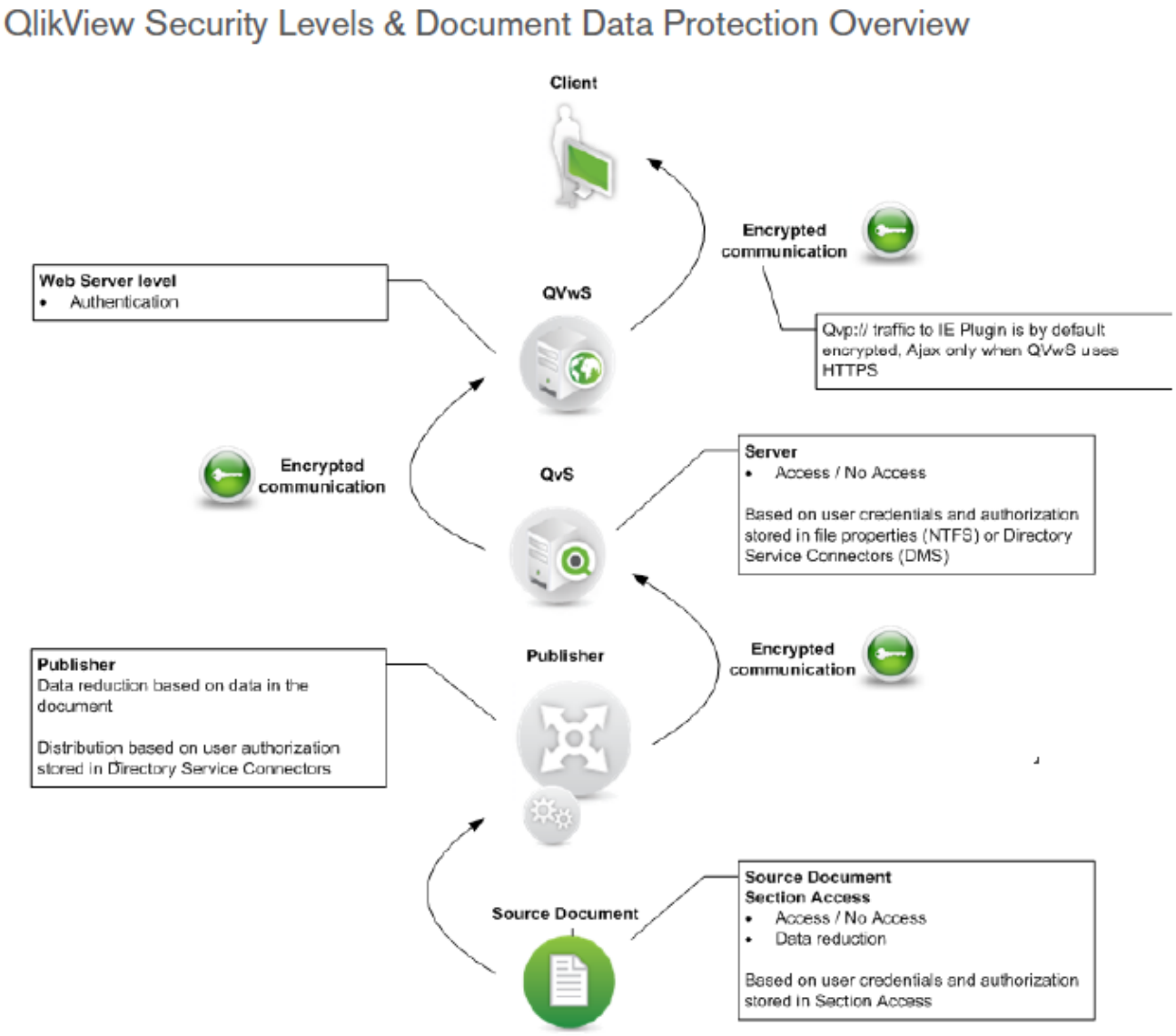
Y銀行には各支店ごとに多くのエンドユーザーがおり、各エンドユーザーは自支店のデータにしかアクセスできないようにしたい。

- Publisher の“Reduction and Distribution” タスクを使えば、一つの大きなドキュメントをデータのサブセットごとに分割して複数ドキュメントを作成することが可能
- ドキュメントサイズが小さければ、パフォーマンスの点からも向上が見込めるし、ファイルサーバーへ配置したり、メール添付して配信するのも容易。（データが少なければRAM消費量も少ない）
- 要 Publisher ライセンス

まとめ

- 認証
  - 統合Windows認証
  - HTTP Header
  - Ticket認証
  - Custom Directory
- ドキュメントに対するアクセス許可
  - NTFS
  - DMS
- データに対するアクセス許可
  - Section Access
  - Document Reduction

参考：QlikViewのセキュリティレベル



Thank you