

## Permission denied for javascript methods, SSL security error between parent and child windows

Posted At : 2 November 2010 15:53 | Posted By : Shaun McCran

Related Categories: Security, Javascript

I recently integrated a postcode lookup service into a checkout process, it constituted a pop up window, with a Webservice http call to return a JSON object of postcode data.

The data itself was returning successfully, and is output into a select field, so that the user can choose one of the address records from the many returned.

The problem I had arose when I ran a script to write the selected address data back from the pop up window to the parent window. Something like this:

```
</script type="text/javascript">
$(document).ready(function() {

    $('#submitButton').click(function() {

        var selectedPcode = $('#address').val();

        if (selectedPcode == undefined) {
            alert('Please select an address')
        }

        else {
            //split the string
            var mySplitResult = selectedPcode.split(",");

            var street = mySplitResult[0];
            var area = mySplitResult[1];
            var town = mySplitResult[2];

            street = jQuery.trim(street);
            area = jQuery.trim(area);
            town = jQuery.trim(town);

            // set the parent form field values
            window.opener.document.form.evAddress1.value = street;
            window.opener.document.form.evAddress2.value = area;
            window.opener.document.form.evTown.value = town;
            window.close();
        }
    });
});
```

```
        }  
    });  
});  
</s/crypt>
```

The code above will just split out the address parts and write them out to the corresponding fields in a form in the parent window. At this point I was seeing an error message:

```
Permission denied for javascript.... Line xxx
```

The problem stems from the fact that the parent window is served under SSL and the pop up was not.

So make sure that your parent and child windows are both served under the same protocol, otherwise I guess it is being stopped as an inject hack, as it appears to be on a different domain.