Magecart Targets Hotel Booking Websites on Mobile

blog.trendmicro.com/trendlabs-security-intelligence/magecart-skimming-attack-targets-mobile-users-of-hotel-chain-booking-websites/

September 18, 2019

We discovered a series of incidents where the credit card skimming attack Magecart was used to hit the booking websites of chain-brand hotels — the second time we've seen a Magecart threat actor directly hit ecommerce service providers instead of going for individual stores or third-party supply chains. Back in May, we discovered a new Magecart-using group called "Mirrorthief," which compromised an ecommerce service provider used by American and Canadian universities.

In early September, we found two hotel websites (from different hotel chains) that were being injected with a JavaScript code to load a remote script on their payment page since August 9. When we first checked the script's link, it downloaded a normal JavaScript code. However, we found that the same link could also download a different script when we requested it from mobile devices like Android or iOS phones. The downloaded script for mobile devices is a credit card skimmer which can steal the information entered on the hotel booking page and send it to a remote server.

We found both of the affected hotel websites were developed by Roomleader, a company from Spain that helps hotels build their online booking websites. The malicious code wasn't injected directly into the website but rather into the script of Roomleader's module called "viewedHotels" that was provided to its clients and subsequently used for two websites of two different hotel chains. Despite the seemingly small number of affected sites, we still consider the attack significant given that one of the brands has 107 hotels in 14 countries while the other has 73 hotels in 14 countries. Note that we have reached out to Roomleader regarding this issue.

The script injected into the hotel booking website

Figure 1. Infection chain of the Magecart skimming attack on the online hotel booking websites

Figure 1. Infection chain of the Magecart skimming attack on the online hotel booking websites

As mentioned, the injection was done on a JavaScript library of Roomleader's "viewedHotels" module located at *hxxps://[hotel website]/modulos/viewedHotels/templates/public/js/history_setter[.]js*. This library is used for saving the viewed hotel information in the visitor's browser cookies. The attacker injected the malicious code in the middle of the original script.

The injected code first checks if an HTML element containing the ID "customerBookingForm" is present on the webpage to make sure it is running on the hotel's booking page. If the injected code is found to not be running on the page, it will go to sleep for one second and check repeatedly thereafter. However, if the code detects the booking page, it will check if the browser debugger is closed and then load another JavaScript from the URL *hxxps://googletrackmanager[.]com/gtm[.]js* — which is where the card skimmer code is actually located. It's worth noting that the style of the URL is meant to emulate the legitimate URL used by Google Tag Manager.

Figure 2. The injected script (highlighted) in the JavaScript library used by hotel websites

Figure 2. The injected script (highlighted) in the JavaScript library used by hotel websites

Analysis of the credit card skimmer

When we first connected to the skimmer URL, it returned normal JavaScript code copied from the GitHub project detect-mobile-browser. However, we suspected it was not the real payload because the code isn't actually used by the affected websites.

Upon further testing of the URL, we found that it downloaded a different script when we made a request using an HTTP User-Agent from a mobile device. This script turned out to be a credit card skimmer. Although we found the skimmer to work on both PC and mobile browsers, it seems the attacker only targeted mobile users. This is most likely because the threat actor behind it wants to avoid detection from PC-based security software. The skimmer is not a new one — we've seen instances where it was used by other groups. Most likely, it is a general skimmer that is shared via underground forums.

Figure 3. The different scripts downloaded from the skimmer URL for desktop and mobile

Figure 3. The different scripts downloaded from the skimmer URL for desktop and mobile

The credit card skimmer is designed to steal data from payment forms. The skimmer hooks its function to the JavaScript events "submit" and "click," which are usually triggered when people submit a payment or a booking. When the hooked event is triggered, the skimmer will check if the browser debugger is closed. Then it copies the name and value from any "input" or "select" HTML element on the booking page. In this case, the gathered information includes names, email addresses, telephone numbers, hotel room preferences, and credit card details.

The copied information is encrypted using RC4 with a hardcoded key: "F8C5Pe4Q". Next, the skimmer will generate a random string to encode the encrypted data again using XOR. The data will then be sent via HTTP POST to the remote URL "https://googletrackmanager[.]com/gtm.php?id=" that uses generated random string appended at the end. Upon receipt of the information, the attacker can then decrypt the data and collect the credit card information.

Figure 4. Credit card skimmer code to steal information from hotel booking page

Figure 4. Credit card skimmer code to steal information from hotel booking page

Magecart replaces the original booking page with a fake one

Although the skimmer itself is not unique, we found that it removes the original credit card form on the booking page and injects another one prepared by the threat actor. We theorize two possible reasons for this. The first is that some hotels don't ask customers to make online payments but instead ask them to pay at the hotel upon arrival. In cases like this, the booking form will ask for credit card information but without the CVC number. To ensure that all credit card information are captured, the attacker replaces the original form with one that contains the CVC number column.

The second possible reason is that, sometimes, the booking page will host the credit card form in a different domain using an HTML iframe element to make it more secure. In this scenario, a regular JavaScript skimmer will not be able to copy the data inside the secure iframe. Therefore, the attacker removes the iframe of the secured credit card form and injects his own form so the skimmer can copy the information.

Figure 5. The original credit card form (above) from the hotel website and the injected form (below) from the skimmer

Figure 5. The original credit card form (above) from the hotel website and the injected form (below) from the skimmer

Figure 5. The original credit card form (above) from the hotel website and the injected form (below) from the skimmer

Figure 6. The skimmer script used to remove the original form from the booking page and replace it with the fake one

Figure 6. The skimmer script used to remove the original form from the booking page and replace it with the fake one

To make it seem more legitimate, the attacker also prepared credit card forms in eight languages: English, Spanish, Italian, French, German, Portuguese, Russian, and Dutch. These languages match the languages supported by the targeted hotel websites. The skimmer will check which language the customer is using for the website and inject the corresponding fake credit card form into the page.

Figure 7. The eight languages of the fake credit card form inside the skimmer

Figure 7. The eight languages of the fake credit card form inside the skimmer

We were unable to find any strong connections to previous Magecart groups based on the network infrastructure or the malicious code used in this attack. However, it’s possible that the threat actor behind this campaign was also involved in previous campaigns.

Conclusion

Recent incidents involving credit card skimmers like Magecart emphasize the need for businesses to secure their websites from potential compromise by implementing security best practices, which include regularly updating software to the latest versions and segregating networks to ensure that as little customer data as possible is exposed.

Furthermore, users can consider using payment systems such as Apple Pay and Google Pay, which offer additional authentication methods — minimizing the chance that attackers will be able to use the credit card even if they manage to collect the card’s details. The following Trend Micro solutions protect users and businesses by blocking the scripts and preventing access to the malicious domains:

- Trend Micro™ Security
- Smart Protection Suites and Worry-Free™ Business Security
- Trend Micro Network Defense
- Hybrid Cloud Security

Indicators of Compromise (IoCs)

SHA-256 Hash/ URL	File Name	Details	Detection name
ac58602d149305bd2331d555c15e6292bd5d09c34ade9e5eebb81e9ef1e7b312	*gtm.js*	Credit card skimmer	TrojanSpy.JS.MAGECART.B
googletrackmanager[.]com		Magecart Domain	

With special thanks to our colleagues at abuse.ch and The Shadowserver Foundation for helping to take down the Magecart domain.

Cyber Threats

We discovered a series of incidents where the credit card skimming attack was used to hit the booking websites of chain-brand hotels — the second time we’ve seen a Magecart threat actor directly hit ecommerce service providers.

By: Joseph C Chen September 18, 2019 Read time: (words)

Content added to Folio