

表 2-4 总结了最常见的一些方案。回顾一下 2.2 节有助于理解表格中的语法部分。

表2-4 常见的方案格式

方 案	描 述
http	<p>超文本传输协议方案，除了没有用户名和密码之外，与通用的 URL 格式相符。如果省略了端口，就默认为 80。</p> <p>基本格式： <code>http://&lt;host&gt;:&lt;port&gt;/&lt;path&gt;?&lt;query&gt;#&lt;frag&gt;</code></p> <p>示例： <code>http://www.joes-hardware.com/index.html</code> <code>http://www.joes-hardware.com:80/index.html</code></p>
https	<p>方案 https 与方案 http 是一对。唯一的区别在于方案 https 使用了网景的 SSL，SSL 为 HTTP 连接提供了端到端的加密机制。其语法与 HTTP 的语法相同，默认端口为 443。</p> <p>基本格式： <code>https://&lt;host&gt;:&lt;port&gt;/&lt;path&gt;?&lt;query&gt;#&lt;frag&gt;</code></p> <p>示例： <code>https://www.joes-hardware.com/secure.html</code></p>
mailto	<p>Mailto URL 指向的是 E-mail 地址。由于 E-mail 的行为与其他方案都有所不同（它并不指向任何可以直接访问的对象），所以 mailto URL 的格式与标准 URL 的格式也有所不同。因特网 E-mail 地址的语法记录在 RFC 822 中。</p> <p>基本格式： <code>mailto:&lt;RFC-822-addr-spec&gt;</code></p> <p>示例： <code>mailto:joe@joes-hardware.com</code></p>
ftp	<p>文件传输协议 URL 可以用来从 FTP 服务器上下载或向其上载文件，并获取 FTP 服务器上的目录结构内容的列表。</p> <p>在 Web 和 URL 出现之前 FTP 就已经存在了。Web 应用程序将 FTP 作为一种数据访问方案使用。URL 语法遵循下列通用格式。</p> <p>基本格式： <code>ftp://&lt;user&gt;:&lt;password&gt;@&lt;host&gt;:&lt;port&gt;/&lt;path&gt;;&lt;params&gt;</code></p> <p>示例： <code>ftp://anonymous:joe%40joes-hardware.com@prep.ai.mit.edu:21/pub/gnu/</code></p>
rtsp, rtspu	<p>RTSP URL 是可以通过实时流传输协议（Real Time Streaming Protocol）解析的音 / 视频媒体资源的标识符。</p> <p>方案 rtspu 中的 u 表示它是使用 UDP 协议来获取资源的。</p> <p>基本格式： <code>rtsp://&lt;user&gt;:&lt;password&gt;@&lt;host&gt;:&lt;port&gt;/&lt;path&gt;</code> <code>rtspu://&lt;user&gt;:&lt;password&gt;@&lt;host&gt;:&lt;port&gt;/&lt;path&gt;</code></p> <p>示例： <code>rtsp://www.joes-hardware.com:554/interview/cto_video</code></p>

38