Defeating pharming attacks at the client-side

Sophie Gastellier-Prevost, Maryline Laurent

Defeating pharming attacks at the client-side

Sophie Gastellier-Prevost and Maryline Laurent
CNRS Samovar UMR 5157, Institut Telecom, Telecom SudParis
Evry, FRANCE
Email: {Sophie.Gastellier, Maryline.Laurent}@it-sudparis.eu

***Abstract*—With the deployment of *"always-connected"* broadband Internet access, personal networks are a privileged target for attackers and DNS-based corruption. Pharming attacks - an enhanced version of phishing attacks - aim to steal users' credentials by redirecting them to a fraudulent login website, using DNS-based techniques that make the attack imperceptible to the end-user. In this paper, we define an advanced approach to alert the end-user in case of pharming attacks at the client-side. With a success rate over 95%, we validate a solution that can help differentiating legitimate from fraudulent login websites, based on a dual-step analysis (IP address check and webpage content comparison) performed using multiple DNS servers information.**

I. Introduction

Most of the end-users trust the legitimacy of a login website by looking at the visual aspect of the webpage displayed by the web browser, with no consideration for the visited URL, passive alerts [10] or the presence and positionning of security components, such as the usual security padlock displayed in case of HTTPS connections [14]. Attackers capitalize on this weakness and design nearperfect copies of legitimate websites, in order to perform phishing attacks. By spoofing the identity of a company that proposes financial services, phishing attacks steal confidential information (e.g. login, password, credit card number) to the Internet users.

However, if the end-user was carefully watching the visited URL, most of phishing attacks could be easily detected. In more sophisticated versions of phishing attacks that exploit DNS vulnerabilities - ie. pharming attacks -, the threat is imperceptible to the user: the visited URL is the legitimate one and the visual aspect of the fake website is very similar to the original one.

Despite efforts to secure DNS protocol, protecting the end-user network against DNS corruption remains difficult, as described by Stamm et al. [19]. DNSSEC extension was designed to secure DNS exchanges, but it does not solve attacks targeting local lookup settings (see Section II).

In a previous paper [11], we introduced a dual approach to provide an anti-pharming protection integrated into the client's browser. In this paper, we propose an advanced approach - that combines both an IP address check and a webpage content analysis, using the information provided by multiple DNS servers - with substantial results that demonstrate its accuracy and effectiveness to detect pharming attacks at the client-side.

This paper is organized as follows: Section 2 discusses client-side attacks and related works. Section 3 introduces our dual-step framework. Section 4 describes test-bed conditions and section 5 details experimental results. Then, section 6 discusses the proposed framework and section 7 gives conclusion and perspectives.

II. Pharming attacks and related works

For the last ten years, the proliferation of fake websites lead researchers to propose many approaches for counteracting identity theft based attacks. Most of these approaches focused either on phishing attacks - by providing multiple detection techniques such as blacklists, heuristics, authentication schemes, etc. - or on DNS-based attacks performed in the ISP network or at the server-side.

In this section, we discuss pharming attacks performed at the client-side as well as related works.

Pharming attacks exploit DNS vulnerabilities to defeat the integrity of the lookup process for a domain name. Many types of DNS-based attacks have been already identified [17]. In this paper, we focus on DNS attacks that are performed at the client-side to modify the local lookup settings.

We can distinguish the following attacks:

- **Local host attack** statically modifies the victim's operating system host files to redirect the user's traffic to a domain under the attacker's control.
- **Browser proxy configuration attack** overrides the victims' web browser proxy configuration options, using DNS spoofing or poisoning techniques, to redirect all the web traffic to a fraudulent proxy server that is under the attacker's control. Another type of browser attack - DNS rebinding attack - tends to convert the user's web browser into an open network proxy [13], e.g. the client's browser can visit a malicious website that embeds a Flash movie which opens a socket to an arbitrary port number rebounded by the attacker. As a result, the attacker is enabled to read arbitrary documents, compromise internal machines, hijack some IP addresses, etc.
- **Rogue DHCP**. The attacker uses malicious softwares, by installing a rogue DHCP on the client's network, to control the DHCP local options. The objective is to modify the DNS server of the user to provide incorrect host resolutions.
- **Home or border router attack** aims to access and compromise the home or border router so that, by adding or modifying DNS entries, the user's traffic is redirected to the attacker's server. Stamm et al. [19] describes several attacks scenarios to compromise home routers.

As far as we know, the closest relevant paper to our works was recently published by Bin et al. [7]. They focused on a DNS-based approach to detect whether a credit card number is sent to a suspicious website. As such, their system is based on a database of banks names, associated registered IP adresses and issued card number ranges. Each time the end-user enters a credit card number, the system sends an inverse DNS query to check whether the visited webpage is related to the expected bank. This approach needs both maintaining and efficiently protecting the bank database, as well as providing a low latency alert. Because the detection system is based on recognizing a card number range, it means the end-user already started to enter it on the suspicious website. This approach looks similar to our proposal as DNS queries are sent to verify the legitimacy of websites, but it focuses on phishing attacks and protection of credit card numbers only, while our approach aims at protecting any types of end-user credentials against pharming attacks.

III. Framework description

Our framework aims at detecting fraudulent login websites at the client-side, and displaying active and passive notifications to the end-user in case a pharming attack is suspected.

Our proposal intends to be integrated into the web browsers (see Figure 1) using two components [11] :

- An active warning alert, displayed as a pop-up message, that requires an action of the end-user in case a DNS compromise is detected at the client-side.
- A visual indicator which is integrated in the address bar of the web browser, to notify the current trust level to the end-user.

The core idea of our framework (see Figure 2) is to detect pharming attacks, thanks to multiple DNS servers responses, by performing a dual-step analysis composed of:

- An IP address check of the visited domain.
- A webpage content comparison of the displayed webpage against a reference webpage.

A. IP address check

Each time the web browser accesses a URL, the domain name of the visited website is checked out. Then, a DNS request is sent to two DNS servers - the default one and a reference one -, in order to compare the IP addresses returned for the evaluated domain name. The default DNS server returns the IP address of the site displayed in the web browser (further named "default IP address" or IPdef), and the reference DNS server can return one or several IP addresses (further named "reference IP addresses" or IPref), including or excluding the one used by the browser.

If the default IP address is included in the reference IP addresses, the site is considered as legitimate. Otherwise, the webpage content analysis is performed (see Figure 2).

Reference server definition : Defining the reference server is a critical issue as it participates to the core decision-making process. Two approaches may be considered:

- The definition of the reference server is left to the discretion of the end-user through an interface embedded in his web browser. Of course, the end-user might select a reference server different from his ISP.
- A set of reference servers are pre-defined and embedded in the framework. Each time the legitimacy of a website has to be checked, a reference server is randomly selected among the pre-defined list of DNS servers (e.g. OpenDNS, GoogleDNS, etc.). Our tests results demonstrate that the three reference servers used in our experimentations gave similar results (see section V).

Even if the first approach lets more flexibility to the end-user, it also implies a more static and vulnerable configuration. As such, we believe that the second approach is the strongest one as long as it does not overlap the default DNS configuration of the end-user.

B. HTML source-code analysis

In the second step of our approach, we analyze the HTML source code of the visited webpage. To avoid the main drawback of previous approaches that need to maintain an up-to-date database [15] [18] [12], our framework aims to compare webpages on-the-fly, without any storage of their content.

The source code of the two following webpages is downloaded as follows:

- The "visited webpage", returned by the IPdef address, is the usual webpage downloaded by the web browser when asking for a URL.
- The "reference webpage" is obtained by targeting the GET HTTP request to IPref. If the reference DNS answer returns only one IP address, we use it as IPref. If multiple IP addresses are returned by the reference DNS server, IPref is determined by comparing the answers of the two DNS servers and choosing the first IP address different from IPdef.

Next, the contents of the visited and reference webpages are compared (see Figure 2).

For instance, the web browser of the end-user displays the following login webpage: https://twitter.com/ using the IP address 199.59.148.83 (IPdef) returned by the default DNS server. For the domain name twitter.com, the reference DNS server returns three IP addresses: 199.59.148.11, 199.59.148.10 and 199.59.148.83. Based on the comparison of the two DNS answers, IPref is selected as 199.59.148.11. Then, the reference webpage is downloaded using the original URL (i.e. https://twitter.com/, sent to the IP address 199.59.148.11.

As underlined in our previous works [11], analyzing webpages for content comparison introduces many difficulties due to dynamic contents, difference of structures, etc. Then, our webpage comparison focused on the HTML source-code analysis and compares two approaches:

Character approach (based on N-gram approach [9]): A score is calculated for each webpage (p) depending on the occurrence frequency (occ) of each character (i), as follows:

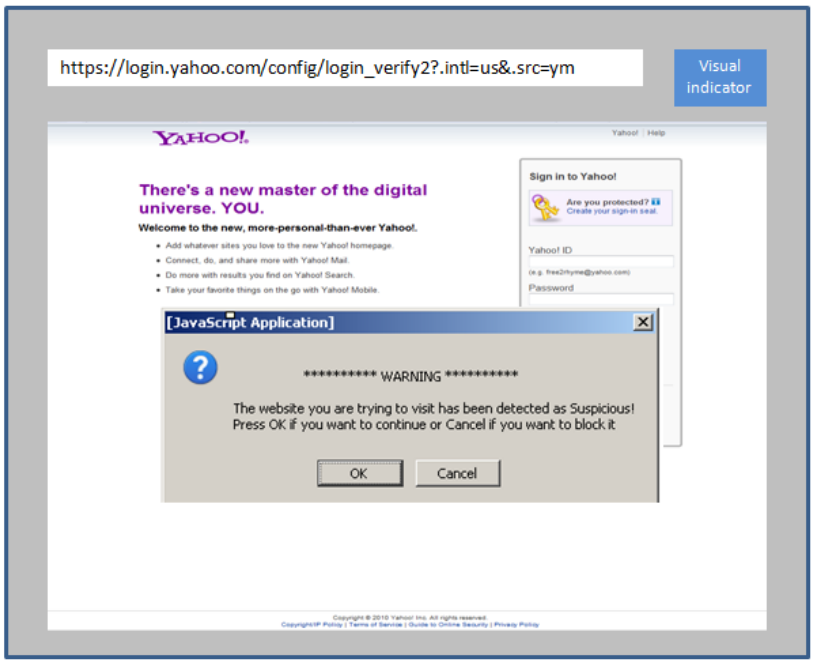

Fig. 1. Framework integration into the web browser

$$Score(p) = \sum occ(i) \times valAscii(i)$$
$$where\ valAscii(i)\ represents\ the\ Ascii\ value\ of\ i.$$

The percentage of similarity between the two webpages is determined by comparing their score. A percentage higher or equal to the "decisional threshold" leads to consider the visited webpage as legitimate, as displayed by the visual indicator. A percentage of similarity lower than the "decisional threshold" leads to assess the visited webpage as suspicious, and both the visual indicator and the pop-up window display alerts to the end-user (see Figure 2).

Word approach (based on Diff approach [16], [20]): Each webpage is split into words and the resulting files are compared using Diff approach. The percentage of similarity between the two webpages is obtained by comparing the words contained in each file (i.e. it determines how many words are unchanged, deleted, added or modified and calculates a resulting score) and their associated location in the HTML document.

Then, the percentage of similarity is compared to the "decisional threshold" - as described in the character approach - to determine the legitimacy of the visited website.

IV. Test-bed conditions

To evaluate the effectiveness of the proposed framework and define decisional threshold, we performed two sets of experimentations: A) Comparison between legitimate sites retrieved using multiple DNS servers (performed at different locations), and B) Comparison between legitimate and fraudulent websites that look very similar.

A. Legitimate sites

To compare legitimate websites retrieved using multiple DNS servers information, we performed two sets of experimentations by selecting up to 328 legitimate login sites, tested from up to 11 different locations over 5 continents, from December 2010 to April 2011:

- The first set of experimentations tested 108 login websites from 11 locations over 5 continents, from December 2010 to January 2011.
- The second set of experimentations tested 328 login websites from 10 locations over 5 continents, from March to April 2011.

URL selection: The HTTPS URLs selected to compare legitimate sites are retrieved from different business sectors, different locations in the world, developed in different languages and using different TLDs. We classified the selected URLs based on two characteristics:

- *The business sector* is divided into five categories: banks (most of bank websites were selected using Levoyageur website [4]), social networks, e-commerce, email and others. *Others* category includes login sites from administration, insurances, online support for softwares, online games, industry (e.g. cars, solar panels, etc.), universities, video-sharing, photo-sharing or news.
  For example, the 328 legitimate login URLs used for the second set of experimentations is mainly composed of banks (62.20%), followed by others (21.65%), e-commerce (13.41%), social networks (1.52%) and (1.22%).

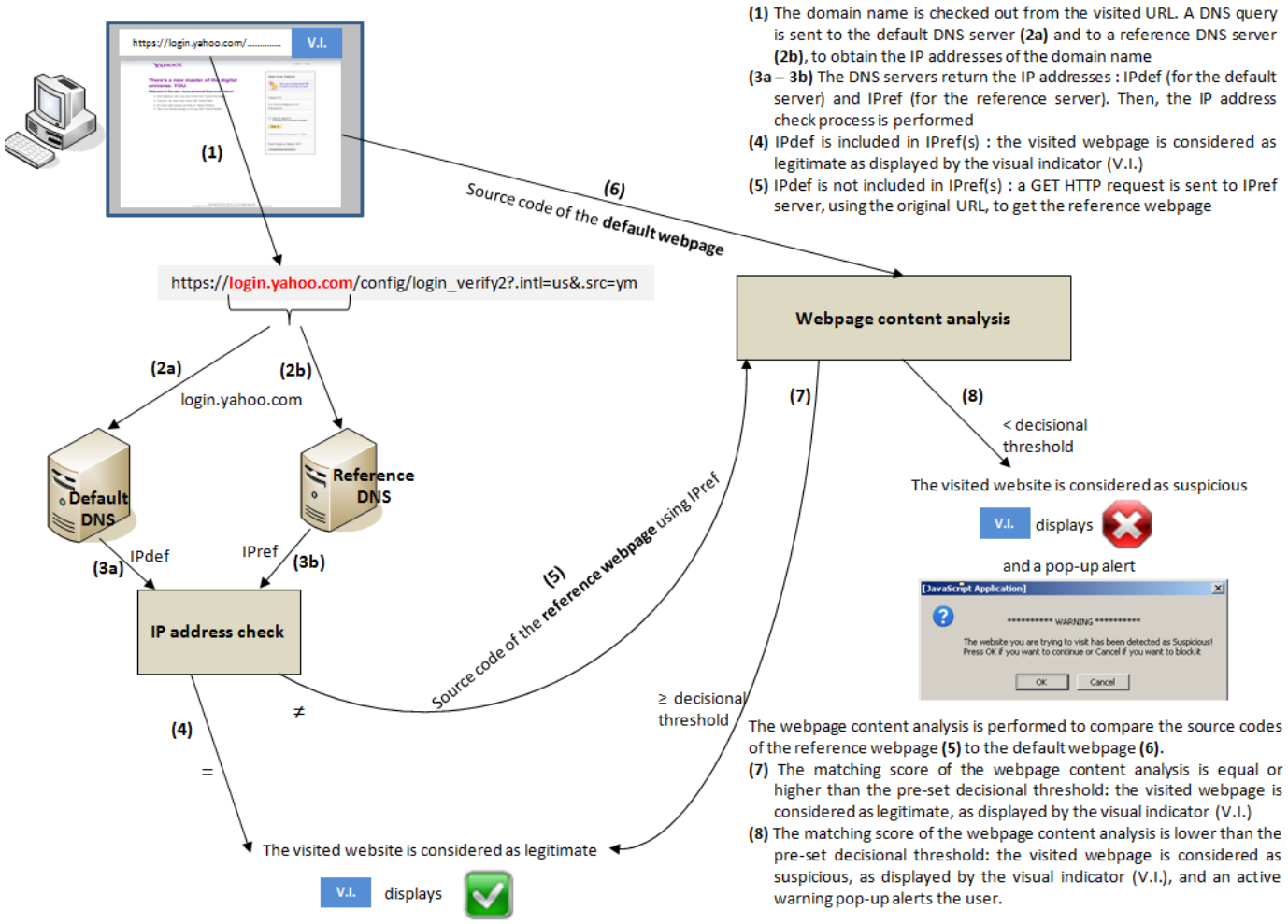

Fig. 2. Framework description

- *The TLD* is divided into two categories: cc-TLD (Country-Code Top-Level Domain) and g-TLD (Generic Top-Level Domain).
  For example, in the second set of experimentations, the URLs with cc-TLD are distributed as follows: Europe (38.11%), Asia (5.79%), Australia (4.27%), South America (3.35%) and North America (0.61%). For the URLs with g-TLD, the distribution is as follows: Commercial (44.82%), Network (1.22%), Organization (0.61%) and Cooperative (0.30%).

DNS servers: The two sets of experimentations - over 108 and 328 login sites - were performed using four DNS servers. For each location, three public reference DNS servers were used: OpenDNS [5], GoogleDNS [3] and DNSAdvantage [2]. In addition, the default DNS server proposed by the ISP - verified as different from the reference servers - was used as the default DNS server.

B. Fraudulent sites

To compare legitimate and fraudulent websites - in order to determine a ”decisional threshold” that helps to detect DNS corruption at the client-side -, we selected phishing sites that looks very similar to legitimate ones. As such, we used 76 phishing sites reported as valid phising sites by Phishtank [6] and APWG [1] websites from January to May 2011. We selected exclusively phishing sites that look like nearperfect copies of the legitimate sites (see example in Table I). For each phishing site we selected - later used as the default webpage -, we retrieved and stored the associated legitimate webpage and used it as the reference webpage. We made effort selecting phishing sites from as many business sectors as possible: banks (e.g. Lloyds, Bank of America, BMO, BBVA, Chase, Natwest, HSBC, Paypal), social networks (e.g. Facebook), online games (e.g. Runescape, Battle.net), e-commerce (e.g. eBay) and e-mail (e.g. Hotmail).

TABLE I
SCREEN CAPTURES OF HOTMAIL LEGITIMATE SITE (HTTPS://LOGIN.LIVE.COM/...)
AND FRAUDULENT COPY (HTTP://LLHOTMAILL.WEBCINDARIO.COM/LOGIN.SRF.PHP), RETRIEVED ON MARCH 22TH, 2011

[Legitimate site]

[Fraudulent site]

V. EXPERIMENTAL RESULTS

A. IP address check

Our first set of experimentations - over 108 legitimate login websites - introduced the stability of the IP addresses used by login websites (see Table II), as previously identified by Cao et al. [8]. These tests, performed from 11 different locations over 5 continents, demonstrated that the IP adresses returned by the reference DNS servers match - partially or fully - the IP addresses returned by the default DNS server of the end-user for 81.22% to 82.90% of the requested domain names.

We confirmed this trend with the second set of experimentations over 328 legitimate login sites (see Table II) with higher matching rates (from 86.33% to 87.90%) between the IP addresses returned by the default and reference DNS servers. This confirms the IP address check as a significant indicator of the legitimacy of a visited login website.

B. HTML source-code analysis

Our webpage comparison focuses on the HTML source-code analysis and considers two approaches:

Character approach: We implemented this approach using Java language and we tested it over legitimate and fraudulent sites.

- *Comparison of legitimate sites* (see Table III): Using the character approach, our first set of experimentations - over 108 login sites - indicates that the default login webpages matches from 98.63% to 98.71% (average values) the reference login webpages, with a low standard deviation (from 0.26% to 0.32%).
  The second set of experimentations - over 328 login sites - improves these results: the default login webpages matches from 99.83% to 99.85% (average values) the reference login webpages, with a lower standard deviation (from 0.06% to 0.07%).

- *Comparison of legitimate and fraudulent sites*: Our set of experimentations - over 76 couples of legitimate/fraudulent websites - gives a matching score fluctuating from 10.83% to 99.93% between the two compared webpages, with an average similarity level of 76.68% and a standard deviation of 27.01%.

Word approach: This approach, implemented using Java language, was also tested over legitimate and fraudulent sites.

- *Comparison of legitimate sites* (see Table III): Using the word approach, our first set of experimentations - over 108 login sites - indicates that the default login webpages matches from 91.38% to 91.56% (average values) the reference login webpages, with a standard deviation from 3.13% to 3.46%.
  The second set of experimentations - over 328 login sites - improves these results: the default login webpages matches from 96.95% to 97.38% (average values) the reference webpages, with a lower standard deviation (from 0.26% to 0.81%).

- *Comparison of legitimate and fraudulent sites*: Our set of experimentation - over 76 couples of legitimate/fraudulent websites - gives a matching score fluctuating from 2% to 97% between the two compared webpages, with an average similarity level of 60.38% and a standard deviation of 27.41%.

The first set of experimentations - over 108 login websites - indicates that the character approach gives the best results for comparison between legitimate webpages (98.63% to 98.71% of average similarity), as the word approach gives also a high matching score (91.38% to 91.56% of average similarity).

We confirm this trend with the second set of experimentations (over 328 login websites): the character approach

TABLE II
DNS QUERY RESULTS

		Matching rate with IP addresses returned by the default DNS server (min $\leq$ average $\leq$ max)	Standard deviation
OpenDNS	108 login sites	76.47% $\leq$ **82.90%** $\leq$ 95.19%	5.00%
	328 login sites	82.62% $\leq$ **86.33%** $\leq$ 92.66%	2.57%
GoogleDNS	108 login sites	76.92% $\leq$ **81.61%** $\leq$ 91.35%	3.62%
	328 login sites	85.80% $\leq$ **87.90%** $\leq$ 93.60%	2.57%
DNSAdvantage	108 login sites	74.51% $\leq$ **81.22%** $\leq$ 89.42%	4.10%
	328 login sites	84.36% $\leq$ **86.47%** $\leq$ 87.50%	1.10%

TABLE III
COMPARISON BETWEEN LEGITIMATE WEBPAGES USING CHARACTER AND WORD APPROACHES

		CHARACTER APPROACH		**WORD APPROACH**	
		Matching score with the default webpage (min $\leq$ average $\leq$ max)	Standard deviation	**Matching score** with the default webpage (min $\leq$ average $\leq$ max)	Standard deviation
OpenDNS	108 login sites	98.14% $\leq$ **98.63%** $\leq$ 98.96%	0.26%	81.96% $\leq$ **91.56%** $\leq$ 93.57%	3.24%
	328 login sites	99.67% $\leq$ **99.83%** $\leq$ 99.91%	0.07%	95.24% $\leq$ **96.95%** $\leq$ 97.58%	0.81%
GoogleDNS	108 login sites	98.14% $\leq$ **98.68%** $\leq$ 98.96%	0.27%	81.61% $\leq$ **91.50%** $\leq$ 94.53%	3.46%
	328 login sites	99.70% $\leq$ **99.84%** $\leq$ 99.89%	0.06%	95.24% $\leq$ **96.95%** $\leq$ 97.58%	0.81%
DNSAdvantage	108 login sites	98.22% $\leq$ **98.71%** $\leq$ 99.26%	0.32%	82.20% $\leq$ **91.38%** $\leq$ 92.76%	3.13%
	328 login sites	99.73% $\leq$ **99.85%** $\leq$ 99.90%	0.06%	96.72% $\leq$ **97.38%** $\leq$ 97.60%	0.26%

TABLE IV
COMPARISON OF LEGITIMATE VS. FRAUDULENT WEBPAGES USING CHARACTER AND WORD APPROACHES OVER 76 COUPLES OF LEGITIMATE/FRAUDULENT WEBSITES

	Matching score (min $\leq$ average $\leq$ max)	Standard deviation	FPR (False Positive Rate) if decisional threshold set to 95%	FPR (False Positive Rate) if decisional threshold set to 99%	Determining approach for % of couple of sites
CHARACTER APPROACH	10.83% $\leq$ **76.68%** $\leq$ 99.93%	27.01%	23.68%	6.58%	17.11%
WORD APPROACH	2% $\leq$ **60.38%** $\leq$ 97%	27.41%	3.95%	-	82.89%

gives similar high results for comparison between legitimate webpages (99.83% to 99.85% of average similarity), while the word approach improves its matching score (96.95% to 97.38% of average similarity).

When considering the legitimate vs. fraudulent website comparison, the character approach gives a high matching score (average of 76.68%), with a maximum value (99.93%) very similar to the average similarity score obtained with comparison between legitimate webpages. On the other hand, the word approach gives a lower and more interesting matching score (average of 60.38%), even if the maximum value (97%) is very similar to the average similarity score between legitimate webpages.

We also determine the potential false positive rate (FPR). We check how many couples of legitimate/phishing websites can be considered as borderline, from a potential decisional threshold used to differentiate legitimate from fraudulent sites. It appears clearly that the word approach is the most interesting one (see Table IV): only 3.95% of webpage comparison give a matching score higher than 95% and none over 99%, while using the character approach 23.68% of couples of tested websites give a matching score higher than 95%, and 6.58% over 99%.

In addition, our analysis indicates that the word approach is the determining approach (i.e. giving the lowest matching score) for 82.89% of couples of legitimate/fraudulent websites when comparing the full webpage content (see Table IV).

Based on the above results, we go one step further in our analysis by considering the comparison of subparts of the webpages. The objective is to determine the most prevalent parts of the HTML source-code for differentiating legitimate from phishing sites, reducing the FPR (by multiplying decision indicators), and optimizing the processing time.

For example, on-going experimentations tend to indicate that the character approach should be more interesting when applied on smaller or specific parts of the webpages (e.g. links, specific tags). In addition, the word approach applied to the HEAD subpart of the HTML source-code gives promising results: the average matching rate, when comparing legitimate vs. fraudulent webpages, is about 70%, while the minimum matching score when comparing legitimate webpages is over 99%.

Future experimentations will tend to corroborate above results and to conduct a thorough analysis of webpage subparts.

VI. DISCUSSION

To determine the viability and the scalability of the proposed framework, several indicators are carefully examined such as

TABLE V
SUCCESS RATE TO RETRIEVE THE REFERENCE WEBPAGE

	PREVIOUS APPROACH		NEW APPROACH	
	Success rate for downloading webpage (min $\leq$ average $\leq$ max)	Standard deviation	**Success rate** for downloading webpage (min $\leq$ average $\leq$ max)	Standard deviation
OpenDNS	71.30% $\leq$ **77.27%** $\leq$ 79.63%	2.09%	84.15% $\leq$ **95.43%** $\leq$ 98.48%	4.38%
GoogleDNS	68.52% $\leq$ **77.44%** $\leq$ 79.63%	3.11%	84.15% $\leq$ **95.18%** $\leq$ 98.48%	4.28%
DNSAdvantage	75.93% $\leq$ **77.61%** $\leq$ 78.70%	0.81%	84.45% $\leq$ **95.21%** $\leq$ 98.17%	2.57%

the success rate, the processing time and the limitations of our approach.

A. Success rate

Reference webpage: In comparison to our previous approach [11], we significantly improved the success rate of the proposed solution. Previously, the reference webpage was retrieved by replacing the domain name by IPref in the original URL. This leads to substantial failures and lower success rates (from 77.27% to 77.61%) for downloading the reference webpage (see Table V). These failures are due to lack of reverse DNS configuration, the use of load sharing and webservers virtualization.

With our new approach (see Section III), we substantially improve the success rate from 95.18% up to 95.43%. Residual errors are due to some minor certificate exchange issues. Note that, depending on the location (where the tests were experimented), we get sometimes higher error rates due to some connectivity issues (outage or loss of Internet access while processing test experimentations).

Reference DNS servers: We notice that the three reference DNS servers, used for our experimentations, give similar results, both for the IP address check and webpages comparison. This strengthens our decision to better integrate the reference server definition into our framework as explained in section III, in order to minimize potential weaknesses of our approach.

B. Processing time

Depending on the location the tests were performed, the processing time can greatly vary due to outage or loss of Internet connectivity. By limiting our study to the locations with no connectivity problems - i.e. 7 over the 10 locations tested using the second set of experimentations -, we evaluate the average processing time. It takes about 3.6 secondes to retrieve the reference webpage and to calculate the associated scores (using character and word approaches). This prevents the end-user to enter his credentials before any alerts from the framework.

C. Limitations

Webpage content redirection : Our framework compares the HTML source-code of the two (reference and default) webpages without looking for content redirections. For example, a login website can use multiple frames displayed thanks to URL redirections embedded in the source-code. Then, our webpage analysis is limited to compare the embedded links and does not examine the content of the external frames. A potential improvement of the proposed framework could be to analyze all content redirections, but probably at the expense of the processing time.

No authentication : Our webpage content analysis does not aim to authenticate login websites - i.e. it is possible to obtain the same score for two unrelated webpages, especially using the character approach - but it helps defining decisional thresholds to differentiate legitimate from fraudulent websites. However, considering that attackers' goal is to lure as many users as possible with nearperfect copies of legitimate sites, the end-user would easily detect a fraudulent website that does not reproduce the legitimate one.

Browser vulnerability and framework implementation : One limitation of our proposal is related to its location (into the browser) and its implementation. It might be subject to web browser vulnerabilities as well as web browser implementation issues, such as the integration of JavaScript language - to design an appropriate interface for the user - and Java language - to make multiple DNS requests - both at the client-side.

DNS filtering : Another limitation of our proposal is due to specific configurations of the end-user's network connection. In some few cases, DNS queries filtering is enforced, which might defeat the IP address check of our proposal.

VII. CONCLUSION

Our framework - based on a dual-step analysis and collaboration of multiple (default and reference) DNS servers - proposes an anti-pharming protection at the client-side for detecting DNS corruptions. Its implementation into the client's browser can be part of a global solution that combines both protection against phishing and pharming attacks.

In this paper, we demonstrate that the IP address check is a significant indicator of the legitimacy of a visited login website. In addition, the webpage content comparison results indicate that the word approach helps significantly to differentiate legitimate from fraudulent websites for up to 82.89% of the 76 couples of tested webpages. Those results lead to preset the decisional threshold around 95%.

Future experimentations are planned to further improve the webpage content comparison. The objective is to combine multiple approaches over different parts of the HTML source-

code content, to improve the false positive rate and to limit the processing time.

REFERENCES

[1] "APWG - Anti-Phishing working group." [Online]. Available: http://www.apwg.org/

[2] "DNS advantage." [Online]. Available: http://www.dnsadvantage.com/

[3] "Google public DNS." [Online]. Available: http://code.google.com/intl/fr/speed/public-dns/index.html

[4] "Levoyageur - banks in the world." [Online]. Available: http://www.levoyageur.net/banks.php

[5] "OpenDNS." [Online]. Available: http://www.opendns.com/

[6] "PhishTank." [Online]. Available: http://www.phishtank.com/

[7] S. Bin, W. Qiaoyan, and L. Xiaoying, "A DNS based anti-phishing approach," in *Proceedings of the 2010 Second International Conference on Networks Security, Wireless Communications and Trusted Computing*, vol. 02. China, Wuhan: IEEE Computer Society, Apr. 2010, pp. 262–265.

[8] Y. Cao, W. Han, and Y. Le, "Anti-phishing based on automated individual white-list," in *Proceedings of the 4th ACM workshop on Digital identity management*. Alexandria, Viriginia, USA: ACM, Oct. 2008, pp. 51–60.

[9] W. B. Cavnar and J. M. Trenkle, "N-Gram-Based text categorization," in *Proceedings of SDAIR*, 1994.

[10] S. Egelman, L. Cranor, and J. Hong, "You've been warned: an empirical study of the effectiveness of web browser phishing warnings," in *Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*. Florence, Italy: ACM, Apr. 2008, pp. 1065–1074.

[11] S. Gastellier-Prevost, G. G. Granadillo, and M. Laurent, "A dual approach to detect pharming sites at the client-side," in *Proceedings of the 4th IFIP International Conference on New Technologies, Mobility and Security (NTMS)*, Paris, France, Feb. 2011.

[12] M. Hara, A. Yamada, and Y. Miyake, "Visual similarity-based phishing detection without victim site information." Nashville, Tennessee, USA: IEEE, Apr. 2009, pp. 30–36.

[13] C. Jackson, A. Barth, A. Botz, W. Shao, and D. Boneh, "Protecting browsers from DNS rebinding attacks," *ACM*, vol. 3, Issue 1, no. 2, Jan. 2009.

[14] C. Ludl, S. McAllister, E. Kirda, and C. Kruegel, "On the effectiveness of techniques to detect phishing sites," in *Proceedings of the 4th international conference on Detection of Intrusions and Malware, and Vulnerability Assessment*, vol. Lecture Notes In Computer Science; Vol. 4579. Lucerne, Switzerland: Springer-Verlag Berlin, Heidelberg, 2007, pp. 20–39.

[15] E. Medvet, E. Kirda, and C. Kruegel, "Visual-Similarity-Based phishing detection," in *Proceedings of the 4th international conference on Security and privacy in communication networks*. Istanbul, Turkey: ACM, Sep. 2008, p. Article No. 22.

[16] E. W. Myers, "An O(ND) difference algorithm and its variations," *Algorithmica*, pp. 251–266, 1986.

[17] G. Ollman, "The pharming guide," Jul. 2005. [Online]. Available: http://www.ngssoftware.com/papers/ThePharmingGuide.pdf

[18] A. P. E. Rosiello, E. Kirda, C. Kruegel, and F. Ferrandi, "A layout-similarity-based approach for detecting phishing pages." Nice, France: IEEE, Sep. 2007, pp. 454–463.

[19] S. Stamm, Z. Ramzan, and J. Markus, "Drive-By pharming," in *Proceedings of the 9th international conference on Information and communications security*, vol. Network Security. Zhengzhou, China: ACM, 2007, pp. 495–506.

[20] S. Wu, U. Manber, G. Myers, and W. Miller, "An O(NP) sequence comparison algorithm," *Information Processing Letters*, pp. 317–323, Sep. 1990.