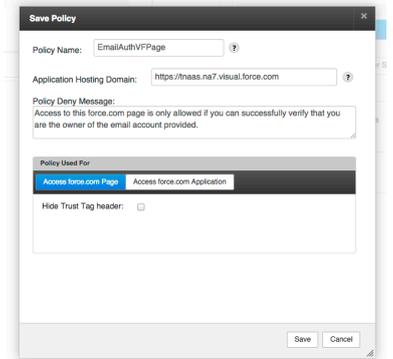
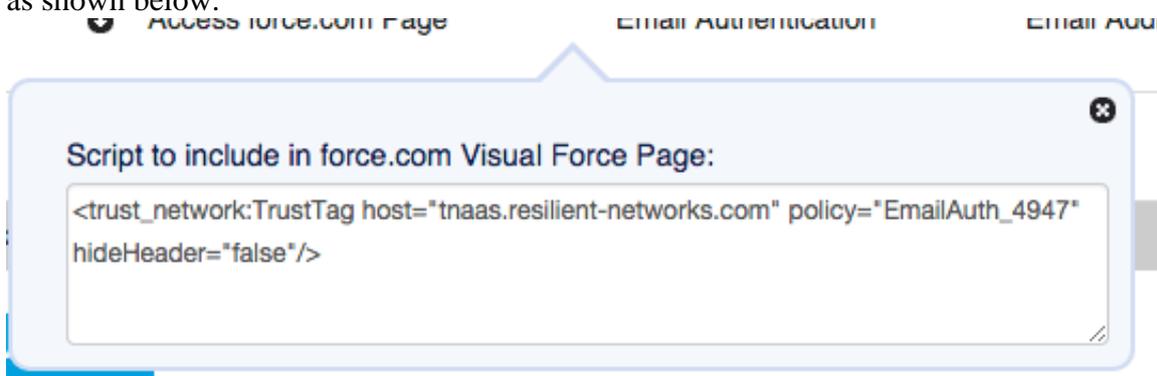


The Salesforce administrator/developer can create policies that use any combination of Resilient Access out of the box authorities and custom authorities created by them as described in [Create Policies](#). To use the policy for access control in a Visual Force Page, follow the steps below

1. Define the Policy expression as described in [Create Policies](#) and specify how the authority



2. In the **Save Policy** popup enter the Salesforce instance domain the Salesforce organization is in for the **Application Hosting Domain** box
3. Provide a custom deny message that users accessing the Visual Force page that uses this Policy will see if the policy evaluation results in a DENY (*optional*)
4. Select the **Access force.com Page** tab in the **Policy Used For** section.
 - o If you wish to suppress the rendering of the TNaaS header that provides the **Logout** button and the user identity attributes click on the **Hide Trust Tag header** checkbox. For a policy that controls access to the force.com application it is recommended to keep the **Hide Trust Tag header** checkbox unchecked
 - o For a force.com page using this policy that is embedded inside another page through `<apex:iframe>` could check the **Hide Trust Tag header** checkbox to not display the Resilient Access header when the page is rendered
5. In the TNaaS policy list click on the  icon in the **Policy Used For** column to display the popup as shown below.



6. Copy the Trust Tag Visual Force component tag and embed in a Visual Force page as shown below. When the page is loaded in a browser it will go through the policy evaluation which will

The screenshot shows the Salesforce Page Editor interface for a Visualforce Page named 'ExampleVFPPage'. On the left is a navigation sidebar with sections for 'Salesforce1 Setup', 'Force.com Home', 'Administer', and 'Build'. The 'Administer' section is expanded, showing various administrative tasks. The main area is titled 'Page Edit' and contains a 'Page Information' section with fields for Label, Name, and Description. The Name and Description fields are filled with 'ExampleVFPPage' and 'This page show how to control access to a Visual Force Page using TNaaS' respectively. Below this is a 'Visualforce Markup' section showing the following code:

```
1 <apex:page >  
2 <trust_network:TrustTag host="tnaas.resilient-networks.com" policy="EmailAuth_4947" hideHeader="false"/>  
3 <apex:iframe frameborder="false" src="http://www.resilient-networks.com" scrolling="auto" width="1024" height="768"/>  
4 </apex:page>
```

control access to the page.