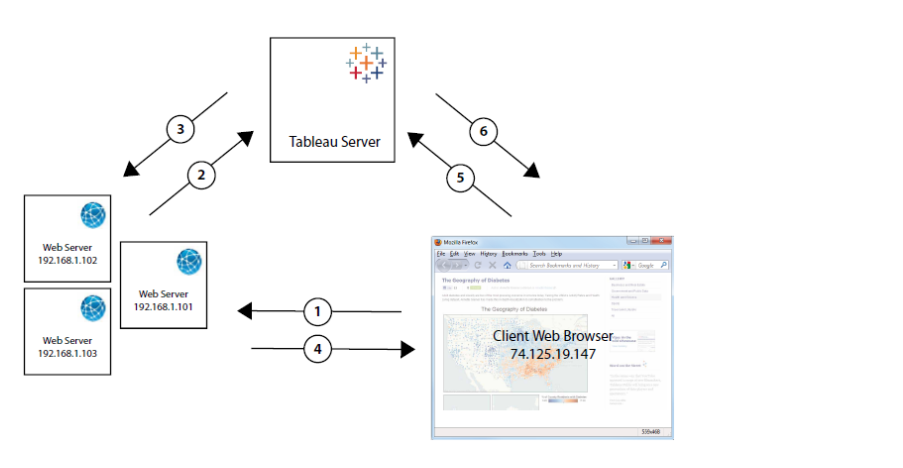
**Web 服务器和 Tableau Sever 之间的连接**

验证的工作原理



1.**用户访问网页**：当用户访问具有嵌入式 Tableau Server 视图的网页时，该网页会向您的 Web 服务器发送一个 GET 请求，要求提供该网页的 HTML。

2. **Web 服务器 POST 到 Tableau Server**：Web 服务器向受信任的 Tableau Server（例如 http://tabaserver/trusted，而非 http://tabserver）发送了 POST 请求。该 POST 请求必须有一个 username 参数。username 值必须是 Tableau Server 许可用户的用户名。如果服务器在运行多个站点，且视图在 Default 站点之外的其他站点上，则 POST 请求还必须包含 target\_site 参数。

3. **Tableau Server 创建票证**：Tableau Server 检查发送 POST 请求的 Web 服务器的 IP 地址或主机名（上图中的 192.168.1.XXX）。如果该服务器已设置为受信任主机，则 Tableau Server 将创建一个票证，形式为包含 24 个字符的唯一字符串（URL 安全、Base64 编码）。Tableau Server 使用该票证来响应 POST 请求。如果存在错误并且无法创建票证，则 Tableau Server 将使用值“-1”进行响应。

4. **Web 服务器将 URL 传递给浏览器**：Web 服务器使用该视图的 URL 或其对象标记（如果是嵌入式视图）构造该视图的 URL，并将其插到该网页的 HTML 中。包括票证（例如 http://tabserver/trusted/<ticket>/views/requestedviewname）。Web 服务器将该网页的所有 HTML 传递回客户端的 Web 浏览器

5. **浏览器从 Tableau Server 请求视图**：客户端 Web 浏览器使用 GET 请求向 Tableau Server 发送请求，该 GET 请求包含带有票证的 URL

6. **Tableau Server 兑换票证**：Tableau Server 检查是否 Web 浏览器请求了含有票证的 URL，并兑换该票证。Tableau Server票证必须在发出后三分钟内兑换。票证必须在发出后三分钟内兑换。兑换票证后，Tableau Server 将用户登录，从 URL 中删除该票证，并发送回该嵌入式视图的最终 URL。

-------------TABLEAU SERVER 端配置

**一、**将 Web 服务器 IP 地址添加到 wgserver.trusted\_hosts

#### Step 1

单击“开始”按钮，并选择“所有程序”>“附件”>“命令提示符”。右键单击“命令提示符”，并选择“以管理员身份运行”

#### Step 2

在“命令提示符”窗口中，根据 Tableau 和操作系统的版本键入以下命令之一：

* cd "C:\Program Files\Tableau\Tableau Server\<version>\bin"
* cd "C:\Program Files (x86)\Tableau\Tableau Server\<version>\bin"

注意：将 <version> 替换为您运行的 Tableau Server 的版本。

Step 3

键入以下命令：

* tabadmin set wgserver.trusted\_hosts <ip-address>
* tabadmin config
* tabadmin restart

注意：将 <ip-address> 替换为 Web 服务器的 IP 地址。IP 地址必须为 IPv4 地址或计算机名称。

### 添加测试用户

#### Step 1

使用您的管理员用户名和密码登录到 Tableau Server。

#### Step 2

在“管理”选项卡中单击“用户”，以创建用户帐户。测试用户必须已获得许可成为“交互者”。

-------------JAVA 访问tableau通用类

**package** com.sto;

**import** java.io.BufferedReader;

**import** java.io.IOException;

**import** java.io.InputStreamReader;

**import** java.io.OutputStreamWriter;

**import** java.net.URL;

**import** java.net.URLConnection;

**import** java.net.URLEncoder;

**import** javax.servlet.ServletException;

**import** javax.servlet.http.HttpServletRequest;

**import** javax.servlet.http.HttpServletResponse;

/\*\*

\* Servlet implementation class for Servlet: TableauServlet

\*

\*/

**public** **class** TableauServlet **extends** javax.servlet.http.HttpServlet {

**private** **static** **final** **long** ***serialVersionUID*** = 1L;

**public** TableauServlet() {

**super**();

}

**protected** **void** doGet(HttpServletRequest request, HttpServletResponse response) **throws** ServletException, IOException {

//final String user = "workgroupadmin";

**final** String user = "test"; //用户名

**final** String wgserver = "192.168.123.102";//tableau server站点IP

**final** String remoteAddr = "192.168.123.102";//本地客户端IP地址

**final** String dst = "views/\_0/sheet0\_1?:iid=1";//tableau server视图URL 可配置到本地数据库中

**final** String params = ":embed=yes&:toolbar=yes";

// ticket 这个tableau server 返回的票证 如果是-1则没有成功

// 票证是访问tableau server 唯一凭证

String ticket = getTrustedTicket(wgserver, user, remoteAddr);

//System.***out***.println("http://" + wgserver + "/trusted/" + ticket + "/" + dst

);

**if** ( !ticket.equals("-1") ) {

response.setStatus(HttpServletResponse.***SC\_MOVED\_TEMPORARILY***);

response.setHeader("Location", "http://" + wgserver + "/trusted/" + ticket + "/" + **dst** + "?" + params);

}

**else**

// handle error

**throw** **new** ServletException("Invalid ticket " + ticket);

}

// the client\_ip parameter isn't necessary to send in the POST unless you have

// wgserver.extended\_trusted\_ip\_checking enabled (it's disabled by default)

**private** String getTrustedTicket(String wgserver, String user, String remoteAddr)

**throws** ServletException

{

OutputStreamWriter out = **null**;

BufferedReader in = **null**;

**try** {

// Encode the parameters

StringBuffer data = **new** StringBuffer();

data.append(URLEncoder.*encode*("username", "UTF-8"));

data.append("=");

data.append(URLEncoder.*encode*(user, "UTF-8"));

data.append("&");

data.append(URLEncoder.*encode*("client\_ip", "UTF-8"));

data.append("=");

data.append(URLEncoder.*encode*(remoteAddr, "UTF-8"));

// Send the request

URL url = **new** URL("http://" + wgserver + "/trusted");

URLConnection conn = url.openConnection();

conn.setDoOutput(**true**);

out = **new** OutputStreamWriter(conn.getOutputStream());

out.write(data.toString());

out.flush();

// Read the response

StringBuffer rsp = **new** StringBuffer();

in = **new** BufferedReader(**new** InputStreamReader(conn.getInputStream()));

String line;

**while** ( (line = in.readLine()) != **null**) {

rsp.append(line);

}

**return** rsp.toString();

} **catch** (Exception e) {

**throw** **new** ServletException(e);

}

**finally** {

**try** {

**if** (in != **null**) in.close();

**if** (out != **null**) out.close();

}

**catch** (IOException e) {}

}

}

----测试

通过 "http://" + wgserver + "/trusted/" + ticket + "/" + dst 输出的地址在浏览器中打开 即可以访问报表不需要再次登录