PROVISIONAL APPLICATION FOR PATENT

INVENTION TITLE

An environment to protect web pages and JavaScript code against unauthorized changes without the need to modify the workflow for the content creators or the content users.

BACKGROUND OF THE INVENTION

Problem Solved: Modern web environments provide services that often contain confidential data. While encrypted (SSL/TLS) transport mechanisms protect the data in transit there is no way to protect the source code of the web pages (scripts or other contents) itself. Malicious attackers can modify the source codes on the hosting servers or through other means without the user noticing. This allows attackers to exploit the users.

Traditional models (only supported in Mozilla Firefox) will not protect the authenticity and integrity of the content, if a host is fully compromised. Additionally, it requires special containers and third party signatures to work.

Our security model provides integrity and authenticity under all circumstances - even if the hosting server has been fully compromised. There is no need for special containers or third party elements or any modification to the web content as it is.

DETAILED DESCRIPTION OF THE INVENTION

As stated above, modern web environments provide services that often contain confidential data. While encrypted (SSL/TLS) transport mechanisms protect the data in transit there is no way to protect the source code of the web pages (scripts or other contents) itself. Malicious attackers can modify the source codes on the hosting servers or through other means without the user noticing. This allows attackers to exploit the users. The invention claimed here solves this problem.

We are adding a piece of software that allows individual or all elements of the web site to be only loaded and executed, if digitally signed verification elements are matching thus preventing any modified content to be loaded and/or executed.

The claimed invention differs from what currently exists. Our method is applicable to all browsers that support extensions or plugins. It protects the authenticity and

integrity of web site content even if the hosting server has been fully compromised. It does not require third party signatures, special files or any modification to the web site as it is.

This invention is an improvement on what currently exists. Our method is applicable to all browsers that support extensions or plugins. It protects the authenticity and integrity of web site content even if the hosting server has been fully compromised. It does not require third party signatures, special files or any modification to the web site as it is.

A security implementation is as good as its weakest link. Because the traditional (Firefox only) model does not offer protection if its environment is compromised, it cannot be considered to be safe. Additionally, the cumbersome need to produce special containers and third party certificates disrupt the work-flow.

Our security model provides integrity and authenticity under all circumstances - even if the hosting server has been fully compromised. There is no need for special containers or third party elements or any modification to the web content as it is.

The Version of The Invention Discussed Here Includes:

1. A computer, a monitor, keyboard and mouse
2. An operating system capable of running a modern web browser
3. Internet access
4. A modern web-browser capable of loading and executing extensions (Mozilla Firefox, Google Chrome...)
5. Our software extension
6. Access to a web site that delivers the required content

Relationship Between The Components:

All computer peripherals (1) or other manually manipulable interface for controlling onscreen cursor activity, along with any necessary peripherals to be able to boot the operating system(2), to start the web browser(4) and connect to the Internet(3). Once the software extension (5) has been loaded into the browser, it activates itself if a connection is made to an appropriate web site (6).

How The Invention Works:

Whenever the browser is started, the invented software (embedded within the users browser) downloads a list of advertised web sites (which themselves are protected

by the same method) and thus has a list of urls and matching public RSA keys of content to be protected. The program now monitors all web traffic without interfering. Once it recognizes a protected website (based on its URL), it intercepts all of the web sites content while it is being delivered to the web site. It calculates the hashes of said elements and compares it to the list as provided by the content creator. An icon will be displayed in the web browsers tool bar informing the user of the software's activities and status. As long as the signatures and hash-codes are matching, the software will not do anything. The following conditions will cause an alarm:

- There is no script with appropriate hash codes;
- The RSA signature is not available;
- The RSA signature doesn't match;
- The hash-codes of one or more elements do not match;
- The hash code of an element is missing;
- Other elements (like Content-Security-Policy headers) are not available;

Once an alarm has been triggered, the extension will clear the browsers (DOM-) memory and disable JavaScript for this page (and browser tab) thus preventing any malicious code to be executed. It also informs the user by displaying an appropriate message and it will signal the security violation to the advertising service which will send an email to the content-provider.

Depending on the type of browser used (4), a different extension has to be loaded.

How To Make The Invention:

To make this invention, one has to understand the underlying principles of web design, data delivery, encryption and has to able to implement the technologies to intercept data traffic within a browser as well as to create the necessary encryption, hash-code and digital signing mechanisms.

All elements are necessary. The inventions will be able to adapt to challenges in the encryption and/or signing methods used by implementing different or more advanced hashing algorithms and/or asynchronous encryption mechanisms.

By using different advertising services, different signing, encryption or hashing mechanisms, different ways to provide the hash-codes of the protected contents or by using alternative ways to intercept and to verify the protected elements.

How To Use The Invention:

The inventions is beneficial to two parties: The content creator can rely on the fact that his content will only be accessible if unmodified and the web site visitor (user) can rely on the fact that his data only flows as designed by the content creator.

In order to use the protection of the invented software, the content creator of a web site advertises his wish to use the protection of the invented software by generating an RSA key pair of his choosing and by publishing the RSA public key, in combination with the url of his web site and a notification email address through our service or other means. This step doesn't need to be repeated.

He generates a file containing a JSON string with the names and sha-256 hash codes of all the web sites elements. He signs this created file with the published matching RSA private key. He adds this generated file and its signature to the list of scripts to be loaded. This step has to be repeated whenever the content of this web site changes.

The user only has to download and to install the invented software as an extension into his web browser. His access to protect web sites will now be monitored and verified.

Additionally: The invented software can be used to protect and/or to verify all elements of downloadable content if this content is static (e.g. it does not change frequently).

ABSTRACT

An environment to protect web pages and JavaScript code against unauthorized changes without the need to modify the workflow for the content creators or the content users is disclosed. Our security model provides integrity and authenticity under all circumstances - even if the hosting server has been fully compromised. There is no need for special containers or third party elements or any modification to the web content as it is.