The Reaction Time to Web Site Defacements

Web site defacement has become a common threat for organizations exposed on the Web. Several statistics indicate the occurrence rate of these incidents but not how long these defacements typically last. The authors present the results of a two-month study of more than 62,000 defacements to determine whether and when a reaction to a defacement occurs. Such reaction times tend to be unacceptably long — often several days — and with long-tailed distribution.

Alberto Bartoli, Giorgio Davanzo, and Eric Medvet
University of Trieste

Web site defacement has become a fact of life on the Internet – similar to phishing, worms, denial of service, and other similar phenomena. Zone-H – a public Web-based archive devoted to gathering defacements (www.zone-h.org) – collected approximately 481,000 defacements during 2007 alone and more than 1.7 million defacements from 2005 to 2007. The Computer Security Institute's (CSI's) Computer Crime and Security survey has documented this problem since 2004 and has recently stated that defacement "continues to plague organizations."[1]

Prior to starting our research, the only available statistics on defacements were primarily concerned with the number of occurrences.[2,3] However, we believe statistics on the typical duration of a defacement are crucial for assessing this phenomenon's practical impact. In this article, we analyze organizations' actual reaction time to defacements. We base our analysis on a sample of more than 62,000 incidents that we monitored in near real time for approximately two months. We found the typical slow reaction time surprising – often several days. For example, roughly 43 percent of the defacements in our sample lasted for at least one week, and more than 37 percent were still in place after two weeks.

We've also analyzed reaction time by isolating mass defacements (described later). We expected to find a much faster reaction time with them,

1089-7801/09/$25.00

but this often isn't the case. Additionally, we analyzed reaction time as a function of the corresponding pages' PageRank values, which measure page importance.[4] We found that pages with higher PageRank values tend to exhibit a more prompt reaction time. Yet, the reaction time tends to be unacceptably long, even in pages whose importance, as quantified by the PageRank, is manifest. We advise researchers to interpret our findings with care — for example, we have no data on the security budget at the organizations involved or on how many clients actually visited the defaced sites. Despite their intrinsic limitations, however, we believe that our findings could highlight important issues that deserve the research community's attention. First, let's look at the effects Web site defacements can have.

The Effects of Web Site Defacements

A defacement that lasts a few weeks is intrinsically much more harmful than one that lasts a few minutes, but it's generally very difficult to quantify this type of damage. These attacks could result in lost revenue due to damaged reputation or missed business opportunities for the company. They could also harm a site's users by hiding or misrepresenting important information. The IT staff costs involved in repairing the site — building a temporary site, investigating the problem, fixing the security breach, and restoring the original content — were estimated at roughly UK£6,000 in 2003,[5] and a CSI survey estimates a total loss of US$725,000 in 2007.

Some of the more potentially damaging defacements that have occurred include those suffered by the Company Registration Office in Ireland, from December 2006 through mid-January 2007,[6] and the Italian Air Force, on 26 January 2007. In August 2007, even the Web sites for the United Nations[7] and the Spanish Ministry of Housing[8] were defaced.

Defacements can be classified in two broad categories: substitutive defacements — that is, the replacement of legitimate content — and additive defacements — that is, the addition of a Web page to a URL where there shouldn't be one. Visitors to an attacked site don't usually notice additive defacements, and such attacks are often meant to constitute covert proof of the successful attack. However, this type of defacement could cause harm if it's part of a more complex attack strategy. For example, a phishing attack could direct users to a fake login page inserted within a trusted site, as an additive defacement.

Our Methodology

To conduct our study, we used information from the Zone-H Digital Attacks Archive. This archive contains a list of URLs that correspond to defacements. A Web form on the Zone-H Web site enables Web surfers or attackers themselves to report the URL u of a defaced page. Then, Zone-H automatically downloads a snapshot $Z(u)$ of the claimed defacement. Later, a human operator at Zone-H verifies the snapshot and, if valid, inserts an entry $\langle Z(u), t_1, t_2 \rangle$ into Zone-H's archive, where t_1 denotes the time instant at which the snapshot $Z(u)$ was taken, and t_2 denotes the time instant at which the operator verifies the defacement. The human operator observes only the snapshot $Z(u)$, which means that, at t_2, the content $W(u)$ exposed on the Web might or might not still be defaced.

We consider that URL u has been defaced when we find that the content exposed on the Web $W(u)$ becomes different from the one that Zone-H stores, $Z(u)$. Our implementation of this basic idea is much more complex than it might seem. For example, we've observed defaced pages that include dynamic content, in which case $W(u)$ differs from $Z(u)$, even though u's administrator hasn't repaired the page. Some defaced pages have intermittent connectivity or are never reachable. We've also observed that the respective administrators have healed some pages before our first check. Extracting useful information from all possible cases automatically turned out to be a significant challenge, and the number of sites that we had to check grew beyond our expectations every hour — we usually had to check more than 20,000 pages per hour. Moreover, the Zone-H archive doesn't provide any indication as to whether a given defacement is substitutive or additive, which has further complicated our analysis.

To perform our analysis, we constructed a tool composed of two components: a *crawler* and a *checker*. The crawler builds list L of defaced pages by querying the Zone-H archive every hour. It then inserts each new entry found at Zone-H into L. For each L entry, the checker implements a finite state machine aimed at tracking the defacement's status, which is checked hourly on the Web to determine whether it's

still in place, removed, or the page is unreachable. For simplicity's sake, we describe only this machine's final states, which contain crucial information for our analysis.

Final States

We associate each entry in L with the time instant, t_f, at which the entry reached the final state. We say that we could verify a defacement for u to mean that our checker indeed verified at least once that $Z(u) = W(u)$. We store $Z(u)$ and $W(u)$ in the form of a hash. We define T_{max} = 2 weeks. We identified five final defacement states. `Undetected` defacements remain in place for at least T_{max}.

`Patched` defacements are those in which we verified the defacement, but sometime later the page changed. We visually inspected 200 entries in this final state and found the following reasons and their respective occurrence rates:

- the legitimate content was restored (65 percent occurrence);
- a maintenance or error message replaced the defacement (20 percent occurrence);
- the defacement was dynamic, or it was a partial defacement of a dynamic Web page, thereby leading to different hash values at different checks (3 percent occurrence); or
- another defacement replaced the first defacement (8 percent occurrence).

We weren't able to identify the remaining 4 percent.

We verified `Removed` defacements in which pages remain systematically unreachable for at least one week. Because of our domain knowledge, we believe that the entries in this final state correspond to previously removed additive defacements. We set t_f to the instant of the first unreachable observation among the consecutive ones lasting one week.

`NotVerified` defacements are those that we didn't verify because the first snapshot $W(u)$ didn't satisfy $W(u) = Z(u)$. We visually inspected 200 entries in this final state, and we found that there might be two main reasons for this outcome:

- before our first check, either the original content was restored or a maintenance message had replaced the defacement (16 percent); or
- the defacement was dynamic, or it was a partial defacement of a dynamic Web page (83 percent).

We weren't able to verify the remaining 1 percent.

Finally, for the `NotChecked` defacements, we couldn't fetch any snapshot of the page — that is, we couldn't fetch $W(u)$ for T_{max}. The most likely reason for this final state is that the entry represents an additive defacement that the site administrator removed before our first check.

Based on the meaning of the final states, we constructed estimates for the incident-reaction time T_r. For `Undetected` entries, $T_r \geq T_{max}$. For either `Patched` or `Removed` entries, $T_r \approx t_f - t_1$. For entries either `NotVerified` or `NotChecked`, all that we can safely say is, if the page had indeed been healed, it was healed at some point before our first check.

We attempted to smooth the effects of this uncertainty in our analysis by discussing our results in two ways: first, we took into account only `Verified` entries — that is, those that were `Undetected`, `Patched`, or `Removed`; then we made an optimistic hypothesis about the reaction time of `NotVerified` and `NotChecked` entries.

Results

We started the crawler and the checker on 17 February 2007. We stopped the crawler on 13 April 2007, after 49 days. The checker continued to run until L was empty, which took another 13 days.

Table 1 summarizes our results in terms of final states. The salient observation is that we found more than 62,000 new defacements on Zone-H in 49 days, which corresponds to roughly 1,250 new URLs every day — a sufficiently large amount to make evident this phenomenon's practical relevance. Table 1 shows 41,002 `Verified` entries, which account for 65.7 percent of all entries.

Verified Entries

Figure 1 shows the percentage of `Verified` entries that the respective administrators detected within a given reaction time t — that is, entries for which $T_r \leq t$ (bold line; we'll discuss the dashed line later). A reaction occurs within one day in less than 25 percent of the `Verified` entries and within one week in roughly 50 percent of them. The average reaction time is $\bar{T}_r$ = 72.4 hours. We were quite surprised by such long re-

action times and such long-tail distributions, which appear to be unacceptably long under any metric.

We analyzed the incident-reaction time T_r for `Patched` and `Removed` entries separately and found that $\overline{T}_r$ is 66.1 and 86.3 hours, respectively. We believe this difference occurs because most `Patched` entries correspond to substitutive defacements, whereas most `Removed` entries correspond to additive defacements, which are intrinsically more difficult to detect.

Note that we base our discussion on the optimistic assumption that `Patched` entries — roughly 40 percent of all `Verified` entries — always involve a reaction to a defacement. Visual inspection of a sample of 200 `Patched` entries (see the definition of the `Patched` final state in the previous section) suggested that only 85 percent of `Patched` entries indeed identified a form of reaction to the attack. It follows that the actual distribution of reaction time is worse than shown here.

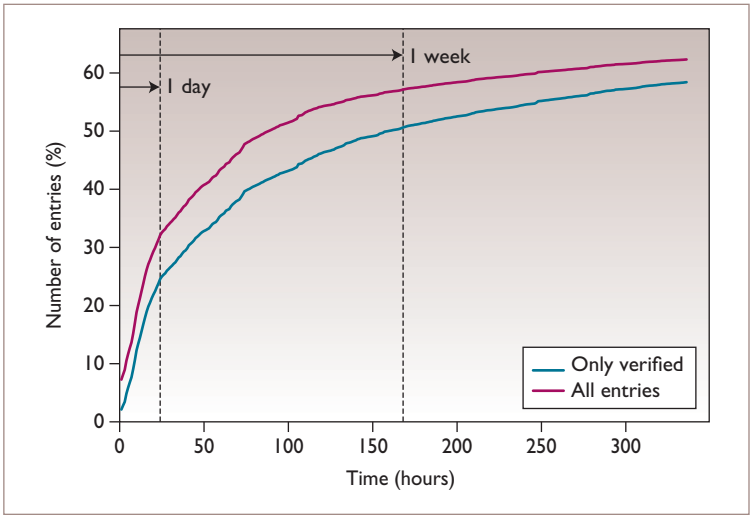

Figure 1. Percentage of defacements that have been detected within a given time (t): the solid line refers to attacks that we were able to verify; the dashed line refers to all attacks (see the text for details).

Table 1. Summary of final states.

State	Number of entries	%
`NotVerified`	18,608	29.8
`NotChecked`	2,823	4.5
`Removed`	7,437	11.9
`Patched`	16,593	26.6
`Undetected`	16,972	27.2
Total	62,433	100.0

Mass Defacements of Verified Entries

We further analyzed defacements by distinguishing between mass and single defacements. A mass defacement consists of multiple URL defacements that are associated with the same IP address. This phenomenon typically occurs when an attacker manages to exploit a vulnerability in a server that hosts multiple Web sites. We thought that the reaction to a mass defacement should be fast because, broadly speaking, sites involved in a mass defacement should be part of a professionally administered hosting infrastructure.

We define an entry as a mass defacement if at least one other entry exists with the same tuple $\langle$`IP address`, $Z(u)$, $t_1\rangle$; Table 2 summarizes the results. The first two columns describe each subset's size: mass and single defacements account for 72.5 and 27.5 percent of the `Verified` entries, respectively. The remaining columns indicate the reaction times for each subset. For example, 28.8 percent of mass defacements have a reaction time of one day. The bottom row indicates the baseline — that is, the corresponding values computed across the whole set of verified entries.

A substantial gap indeed exists between the two subsets' reaction times, both in overall number and percentages at the salient instants considered. This fact confirms our expectation that the reaction time to mass defacements is quicker than it is to single defacements. Nevertheless, the reaction time remains quite slow, even with mass defacements — after one week, site administrators detect only half the entries, and the average reaction time is 2.7 days. It seems reasonable to claim that defacements and reaction times to defacements are also a significant issue for Web site providers.

Verified Entries Based on PageRank Value

We attempted to gain some insight into the possible correlation between a defaced Web page's importance and the corresponding reaction time. Quantifying the former is obviously quite difficult, particularly when dealing with thousands of Web pages. For example, a Web page might be useful only to a very small set of users who might heavily depend on that page's integrity. A Web page could also remain defaced only during a time interval in which few of its users, if any, access that page and could thus go unnoticed.

Rather than plainly neglecting this issue,

Table 2. T_r characterization for mass and single defacements.

Partition	Number of entries	%	Detected at one day (%)	Detected at one week (%)	Detected at two weeks (%)	$\bar{T}_r$ (hours)
Mass defacement	29,719	72.5	28.8	53.2	59.9	64.9
Single defacement	11,283	27.5	13.3	43.9	54.4	94.2
Total	41,002	100.0	24.6	50.6	58.4	72.4

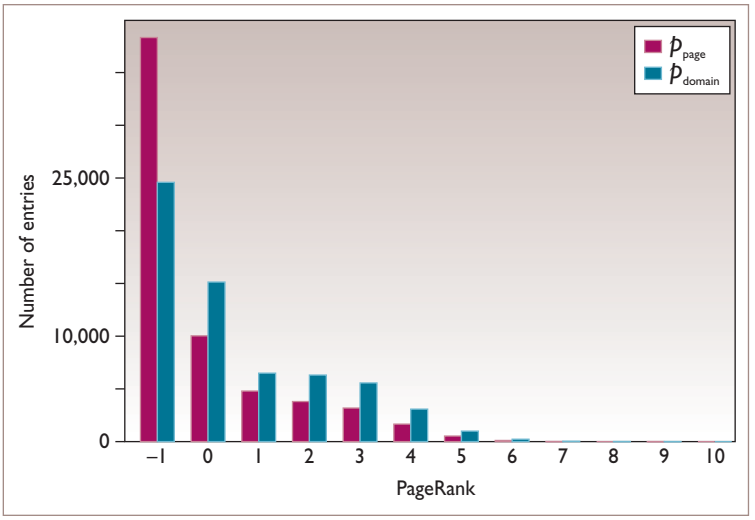

Figure 2. Distribution of p_{page} and p_{domain} for all considered entries. Value −1 corresponds to URLs for which PageRank isn't available.

however, we decided to incorporate the PageRank values of the defaced pages into our analysis.[4] A page's PageRank is an integer in the range [–1, 10] — the higher the value, the more important the page; a value of –1 means that a PageRank value isn't available for the corresponding URL. We couldn't see any sensible alternative for assessing a page's importance in our scenario — tens of thousands of defaced pages exist, and their access patterns are unknown. Note, though, that you can't establish any immediate relation between a page's importance, as quantified by its PageRank, and the actual damage the defacement causes — for example, pages with a very low PageRank might actually be, and indeed often are, quite important to their users.

For each entry with URL u, we automatically collected both the PageRank value p_{page} associated with u and the PageRank value p_{domain} associated with the u domain. Figure 2 shows the distribution of p_{page} and p_{domain} across all the entries. Note that p_{page} is –1 for approximately 39,000 entries, but a substantial number of entries exist for which p_{page} isn't negligible: $p_{page} \geq 3$ for 5,488 entries — roughly 9 percent of all entries. We can say the same of p_{domain}: for thousands of entries, $p_{domain} \geq 3$. These results suggest that defacements also affect PageRank-important sites.

Concerning only `Verified` entries, we found that the relative occurrence of entries with $p_{page} \neq -1$ is much higher in `Patched` entries than in `Removed` entries: 35 percent versus 7.4 percent. This fact confirms our hypothesis that most `Removed` entries probably correspond to additive defacements — that is, pages that aren't supposed to exist. For such pages, the PageRank value isn't available.

Figure 3 correlates reaction times to PageRank values, which constitutes our analysis's salient point: it shows the percentage of `Verified` entries that the respective administrators have detected within one day or one week, as a function of their p_{page} values. For example, in 35.3 percent of the entries with $p_{page} = 0$, some form of reaction (either patching or removing) occurs within 1 day. As a baseline, dotted horizontal lines represent the values averaged for all `Verified` entries independently of their PageRank value (described earlier).

The figure confirms our hunch that pages with a higher PageRank value have a faster reaction time to defacements — particularly if we consider detection percentages at one week. However, it seems fair to claim that, even from this viewpoint, the reaction time isn't as fast as you'd probably expect — for example, we never observed a one-day detection percentage significantly greater than 50 percent.

All Entries

We wanted to verify whether the incident-reaction times we described earlier were distorted by the fact that we considered only `Verified` entries, hence discarding the remaining 34.3 percent of the 62,433 entries. To this end, we reasoned about how to take into account entries that were either `NotVerified` — that is, the site as exposed on the Web was never the same as its snapshot stored in ZoneH — or `NotChecked` (we couldn't fetch or read anything from the site for at least two

weeks). For these entries, we can only say that, if the page had indeed been healed, it had been healed at some point before our first check.

We formulated an optimistic set of hypotheses:

- Each `NotChecked` entry was healed as soon as Zone-H downloaded the defacement (that is, $T_r = 0$).
- 16 percent of `NotVerified` entries were also healed immediately (consistent with our visual inspection of a sample, described earlier).
- The remaining `NotVerified` entries were distributed as `Verified` entries: 41.4 percent of them were `Undetected`, and the others were either `Patched` or `Removed` with a reaction time distributed as shown in Figure 1.

Figure 1 presents the corresponding results. As expected, the dotted line shows a significant detection rate immediately ($T_r \leq t = 0$), more so than the solid line, which depicts the `Verified` entries. The first-day and first-week detection rates are 32.1 and 57.2 percent, respectively, as opposed to 24.6 and 50.6 percent for the `NotVerified` entries. The average reaction time is 54.5 hours for the entries associated with the dotted line in Figure 1, as opposed to 72.4 hours for the `Verified` entries. The improvement is significant, yet it seems reasonable to claim that the reaction time remains unacceptably long. If we assume, even more optimistically, that all `NotVerified` entries were healed immediately, first-day detection rates become 50.5 percent, first-week detection rates become 67.6 percent, and average reaction times become 38.3 hours. Once again, the essence of our findings remains unchanged.

Although existing technology allows for Web-defacement detection within a few minutes of its occurrence, our results show that the actual reaction time tends to be much slower, usually on the order of several days. It seems reasonable to claim that most sites involved in our monitoring campaign either don't use any of the technologies we've described (see the sidebar, "Tools for Web Defacement Detection"), or they don't react promptly to relevant alerts, and further research is needed to provide an objective explanation for this finding. You could argue that the defaced content is of little value to Web site owners, but we're reluctant to accept this explanation based on our analysis in terms of mass defacements and PageRank.

We believe that our work might help people

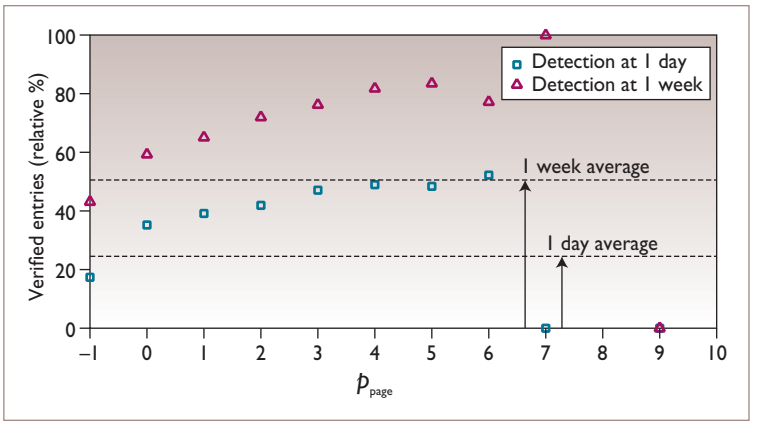

Figure 3. The percentage of `Verified` entries (y-axis) that have been detected within one day (square) or one week (triangle), as a function of the corresponding p_{page} (x-axis). Percentages for $p_{page} > 6$ aren't significant because too few entries have such a PageRank value.

Tools for Web-Defacement Detection

Several tools and services are available for automatic Web-defacement detection, such as Tripwire[1] — a popular file-system integrity checker that detects any deviations from a baseline content previously established by a Web site's administrator and kept at a secure location. Storage-based intrusion-detection systems offer similar functionalities, and these systems let the administrator specify an extensible set of suspicious update activities.[2–6] Tools specifically devoted to Web applications compare content exposed on the Web to a copy of the content that's stored in a safer location.[7,8] Clearly, whenever Web site administrators modify a Web resource, they must update the corresponding copy accordingly. The tools can perform a comparison periodically or whenever the content is about to be returned to a client, and they can be run within the site they're monitoring[7] (for example, WebAgain; www.lockstep.com[8]) or remotely (for example, Catbird; www2.catbird.com). The alerts that these technologies generate hardly represent a false positive, thus the Web site administrator should give them much higher priority than, for example, the flow of alerts that network-based intrusion-detection systems generate.

References

1. G.H. Kim and E.H. Spafford, "The Design and Implementation of Tripwire: A File System Integrity Checker," *Proc. 2nd ACM Conf. Computer and Comm. Security* (CCS 94), ACM Press, 1994, pp. 18–29.
2. A.G. Pennington et al., "Storage-Based Intrusion Detection: Watching Storage Activity for Suspicious Behavior," *Proc. 12th Usenix Security Symp.* (Usenix 03), Usenix Assoc., Aug. 2003.
3. M. Banikazemi, D. Poff, and B. Abali, "Storage-Based File System Integrity Checker," *Proc. 2005 ACM Workshop on Storage Security and Survivability* (StorageSS 05), ACM Press, 2005, pp. 57–63.
4. G. Sivathanu, C.P. Wright, and E. Zadok, "Ensuring Data Integrity in Storage: Techniques and Applications," *Proc. 2005 ACM Workshop on Storage Security and Survivability* (StorageSS 05), ACM Press, 2005, pp. 26–36.
5. A. Gehani, S. Chandra, and G. Kedem, "Augmenting Storage with an Intrusion Response Primitive to Ensure the Security of Critical Data," *Proc. 2006 ACM Symposium on Information, Computer, and Comm. Security* (Asiaccs 06), ACM Press, 2006, pp. 114–124.
6. N. Fujimura and J. Mei, "Implementation of File Interpolation Detection System," *Proc. 35th Ann. ACM Siguccs Conf. User Service*, ACM Press, 2007, pp. 118–121.
7. S. Sedaghat, J. Pieprzyk, and E. Vossough, "On-the-Fly Web Content Integrity Check Boosts Users' Confidence," *Comm. ACM*, vol. 45, no. 11, 2002, pp. 33–37.
8. W. Fone and P. Gregory, "Web Page Defacement Countermeasures," *Proc. 3rd Int'l Symp. Comm. Systems Networks and Digital Signal Processing*, IEEE CS Press, 2002, pp. 26–29.

understand the Web defacement phenomenon and its practical impact. Although the Web has become an essential component in our society, organizations tend to expose resources that not only could be hacked and defaced, but that tend to remain defaced for several days.

References

1. R. Richardson, "2007 CSI Computer Crime and Security Survey," 2007; www.gocsi.com/forms/csi_survey.jhtml.
2. Zone-H.org, "Statistics Report 2005–2007," Mar. 2008; www.zone-h.org/news/id/4686.
3. K.N. Srijith, "Analysis of the Defacement of Indian Web Sites," *First Monday*, vol. 7, no. 12, 2002; http://firstmonday.org/issues/issue7 12/srijith/.
4. L. Page et al., *The PageRank Citation Ranking: Bringing Order to the Web*, tech. report, InfoLab, Stanford Univ., 1998.
5. G. Killcrece et al., *State of the Practice of Computer Security Incident Response Teams (CSIRTs)*, tech. report CMU/SEI-2003-TR-001, ESC-TR-2003-001, Software Eng. Inst., Carnegie Mellon, 2003; www.sei.cmu.edu.
6. G. Smith, "CRO Web Site Hacked," Apr. 2007; silicon republic.com/news/news.nv?storyid=single7819.
7. G. Keizer, "'Hackers' Deface UN Site," *Computerworld*, Aug. 2007; www.computerworld.com/action/article.do?command=viewArticleBasic&articleId=9030318.
8. H.B., "Hacker Attacks the Ministry of Housing's Web Site as Spanish Mortgages Come Under the International Spotlight," Typicallyspanish.com, Aug. 2007; www.typicallyspanish.com/news/publish/article_12212.shtml.

Alberto Bartoli is an associate professor in the Department of Computer Engineering at the University of Trieste. His research interests include Web security and distributed systems. Bartoli has a PhD in computer engineering from the University of Pisa. Contact him at bartoli.alberto@univ.trieste.it.

Giorgio Davanzo is a PhD candidate in the Department of Computer Engineering at the University of Trieste. His research interests include Web security. Davanzo has an MS in computer engineering from the University of Trieste. Contact him at giorgio.davanzo@deei.units.it.

Eric Medvet is a researcher at the University of Trieste. His research interests include Web security. Medvet has a PhD in computer engineering from the University of Trieste. Contact him at emedvet@units.it.