A Look at Hidden Web Pages in Italian Public Administrations

Enrico Sorio, Alberto Bartoli, Eric Medvet
DIA - Engineering and Architecture Dept.
University of Trieste, Italy
Email: enrico.sorio@phd.units.it, bartoli.alberto@units.it, emedvet@units.it

***Abstract*—Preventing illegitimate modifications to web sites offering a public service is a fundamental requirement of any e-government initiative. Unfortunately, attacks to web sites resulting in the creation of fraudulent content by hackers are ubiquitous. In this work we attempted to assess the ability of Italian public administrations to be in full control of the respective web sites. We examined several thousands sites, including all local governments and universities, and found that approximately 1.5% of the analyzed sites serves contents that admittedly is not supposed to be there. Although these contents do not constitute an immediate threat to citizens, this result does not seem very encouraging also because our methodology leads to very conservative estimates. We believe that our analysis allows gaining useful insights into this novel and peculiar threat.**

***Keywords*-e-government, web security, web trust and dependability**

I. Introduction

The web is increasingly becoming one of the fundamental means for accessing government and public services. On the other hand, the web is also plagued by ICT-based security incidents at all levels: user devices, browsers, networks, server infrastructure [2]. An effective strategy for tackling Internet security threats is thus essential for fueling innovation and diffusion of e-government initiatives [7]. Indeed, it is not surprising that one of the seven goals of the European Digital Agenda is "enhance trust and security" [1].

A peculiar form of Internet security threats consists in the illegitimate modification of a web site that offers a public service. These risks are web-specific and have no counterpart in the traditional, i.e., "physical", access to public services. A common form of illegitimate modification is the *defacement*, i.e., the replacement of the original page with an entirely different page carrying political messages, offensive content and alike [4], [3]. In this case the user perceives immediately that the service cannot be used, thus the main effect of the attack is on the availability and utility of the application.

Other forms of attacks are aimed at remaining undetected by users and may take different forms:

- Subtle changes aimed at injecting malware in browsers by exploiting software vulnerabilities [10].
- Addition of new pages at URLs at which no page is supposed to exist. These pages are essentially defacements, except that they do not replace the original content and are visible only to users that know their URL. Attacks of this form are meant to be merely a proof of the ability of the hacker. A significant fraction of the defacements archived in Zone-H fall in this category[1].
- Addition of illegitimate content aimed at deliberately manipulating search results—*search spam* [5]. The illegitimate content consist of links to the pages to be fraudulently promoted. The links may be added on existing pages or on pages created explicitly and solely to this purpose, at URLs at which no page is supposed to exist.
- Modification of the site aimed at redirecting the browser to a site chosen by the attacker only when the user comes from a page returned by a search engine—*search redirection*. The relevance and diffusion of these attacks have been illustrated recently in the context of illegal drug trade[8].

Attacks of these forms are hard to detect because site administrators will likely never see any anomaly in the served content. Their effects on web sites of public interest may be very dangerous. Even leaving the malware injection threat aside, careful exploitation of the other forms of attack may create a very odd scenario: pages hosted on a trusted site that serve content fully controlled by attackers, tailored to the navigation path followed by users, visible only to certain users. An analogy with the physical world may illustrate the issue more clearly: when entering into the building of a public administration, one would not expect to find offices that are not supposed to exist and are visible only to certain citizens, perhaps depending on where they come from. Unfortunately, this is exactly what it could happen in web sites of public administrations. It is important to point out that HTTPS—the main and ubiquitous line of defense in sensitive web sites—does *not* provide any defense in this respect. HTTPS ensures secrecy, integrity and authentication by means of cryptographic techniques. The problem is, the server site is authenticated as a whole—any page coming from that site appears as being legitimate.

In this work we attempted to assess the ability of Italian public administration to be in full control of the respective web sites. We examined several thousands sites, including all local governments and universities, in search of evidence of attacks of categories (iii) and (iv)—a quick look at Zone-H will reveal that attacks of category (ii) occur more or less routinely. We defined a methodology based on carefully constructed

[1] http://www.zone-h.org

978-1-4673-4794-5/12/$31.00

search engine queries for identifying pages that could be the results of those attacks, and inspection of those pages for filtering false positives out. We found that approximately 12% of the analyzed pages corresponds to contents that admittedly is not supposed to be there. Although these contents do not constitute an immediate threat to citizens, this result is not very encouraging and somewhat surprising. To place this result in perspective, we observe that a state-of-the-art system recently developed for efficiently searching malicious web pages, manages to construct a stream of URLs in which 1.34% of them identify malicious pages and this system improves earlier strategies by one order of magnitude [6]. While our results *cannot* be compared directly to those of the cited paper (the cited paper focussed on pages that distribute malware, i.e., attacks (i), and we could process a much smaller sample), it seems fair to claim that our results are indeed surprising. Besides, as clarified in the next sections, we could inspect just a few of the possible attack signatures, hence our data are very conservative. We believe that our analysis allows gaining useful insights into this novel and peculiar threat.

II. Our methodology

A. Preliminary Observations

Performing a full crawl of all the web sites of interest is clearly not feasible, even leaving aside the problem of discriminating between legitimate and illegitimate content. Moreover, a full crawl would not highlight search redirection attacks: in these attacks the fraudulent code on a compromised server identifies the requests to be redirected based on the HTTP request header, whose value identifies the page containing the link followed by the user—we would have to repeat the full crawl with differing values for such header, one indicating a click on a Google result page, another indicating an URL typed directly and so on.

For these reasons, we defined a methodology that may only search for a predefined set of attack signatures but can be implemented with moderate effort. As described in full detail in the next sections, the basic idea consists in querying search engines for the presence of certain words in the target sites. These words are chosen so as to be unlikely to be found in legitimate pages at those sites. The results provided by the search engines are then analyzed carefully so as to filter false positives out, i.e., to identify pages that are indeed fraudulent. Of course, this methodology cannot provide a full coverage of intrusions—querying search engines for *all* words or sentences that "should not be found" in legitimate pages is not feasible. Consequently, our analysis can only provide a partial view and conservative estimate of this phenomenon. On the other hand, we believe the results are indeed useful.

B. Data Collection

We constructed a list D containing 5965 domains belonging to Italian local government administrations (municipalities, provinces, counties) and universities. Domains belonging to public administrations were obtained from a specialized web site[2] whereas those belonging to Universities were downloaded from "Ministry of Instruction, University and Research" web site.

Pharmacy (W_F)	Illegal Drugs (W_D)
viagra	pills
cialis	marijuana
propecia	bong
zithromax	crack
doxycycline	cocaine
clomid	cannabis
levitra	lsd
nolvadex	heroin
lexapro	stoned
amoxil	opium
prednisone	koks
lasix	morphine
silagra	narcotic
tadalis	stimulant
zenegra	reefer

Table I
Lists of target words that we used in our searches (we preferred to omit the Pornography list W_P).

We compiled three lists W_F, W_P, W_D containing *target words* in three categories—pharmacy, pornography, illegal drugs—as follows. We initialized W_F with all the best seller drugs of an on-line shop[3]; W_P and W_D with all the blacklisted words in the parental filtering software Dansguardian[4]. Then, we removed all the non-word items (phrases, URLs, etc.) and performed a web search using the word as query and annotated the number of obtained results. Finally, we retained in each list only the 15 words with the highest number of results (see Table I).

We generated two lists Q_{Bing} and Q_{Yahoo} of *search queries* as follows. For each domain d in D and for each word $w \in W_F \cup W_P \cup W_D$, we added to Q_{Bing} the search query "`site:`d w". The "`site:`d" query portion instructs the search engine to include only results in the domain d. For each domain d in D and for each word list W_F, W_P, W_D, we added to Q_{Yahoo} the search query "`site:`d w_1 `OR ... OR` w_{15}", where $w_i, i \in [1, 15]$ is the set of words in the word list (i.e., 3 queries for each domain d). The "`OR`" keyword instructs the search engine to find web pages containing at least one of the words in the query. While the Bing search API can be used free of charge, usage of the Yahoo search API is charged on a per-query basis. For this reason, we constructed Q_{Yahoo} so as to search for several words at once.

We submitted each query in Q_{Yahoo} to the Yahoo search API and each query in Q_{Bing} to the Bing search API. We kept the first 50 results returned by each query. Using these results we compiled a list R where each element is a tuple $\langle d, W, u_{\text{SE}} \rangle$: d is the queried domain, W is the set of words contained in the query , u_{SE} is the URL returned by the query.

We obtained a list composed of 9459 elements: 6003 returned from the Yahoo API, 3456 from the Bing API. We

[2]http://www.comuni-italiani.it
[3]http://www.drugs-medshop.com
[4]http://www.dansguardian.org

found that 2305 results were obtained from both engines. We merged those duplicate items by setting the W value to the single word obtained from the Bing API and obtained a set R of 7154 elements.

We then collected additional data for each R element $\langle d, W, u_{\text{SE}} \rangle$, as follows. First, we performed 4 HTTP GET requests for each u_{SE}: 3 with the HTTP `Referrer` header set as if the request were generated by a user clicking on a search result obtained from one of the three major search engines (Google, Yahoo, Bing); 1 without including such header. For each GET request, we followed all redirections and saved the *landing URL* of the final web page ($u_{\text{direct}}, u_{\text{Google}}, u_{\text{Yahoo}}, u_{\text{Bing}}$) as well as the corresponding image snapshot ($\mathcal{I}_{\text{direct}}, \mathcal{I}_{\text{Google}}, \mathcal{I}_{\text{Yahoo}}, \mathcal{I}_{\text{Bing}}$).

Second, we associated each element with a single target word $w_a \in W$ as follows. We saved all the DOM trees obtained after the rendering of each landing URL. For elements obtained only from the Yahoo API we took the DOM tree obtained after the rendering of u_{Yahoo}, whereas for all the other elements we took the DOM tree after the rendering of u_{Bing}. We removed from the selected DOM tree: (i) all `script` and `style` HTML elements (along with their content) and (ii) all the HTML tags, hence obtaining a plain text string t which contains all the text rendered in the corresponding web page. We chose as w_a the first word in W found in t. In several cases we could not find in t any word in W, which may be an artifact of our procedure for choosing w_a but also of a change in the web site that had removed all words in W and had not yet been indexed by the search engine.

C. Data Analysis

At this point we analyzed the elements in R to determine whether the corresponding content was legitimate or contained evidence of an attack. To this end we defined four categories, as described below. The analysis required visual inspection of each image snapshot, which was carried out in our lab by five lab staff members who were previously carefully instructed. We could examine a subset of R composed of 3209 elements selected at random.

- NORMAL: the landing URL belongs to domain d and the corresponding page does not appear compromised. Even if the operator detected one of the target words, he deemed its usage legitimate.
- FRAUDMODIFIEDPAGE: the landing URL belongs to domain d; the corresponding page appears compromised, yet part of the legitimate content is still present. The operator identified one of the following scenarios: (a) the page textual content includes a target word and its usage is clearly not legitimate; or (b) the page does not include any target word, yet it includes one or more images which are clearly visible and orthogonal to the page legitimate content.
- FRAUDNEWPAGE: the landing URL belongs to domain d; the corresponding page appears compromised, with no legitimate content apparently present (except for a a few graphical elements such as headers, navigation bar and alike).
- FRAUDOTHERSITE: the landing URL is unrelated to domain d, as a result of a redirection; the corresponding page appears compromised, i.e., totally unrelated to the content of d.

For the elements in which all landing URLs are identical ($u_{\text{direct}} = u_{\text{Google}} = u_{\text{Yahoo}} = u_{\text{Bing}}$), we inspected only $\mathcal{I}_{\text{direct}}$. For the other elements we inspected all the 4 image snapshots, assigned a category to each snapshot and take the most severe value as the category of the element (NORMAL being the least severe and FRAUDOTHERSITE the most severe).

Note that this categorization provides a *conservative* estimate of attacks, emphasizing precision over accuracy. In particular, a page containing fraudulent links could be categorized as being NORMAL.

III. DISCUSSION

A. Key Insights

The main findings of our study are summarized in Table II. We grouped the elements in R associated with the same target word w_a. The table contains a row for the 10 most frequently occurring words w_a, a row describing elements for which we could not find any w_a (labelled $\emptyset$), and a row for all other target words w_a. The row indicating the total counts each domain only once, even when the domain appears in multiple rows. There are several elements containing the same domain-target word pair. We counted such elements only once and considered only the one associated with the most severe category, to simplify the analysis. In other words, Table II counts the number of *different* domains involved in each category.

The key result is that 400 of the 3209 URLs that we could inspect visually (FRAUDMODIFIEDPAGE, FRAUDNEWPAGE, FRAUDOTHERSITE) were actually compromised (12.5%). At the domain level, the compromised domains were 31 out of the 312 that we could analyze (9.9%). As discussed in the final part of the introduction—and keeping in mind the corresponding caveats—these values are indeed surprising (the values for [6] are 1.34% and 1.12%, respectively).

An additional analysis on R elements for which $w_a = \emptyset$ suggests that the number of actually compromised domains could be higher than 31. These elements correspond to pages in which we did not find any target word in the corresponding rendered text. We performed a deeper analysis on a subset of these elements and analyzed the textual snippet that the search engine provided along with the URLs. We found that, for about half of the cases, the page appeared to be actually compromised but later restored—possibly partially. The restored (not compromised) version had not yet been indexed by the search engine. For the remaining cases the visual inspection of the snippet did not enable us to tell whether the page had been actually compromised (note that we did not inspect the DOM, hence we did not search for fraudulent links hidden from the visual content).

w_a	Total URLs	Total Domains	NORMAL URLs	NORMAL Domains	FRAUDMODIFIEDPAGE URLs	FRAUDMODIFIEDPAGE Domains	FRAUDNEWPAGE URLs	FRAUDNEWPAGE Domains	FRAUDOTHERSITE URLs	FRAUDOTHERSITE Domains
crack	404	93	371	87	3	2	30	4	0	0
marijuana	295	75	286	70	2	2	7	3	0	0
viagra	237	44	89	28	4	3	25	5	119	8
cannabis	210	74	210	74	0	0	0	0	0	0
pills	144	35	106	26	4	2	34	7	0	0
prednisone	117	30	113	27	1	1	3	2	0	0
lsd	84	41	84	41	0	0	0	0	0	0
cialis	72	27	38	17	6	2	14	5	14	3
morphine	66	13	66	13	0	0	0	0	0	0
bong	54	26	48	24	0	0	6	2	0	0
$\emptyset$	1213	242	1166	227	18	7	27	6	2	2
Other	313	77	232	67	2	2	54	7	25	1
Total	3209	312	2809	281	40	12	200	11	160	8

Table II
FULL RESULT SET.

w_a	Total URLs	Total Domains	NORMAL URLs	NORMAL Domains	FRAUDOTHERSITE URLs	FRAUDOTHERSITE Domains
viagra	119	8	0	0	119	8
propecia	15	1	0	0	15	1
cialis	14	3	0	0	14	3
levitra	10	1	0	0	10	1
$\emptyset$	22	4	20	2	2	2
Total	180	9	20	2	160	8

Table III
SEARCH REDIRECTION RESULTS: ELEMENTS FOR WHICH THE RETURNED URL DEPENDS ON THE REFERRER OF THE HTTP REQUEST.

Search engines	URLs	Domains
Google	180	9
Bing	74	4
Yahoo	175	9
Google + Yahoo	175	9
Bing + Yahoo	74	4
Google + Bing	74	4
Google + Bing + Yahoo	74	4
no redirection	3029	303

Table IV
NUMBER OF REDIRECTIONS AS A FUNCTION OF SEARCH ENGINES.

B. Redirections and Attack Categories

Table II also shows that, in our sample, redirection to external sites is less frequent than other forms of illegitimate content: there are 23 compromised domains in categories FRAUDMODIFIEDPAGE and FRAUDNEWPAGE, whereas there are 8 domains in FRAUDOTHERSITE.

Moreover, in the analyzed domains illegal drugs appear more frequently than other word categories. On the other side, the ratio between compromised and normal domains which contain a given target word tend to be higher for pharmacy words: e.g., 16 on 44 domains which include the word "viagra" were indeed compromised, whereas only 6 on 93 of domains which include the more frequent word "crack" were compromised.

Table III focuses on search redirection, i.e., it considers only R items for which the returned URL depends on the Referrer of the HTTP request ($u_{\text{direct}} = u_{\text{Google}} = u_{\text{Yahoo}} = u_{\text{Bing}}$ does not hold). Columns for FRAUDMODIFIEDPAGE and FRAUDNEWPAGE are not shown because we did not find any such values in the considered partition of R.

The main finding here is that most of the domains in this partition have been compromised: the partition is composed of 9 domains and 8 of them have been categorized as FRAUDOTHERSITE. In other words, when a redirection is performed basing on the referrer on a web site which contain a target word, the web site has been likely compromised. The 20 web pages categorized as NORMAL were error pages. We could not clearly tell whether the pages have been actually compromised: indeed, these pages could be the result of an attack that succeeded only in part.

It can be seen that we could find only 4 target words w_a and that all of them belong to W_F, the pharmacy category. In other words, all these compromised pages make the user coming from a search engine visit a pharmacy store.

Another interesting outcome is that, in our sample, search redirection does not affect all search engines equally: Table IV describes which search engines actually trigger the referrer-based redirection. Google and Yahoo trigger all the 9 search redirection cases we found, while Bing triggers only 4 of them. We explain this difference because the former is, or is perceived to be, more widely used than the latter and hence attackers concentrate their effort on the former.

Table V shows the results on the other partition of R, i.e., it considers only R items for which the returned URL does not depend on the Referrer of the HTTP request ($u_{\text{direct}} = u_{\text{Google}} = u_{\text{Yahoo}} = u_{\text{Bing}}$ holds). The column for FRAUDOTHERSITE is not shown because we did not find any such values in the considered partition of R. This result corroborates the observation made in the previous Table, i.e., that Referrer-based redirection on an external site is a likely indicator of compromise.

The main finding here is the disparity in the number of compromised domains and URLs in the two categories FRAUDMODIFIEDPAGE and FRAUDNEWPAGE. The two categories exhibit a nearly identical number of compromised domains, but there are much more compromised URLs in the FRAUDNEWPAGE category than in FRAUDMODIFIEDPAGE

w_a	Total URLs	Total Domains	NORMAL URLs	NORMAL Domains	FRAUDMODIFIEDPAGE URLs	FRAUDMODIFIEDPAGE Domains	FRAUDNEWPAGE URLs	FRAUDNEWPAGE Domains
crack	404	93	371	87	3	2	30	4
marijuana	295	75	286	70	2	2	7	3
viagra	118	37	89	29	4	3	25	5
cannabis	210	74	210	74	0	0	0	0
pills	144	35	106	26	4	2	34	7
prednisone	117	30	113	27	1	1	3	2
lsd	84	41	84	41	0	0	0	0
cialis	58	24	38	17	6	2	14	5
morphine	66	13	66	13	0	0	0	0
bong	54	26	48	24	0	0	6	2
$\emptyset$	1211	241	1166	228	18	7	27	6
Other	288	76	232	67	2	2	54	7
Total	3029	311	2809	286	40	13	200	12

Table V
ELEMENTS FOR WHICH THE RETURNED URL DOES NOT DEPEND ON THE REFERRER OF THE HTTP REQUEST.

(this disparity is reflected also in the full set R, Table II). We interpret this result as follows: once an attacker gains sufficient privileges to add a new illegitimate page on a CMS (Content Management System), he will likely exploit these privileges to add further illegitimate pages.

C. URL structure analysis

We investigated the structure of URLs that identify fraudulent web pages. We have found that, as expected, attackers tend to hide fraudulent web pages by placing them "deeply" into the target site.

In detail, for each URL u of fraudulent page we determined its *sub-domain level* as follows: (i) we extracted the domain portion from the URL, say $d(u)$; (ii) we removed from $d(u)$ the trailing string "`www.`" (if present) and the domain obtained from the list D; (iii) we counted the number of dot characters and defined this value to be the sub-domain level of u. Consider for instance the domain name `comune.udine.it`; the URL `www.comune.udine.it/hacked.html` is at level 0 while `segreteria.comune.udine.it/hacked.html` is at level 1.

We have discovered that only 15.2% of fraudulent URLs are at level 0; 71.5% are at level 1 and 13.5% at level 2. In other words, the vast majority of fraudulent URLs (85%) are not at level 0: this result confirms that attackers indeed attempt to hide fraudulent pages—a fraudulent page at level 1 is harder to detect for the web site administrator than a page at level 0.

We also determined the *path depth* of each fraudulent URL u: (i) we removed the two slashes after the protocol name; (ii) we removed the slash character at the end of the domain name; (iii) we counted the number of remaining slashes and defined this value to be the path depth of u. For example, `segreteria.comune.udine.it/hacked.html` has path depth 0, whereas `segreteria.comune.udine.it/people/hacked.html` has path depth 1. We counted 1.6 slashes on the average.

Finally, we found that the *URL length* of fraudulent pages is 77.1 characters on the average, while the URL length of the home page of the target sites is, on the average, only 24.45 characters.

D. Socio-demographic analysis

We performed a few additional analysis to understand whether there is any correlation between compromised sites and some non-technical features of the affected organizations. In particular, we analyzed population and geographical location of municipalities, provinces, and counties. For universities we analyzed number of students and position in a public ranking regarding the quality of the respective IT services. We did not find any significant correlation between compromised sites and these indexes. Of course, the number of compromised sites is too small to draw any statistically relevant conclusion in this respect. However, the lack of any meaningful pattern in this respect seems to be evident: compromised sites tend to be equally distributed in small or large municipalities and universities. Compromised university web sites are scattered more or less randomly across the ranking, going from the 1st to the 51st position. Indeed four of the eleven universities at the top of this ranking have a compromised web site. The geographical position of the organizations owners of the compromised domains are also equally distributed throughout the country.

IV. CONCLUDING REMARKS

We have described recent attack trends in web applications and illustrated their potential dangers for e-government initiatives. Attackers may hide fraudulent content in widely trusted web sites and let those contents appear only to selected users, perhaps depending on the navigation path they followed. The potential effects of attacks of this form are very dangerous: such hidden content is very difficult to detect by administrators and HTTPS—the ubiquitous main line of defense—does not address this threat, thus it does not defend citizens in any way. In our opinion, it is only a matter of time before targeted attacks based on the technical means described here will start appearing. Indeed, criminal attacks on government sites aimed at selling fake certifications are occurring already [9].

We have defined a methodology for detecting fraudulent contents and demonstrated that Italians Public Administrations indeed tend to host fake content. Our study certainly requires further work, in breadth and depth of the analysis, yet we

believe it may help the research community in promoting greater awareness of the problem and in developing solutions both effective and practical.

REFERENCES

[1] Digital Agenda: Commission outlines action plan to boost Europe's prosperity and well-being. May 2010.

[2] Verizon RISK Team. 2012 Data Breach Investigation Report. Technical report, 2012.

[3] A. Bartoli, G. Davanzo, and E. Medvet. The Reaction Time to Web Site Defacements. *Internet Computing, IEEE*, 13(4):52–58, July 2009.

[4] A. Bartoli, G. Davanzo, and E. Medvet. A Framework for Large-Scale Detection of Web Site Defacements. *ACM Transactions on Internet Technology*, 10(3), Oct. 2010.

[5] Z. Gyongyi and H. Garcia-Molina. Web spam taxonomy. In *First International Workshop on Adversarial Information Retrieval on the Web (AIRWeb 2005)*, April 2005.

[6] L. Invernizzi and P. M. Comparetti. Evilseed: A guided approach to finding malicious web pages. *Security and Privacy, IEEE Symposium on*, 0:428–442, 2012.

[7] N. Kroes. A European Strategy for Internet Security. Mar. 2012.

[8] N. Leontiadis and T. Moore. Measuring and analyzing search-redirection attacks in the illicit online prescription drug trade. *Proc. USENIX Security*, 2011.

[9] P. Muncaster. Cyber gang made £30 million from fake gov certs. Technical report, 2012 http://www.theregister.co.uk/2012/07/26/fake_qualifications_scam_busted.

[10] N. Provos, P. Mavrommatis, M. A. Rajab, and F. Monrose. All your iframes point to us. In *Proceedings of the 17th conference on Security symposium*, SS'08, pages 1–15, Berkeley, CA, USA, 2008. USENIX Association.