

Phishing by data URI

Henning Klevjer
henning@klevjers.com

August 29, 2012

1 Abstract

Historically, phishing web pages have been hosted by web servers that are either compromised or owned by the attacker. This paper introduces a new approach to creating working phishing web pages without the direct need of a host. The contents of the phishing web page is simply contained its own URI (link). We present the appropriate steps to do this, and show a working example of such a phishing page.

2 Introduction

Using the data URI scheme it is possible to present media content in a web browser without hosting the actual data on the internet. Data URIs follow this scheme:

```
data:[<mediatype>] [;base64] ,<data>
```

Here, `<mediatype>` are one of the MIME media types described in RFC 2046[1]. The MIME media types were originally intended for use with emailing, but are also used to describe all content on the Internet as well. This means that you can represent any content type (e.g. `image/jpeg`, `text/html`, etc.) from the specification that is supported by the web browser.

Base64 encoding is optional. Using it ensures that any representation of data can be correctly transferred over the internet, by using a manageable alphabet to represent the data rather than raw bytes. Base64 splits the data into pieces of six bits (yielding $2^6 = 64$) different characters to choose from.

To exemplify,

```
data:text/plain;,hello
```

shows the text *hello* without the use of Base64 encoding, and

```
data:text/plain;base64,aGVsbG8=
```

shows the same *hello*, and the `data` field now encoded.

Data URI technology has been available as RFC 2397[2] since 1998 – described as a way to easily embed text, pictures and other data in HTML pages, and for such it may be more efficient and hassle-free than hosting a possibly large number of small files.

However, with the ability to host arbitrary data within a URI, the possibility of doing the same with malicious web content springs to mind. Phishing web pages are minimally modified copies of original web pages, usually hosted at a compromised or malicious web server. Creating a phishing site from PayPal, Inc., for example, usually implies hosting at least a copy of PayPal’s login site, credit card information site, or other web page dealing with sensitive data. All content on the web page can be linked from PayPal’s own content servers. However, using the data URI scheme to contain the entire web page’s contents is also possible. Pictures, JavaScript, style sheets etc. can either be translated into their own data URI embedded in the file, or be linked from their respective sources. In the last case, all content, such as a JavaScript in the `head` tag, must be referenced in an absolute manner to work, i.e.:

```
<script type="text/javascript" src="./javascript.js" />
```

is changed to

```
<script type="text/javascript" src="http://example.org/javascript.js" />
```

or

```
<script type="text/javascript" src="data:text/javascript;contents of javascript.js" />
```

3 Creating a phishing site

An easy recipe of creating a phishing site is shown below:

1. Download the login web page to be copied.
2. Change all file paths relative to the domain to absolute ones, or convert them to data URIs.
3. Make the desired modifications to the site’s code. One can, for example, transfer user credentials or private data to another location.
4. (Optional) Encode the text contents of the web page with Base64 encoding to obfuscate the data to the victim. Base64 encoding will extend the overall data size by about 33 %.

5. Append the encoded material or the text contents of the web page into a data URI: Everything from `<!DOCTYPE>` (or `<HTML>`) to `</HTML>` must be moved into the `<data>` field above.

After this, you will end up with something along these lines. Note that we have used `text/html` as the MIME type as this is the appropriate way to present HTML pages.

```
data:text/html;base64,DQo8IURPQ1RZUEUgaHRtbCBQ
VUJMSUMgIi0vL1czQy8vRFREIEhUTUwgNC4wMSBUcmFuc2
10aW9uYWwvL0V0IgoKImh0dHA6Ly93d3cudzMub3JnL1RS
L2h0bWw0L2xvb3N1LmR0ZCI+DQo8aHRtbD4NCiAgICA8aG
(...)
bnR3ZWlyLndvYS9Db250ZW50cy9XZWJtZXJ2ZXJSZXNvdX
JjZXMvc3dfYmFzZS5jc3MnIHJlbD0ic3R5bGVzaGVldCIg
dHlwZT0idGV4dC9jc3MiPg0KICAgICA8c3R5bGUgdH
1wZT0idGV4dC9jc3MiPg0KCQkJdGQ
```

The length of the final URI is a consequence of the data hidden within. If the original web page is very large, embedding linked material within may not be viable.

An example is provided below (Appendix A), which because of its size has been appended at the end.

4 Spreading the phishing web page

Remembering that the web page is contained within the URI, “only”¹ the URI must be passed on to a potential victim. Historically, phishing URIs are transferred by email, but in recent years, social media phishing has exploded. The use of URL shortening services has provided an additional layer of uncertainty in abstracting the original URL from the user². (At least) One URL shortening services, TinyURL.com³, also provides the possibility of shortening data URIs into short URLs. *Whether or not TinyURL does this unwittingly is not known.*

5 Applicability and limitations

Being a rather old RFC specification, data URIs are supported by all major contemporary web browsers. A possible problem of this approach is rather

¹The URI could easily reach hundreds of kilobytes

²However, as the user reaches the target URL, it will be shown in the address bar.

³<http://TinyURL.com>

the web browsers' memory management. The address field is simply not created for containing the enormous amount of bytes contained in the data URI.

In Google Chrome in particular, a control for unsafe redirection is implemented, disabling the user direct access to a data URI if that URI is the target of a redirection, such as from a URL shortening service. The user is presented with an alert that "This webpage is not available", together with the entire URI. Appended below is the error code `Error 311 (net::ERR_UNSAFE_REDIRECT): Unknown error.` indicating that the request was denied due to an unsafe state. However, the target URI is still present in the address field, and a push of the enter button successfully renders the web page. Note that Google Chrome does not produce an error when the user clicks directly on the data URI, without the redirection.

Internet Explorer has a limit to data URIs,

6 Legal issues

In addition to the obvious issues with phishing, a discussion is appropriate as to whether a web host that keeps malicious data URIs is liable for hosting the malicious content they represent. In the above scenario, it can be argued that the URL shortening service is the host, as it provides and keeps the actual content.

7 Future Work

We may see more of so called "spear phishing", attempts focused on individuals, as phishing pages now can be created more easily. A personalised phishing web page can be created automatically, based on gathered information, and transmitted to one victim only. There is reason to believe that the data URI scheme can provide other unknown attack vectors, so research on this topic and further scrutiny of the scheme is a prudent choice.

8 Conclusion

In this paper we have introduced a new way of presenting phishing web pages using a rather old, seldom used way to present web content. Using this procedure, there is no clear source of the phishing page and its content, which makes it difficult to trace, monitor the movement or establish the origin of the web page. Also, we conclude that phishing no longer requires

web hosting of the page⁴, so phishing web pages may be more elusive passed around the Internet. They have no established anchor point in the Internet.

There is no way to *shut down* or remove a data URI web page, besides removing all instances of its link.

The example presented in this document contains no harmful code. The example, and anything learned from this document should NEVER be used to perform any malicious activity. We do not with this example try to point out any vulnerability or weakness specific to Wikipedia, which was selected because of its international reputation and simple login page.

References

- [1] N. Freed and N. Borenstein. *Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types*. RFC 2046 (Draft Standard). Updated by RFCs 2646, 3798, 5147. Internet Engineering Task Force, Nov. 1996. URL: <http://www.ietf.org/RFC/RFC2046.txt>.
- [2] L. Masinter. *The “data” URL scheme*. RFC 2397 (Proposed Standard). Internet Engineering Task Force, Aug. 1998. URL: <http://www.ietf.org/RFC/RFC2397.txt>.

Appendix A: An example phishing web page

Below we present a phishing edition of the login and registration page of the English Wikipedia, <http://en.wikipedia.org/>. It is a minimal example, in which some relative links have been corrected. Additionally, the functionality of the “Log in” button has been altered, showing the password entered in the password field to the user. The login screen of the English Wikipedia can be found here:

<http://en.wikipedia.org/w/index.php?title=Special:UserLogin>

⁴While transmission of sensitive data can be handled within the phishing page, receiving and storing the data is not taken into consideration here.

The rendered web page

The screenshot shows the Wikipedia 'Log in / create account' page. At the top, there is a navigation bar with the Wikipedia logo and the text 'WIKIPEDIA The Free Encyclopedia'. Below this, there are several menu categories: 'Navigation' (Main page, Contents, Featured content, Current events, Random article, Donate to Wikipedia), 'Interaction' (Help, About Wikipedia, Community portal, Recent changes, Contact Wikipedia), and 'Toolbox' (Upload file, Special pages). The main content area is titled 'Log in / create account' and contains a 'Log in' section with a 'Don't have an account? Create one.' link, 'Username:' and 'Password:' input fields, a 'Remember me (up to 180 days)' checkbox, and 'Log in' and 'Forgotten your login?' buttons. A 'JavaScript Alert' box is overlaid on the page, displaying the message 'Your password is correct horse battery staple' with an 'OK' button. To the right of the login form, there is a 'Secure your account:' section with a list of security tips. At the bottom of the page, there are links for 'Privacy policy', 'About Wikipedia', and 'Disclaimers'. The browser's address bar shows the URL 'data:text/html;base64,PCFEETONUWVBFIGH0bWwgUjVFcTcEIDcITly9XXM0MvLORURCBYSFRNTCAxIjAgVHJhbnNpdGlvbmFslY9FTiIjImh0dHA6Ly93d3cudzMub3JnL1RSR3hodC...'.

Special page

Log in / create account

Log in

Don't have an account? [Create one.](#)

Username:

Password:

Remember me (up to 180 days)

[Log in](#) [Forgotten your login?](#)

JavaScript Alert

Your password is correct horse battery staple

[OK](#)

Secure your account:

- Consider logging in on the secure server.
- If your password only contains letters or only numbers, please read our article on password strength and consider changing it (in [Special:Preferences](#) after you log in).
- To avoid becoming a victim of phishing, always verify that you are viewing [Wikipedia's login page](#) when logging in. Wikipedia will never ask for any information other than your username, password and e-mail address.
- Do not give out your password to anyone.
- If your account is compromised, it may be permanently blocked unless you can prove you are its rightful owner.
- As a safeguard you may "commit" to your identity by adding a [cryptographic hash](#) to your user page as explained [here](#). This makes it almost impossible for an impostor to continue impersonating you once you regain control of your account.

[Privacy policy](#) [About Wikipedia](#) [Disclaimers](#)

Powered by [MediaWiki](#)

Base64 encoded data URI

This data URI consists of 24682 characters and can with ease be shrunk to 26 characters with a supported URL shortening service, such as the one mentioned.

