virus BULLETIN Covering the global threat landscape

DISSECTING THE DESIGN AND VULNERABILITIES IN AZORULT C&C PANELS

Aditya K Sood
Research Team at Office of The CTO, F5

OVERVIEW

AZORult malware has been around in the wild for a couple of years and is very effective at stealing sensitive information from end-user systems. Our research team has been tracking this threat for many months in order to understand its intrinsic details. In this paper, we focus in particular on the command-and-control (C&C) design of the AZORult malware – we discuss our findings related to the C&C design and some security issues that we have identified.

The research:

- Unveils the design of C&C operations, specifically the design of the C&C panels.
- Uncovers inherent security vulnerabilities in the AZORult C&C panels that will help us to collect advanced intelligence regarding the attacker-controlled portals.
- Provides intelligence that can be used in security products to generate advanced analytics and to detect malicious communication with the C&C panels.

BACKGROUND

AZORult has been widely used to steal sensitive information from infected systems, the stolen information subsequently being used for nefarious purposes. A number of research teams have already presented their analyses of the malicious nature of the binary installed on the end-user systems, detailing the techniques and tactics used by the malware itself. Some of the previous research work includes:

- The U.S. Department of Health and Human Services (HHS) [1] released an analysis of AZORult malware characteristics and how the malware has been used in attacks that scam end-users by spreading fake messages related to COVID-19.
- *Kaspersky* [2] highlighted how the AZORult malware campaign has been used to target end-users to steal VPN credentials.
- *Trend Micro* [3] has also presented an analysis of the working nature of AZORult malware.
- *Palo Alto Networks* [4] highlighted the use of earlier versions of AZORult malware in collaboration with the Fallout exploit kit.

UNDERSTANDING THE DESIGN OF THE AZORULT C&C PANEL

In this section, we will dig deeper into the design of the AZORult C&C panel and elaborate on the most important functions and features of the C&C panel. If you are interested in understanding the advanced architecture of C&C panels, please refer to the research related to SpyEye [5] and our empirical analysis [6] of HTTP-based C&C panels.

AZORult C&C web panel layout

Let's first take a look at how the C&C panel of AZORult looks when deployed on a server and exposed to the Internet. Figure 1 shows the AZORult C&C panel. The web panel is very basic in its design and uses a simple HTML web form functionality.

Looking at the panel and admin.php file, the web panel simply prompts a basic HTML web form and asks for a password. No username HTML form field is available as the C&C panel only requires the password to authenticate the user. There is no protection against brute-force attacks in the form of a CAPTCHA either.

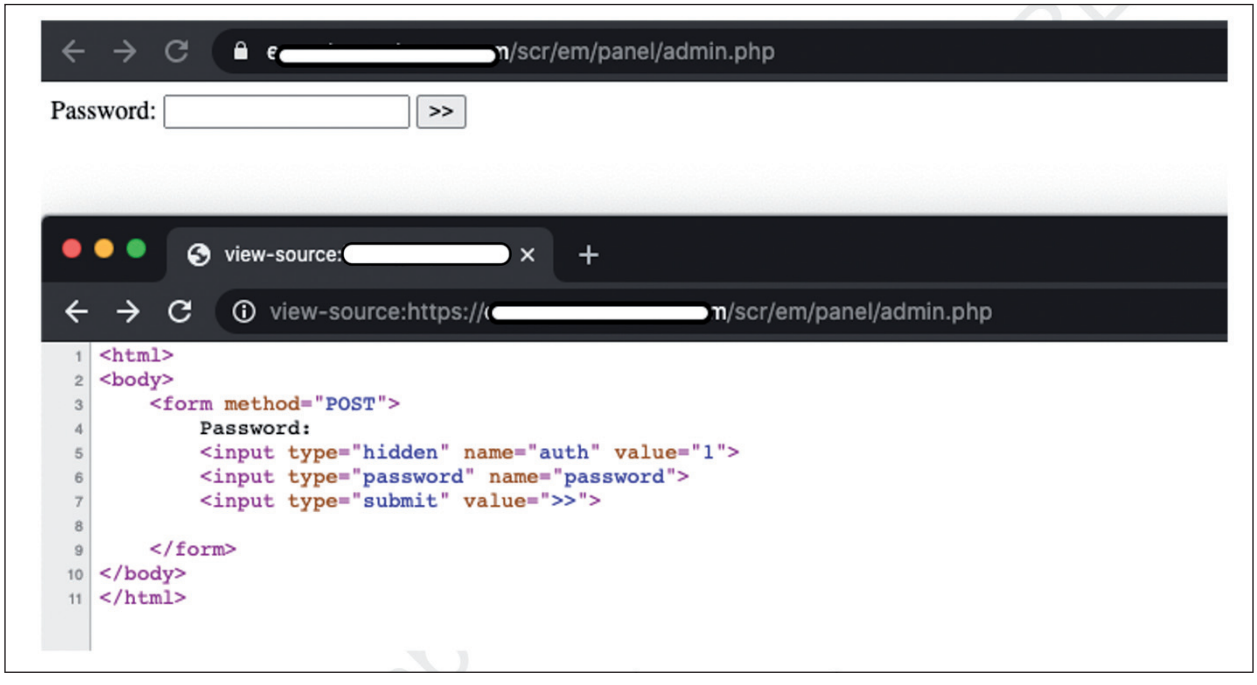

Figure 1: AZORult live C&C panel.

AZORult C&C panel components

In Table 1, we take a look at the different components of the C&C panel.

Component	Description
admin.php	This component configures the settings of the C&C panel including error handling, authentication, authorization, database configuration, and others.
index.php	The main landing page of the C&C panel from where access is granted to the botnet operator.
config.json	A component that defines configuration settings that the bot uses to communicate with the C&C panel. The configuration settings are embedded directly into the binary (exe or dll) and the same binary is distributed to be installed on the infected end-user systems.
functions.php	Supporting functions such as error_reporting, base64Decrypt, traffic_decrypt, etc. are defined in this component.
guest.php	A component that provides restricted access, i.e. guest or anonymous, to the specific functionality of the C&C panel capabilities.
maxmind.php	A component that allows the C&C to use geolocation databases to map bot IP addresses to locations across the Internet.
bin.bin	A binary component which is encrypted and encoded in nature and generated for specific functionality to trigger additional infections on the infected end-user system.
dump.sql	A basic component that allows bulk dumping of SQL data by running specific SQL queries.

Table 1: AZORult C&C panel components.

Cookie creation for authenticated session

Let's understand how the cookies are generated for setting up authenticated sessions with the C&C panel. The code is presented below:

```
function Auth(){
     if (@$_POST['auth']=="1"){
        sleep(
2);
        if (@$_POST['password']==ADMIN_PWD)
        {
             header('Set-cookie: pwd='.md5($_POST['password'].$_SERVER['HTTP_USER_AGENT'].'').'; httpOnly' );
             header("Location: ".$_SERVER['PHP_SELF']);
        };
     };
};
     Auth();
     if((@$_COOKIE["pwd"]) != md5(ADMIN_PWD.$_SERVER['HTTP_USER_AGENT'].'')) die(FileToString('./html/login.html'));
     Main(); ?>
```

If you look at the code above, you will notice that the HTTP POST request is processed and the parameter '*password*' is checked against the known administrative password (ADMIN_PWD). If the supplied password matches the admin password, the '*Set-Cookie*' header is generated with the required values to set up the session. The '*Set-Cookie*' header contains the '*pwd*' parameter, which contains the MD5 hash of the password concatenated with the user-agent string supplied by the client (browser). The '*Set-Cookie*' header is transmitted back to the browser and the session is authenticated. If the MD5 hash value is tampered with, the session is killed. The important point to note here is that the MD5 hash actually contains the admin password of the portal.

CSRF token generation

Let's look at how the AZORult C&C panel generates CSRF tokens for protecting against cross-site application attacks. The code below presents how the CSRF token is created:

```
$CSRF_TOKEN = md5("yukd894as98d4v".md5(ADMIN_PWD.$_SERVER['HTTP_USER_AGENT'].""));
```

If you scan the code above, you will notice that the CSRF token is an MD5 hash of the hard-coded string '*yukd894as98d4v*' appended with the MD5 hash of the session token. This shows that the CSRF token is a hash of the combination of a pre-populated random string and a session token.

Stolen password data storage and retrieval in C&C panel

In this section, we will look into the type of data stored in the C&C panel database. The data here refers to the credentials and sensitive information stolen from the infected end-user systems running the AZORult bot.

```
.....
$arr = SQLToArray(" SELECT 'p_soft_type', COUNT(*) AS CountRec, COUNT(*)/(SELECT COUNT(*) FROM
passwords)*100 AS percent
                                FROM passwords
                                GROUP BY 'p_soft_type'
                                ORDER BY CountRec DESC");
                for ($i=0; $i<sizeof($arr); $i++)
                {
                        if($arr[$i]["p_soft_type"] ==  "1") $arr[$i]["p_soft_type"]="Browsers";
                        if($arr[$i]["p_soft_type"] ==  "2") $arr[$i]["p_soft_type"]="FTP Clients";
                        if($arr[$i]["p_soft_type"] ==  "3") $arr[$i]["p_soft_type"]="Mail Clients";
                        if($arr[$i]["p_soft_type"] ==  "4") $arr[$i]["p_soft_type"]="IM Clients";
                };
```

The '*p_soft_type*' parameter in the code refers to the different types of client software that are susceptible to infection and extraction of credentials by the AZORult bot. These include browsers, FTP clients, mail clients, and instant messaging (IM) clients. You can cross-check this information in the leaked '*config.json*' file as part of the information disclosure vulnerability discussed later. Figure 2, taken from the AZORult C&C panel, shows how the information extracted using '*p_soft_type*' parameters is shown in the form of reports, i.e. password type stats.

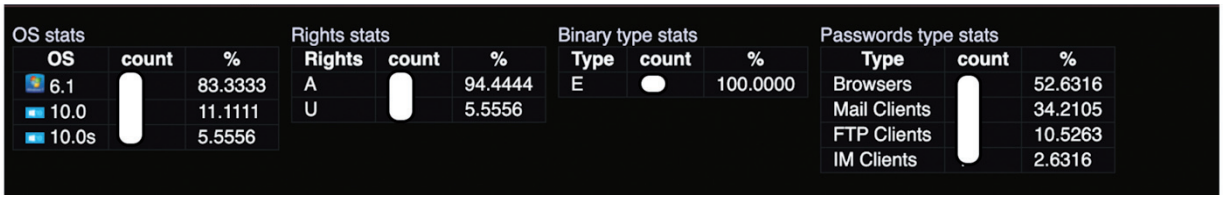

Figure 2: Exposed password type stats.

Further, the code below highlights how the C&C panel manages and stores the stolen passwords collected from infected systems. This information is extracted from the database and displayed in a structured format in the C&C panel so that the AZORult operator can analyse and use the information.

```
function ShowPasswordsPage(){
    foreach ($_GET as $value) {
        if (is_array($value)) die(); }
    $link = mysqli_connect(DB_HOST, DB_USER, DB_PASS, DB_NAME);
    $query="SELECT passwords.p_soft_name,
                passwords.p_p1,
                passwords.p_p2,
                passwords.p_p3,
                passwords.r_id
            FROM passwords, reports
            WHERE reports.r_id=passwords.r_id AND passwords.p_soft_name<>''";
    if(isset($_GET['search'])
    if(strlen($_GET['search'])>0)

    $query .= " AND LOCATE('".mysqli_real_escape_string($link, $_GET['search'])."', concat_ws('|', passwords.p_
    soft_name, passwords.p_p1, passwords.p_p2,passwords.p_p3,passwords.r_id))";

    -- Truncated --

    if(isset($_GET['soft_type1'])) {if ($_GET['soft_type1']=="1") $query .=" ";}
    else $query .=" AND passwords.p_soft_type<>1";

    if(isset($_GET['soft_type2'])) {if ($_GET['soft_type2']=="1") $query .=" ";}
    else $query .=" AND passwords.p_soft_type<>2";

    if(isset($_GET['soft_type3'])) {if ($_GET['soft_type3']=="1") $query .=" ";}
    else $query .=" AND passwords.p_soft_type<>3";

    if(isset($_GET['soft_type4'])) {if ($_GET['soft_type4']=="1") $query .=" ";}
    else $query .=" AND passwords.p_soft_type<>4";

    if(isset($_GET['r_id'])) {if (($_GET['r_id']<>"") and (is_numeric($_GET['r_id']))) $query .=" AND
                passwords.r_id=".mysqli_real_escape_string($link, $_GET['r_id']);};

    if(isset($_GET['inc_il'])) if ($_GET['inc_il']=="1")
    {
            $query .= " AND ( ";
```

```
            $links = explode("\r\n", FileToString("./links.txt"));
            foreach ($links as $key => $value) {
            //echo $value;
            $query .= "(passwords.p_p1 LIKE '".mysqli_real_escape_string($link, $value)."') OR";
            }
$query = substr($query,0,-2};
    $query .= " ) ";} $query .=" ORDER BY passwords.p_soft_type, passwords.p_soft_name";
--- Truncated --
```

The code above reveals exactly how the AZORult C&C panel runs different database SQL queries to dump the stolen passwords in the form of reports to be consumed by the AZORult operator. Let's look into some of the information disclosure vulnerabilities that we discovered during the course of this research.

AZORULT C&C PANEL VULNERABILITIES

In this section, a number of AZORult C&C panel security vulnerabilities are presented that leak information about the panel and its inherent configuration.

Unrestricted access to configuration file

AZORult has a built-in security configuration issue that allows end-users to access the configuration file on the fly. This means that any end-user can access this file in an unauthenticated manner. The file is named '*config.json*' by default. The configuration of broad privileges allows any anonymous user to access this file. Figure 3 highlights the exposed configuration of the C&C panel.

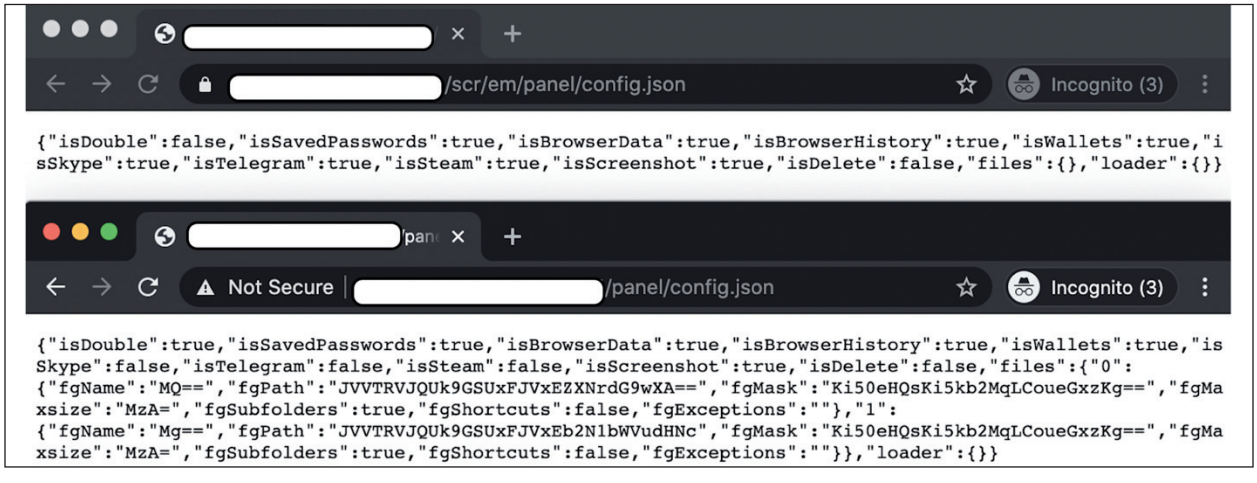

Figure 3: Accessing the configuration file in an unauthenticated manner.

It is important to decode the output present in the configuration file presented below.

```
{
    "isDouble": true,
    "isSavedPasswords": true,
    "isBrowserData": true,
    "isBrowserHistory": true,
    "isWallets": true,
    "isSkype": false,
    "isTelegram": false,
    "isSteam": false,
    "isScreenshot": true,
    "isDelete": false,
    "files": {
      "0": {
```

```
      "fgName": "MQ==",
      "fgPath": "JVVTRVJQUk9GSUxFJVxEZXNrdG9wXA==",
      "fgMask": "Ki50eHQsKi5kb2MqLCoueGxzKg==",
      "fgMaxsize": "MzA=",
      "fgSubfolders": true,
      "fgShortcuts": false,
      "fgExceptions": ""
    },
    "1": {
      "fgName": "Mg==",
      "fgPath": "JVVTRVJQUk9GSUxFJVxEb2N1bWVudHNc",
      "fgMask": "Ki50eHQsKi5kb2MqLCoueGxzKg==",
      "fgMaxsize": "MzA=",
      "fgSubfolders": true,
      "fgShortcuts": false,
      "fgExceptions": ""
    }
  },
  "loader": {}
```


Let's decode the values first to understand more about the configuration parameters. See Table 2 for more details.

Configuration parameter	**Description**
isSavedPasswords	A module that instructs the AZORult bot to steal passwords and transmits them to the backend database configured for the C&C operations.
isBrowserData	A module that allows the AZORult bot to extract information about the active browser running in the compromised end-user system.
isBrowserHistory	A module that directs the AZORult bot to steal browser history from the infected system.
isWallets	A module that directs the AZORult bot to steal stored wallet information from the infected system.
isSkype	A module that directs the AZORult bot to steal Skype Instant Message (IM) client information from the infected system, including stored credentials, chats, etc.
isTelegram	A module that directs the AZORult bot to steal Telegram client information from the infected system, including stored credentials, chats, etc.
isScreenshot	A module that instructs the AZORult bot to capture screenshots of the active browser and other software sessions and transmit them to the C&C for storage
isDelete	A module that allows the AZORult bot to conduct delete operations on the infected system to clean up its tracks if required
isSteam	A module that directs the AZORult bot to steal Steam software information from the infected system, including stored credentials, etc.
files{}	A module used to define the file settings used by the bot on the client-side
loader{}	A module used to direct the bot to load and execute additional malicious code on the compromised end-user system

Table 2: AZORult C&C configuration file parameters.

Let's further decode the parameters in the '*files*' configuration parameter in Table 3.

Configuration parameter	Encoded value if any	Decoded value	Details
fgName	MQ==	1	Potentially used to define the file name and numbering
fgPath	JVVTRVJQUk9GSUxFJVxEZXNrdG9wXA==	%USERPROFILE%\Desktop\	Directory path on the end-user system where the file is stored
fgMask	Ki50eHQsKi5kb2MqlCoueGxzKg==	*.txt, *.doc**.xls*	File types supported for writing data and processing
fgMaxsize	MzA==	30	Defines the size of the file
fgSubfolders	true	true	Interacts with subfolders on the end-user system.
fgShortcuts	false	false	Supporting file shortcuts
fgExceptions	""	null	Handling and reporting any exceptions

Table 3: AZORult C&C configuration parameter 'file {}' array.

Exposing configuration parameters reveals how the AZORult bot is operating in the infected end-user system. With this information, extensive intelligence can be built without reversing the AZORult binary. Additionally, a number of IoCs can be generated to detect the presence of AZORult malware in the system and further eradicate infections accordingly.

Guest and anonymous access allows traffic stats

Another interesting security issue that has been identified is the exposure of a successful infection stats web page to unauthenticated users. Figure 4 highlights one such example of the real-world AZORult C&C panel.

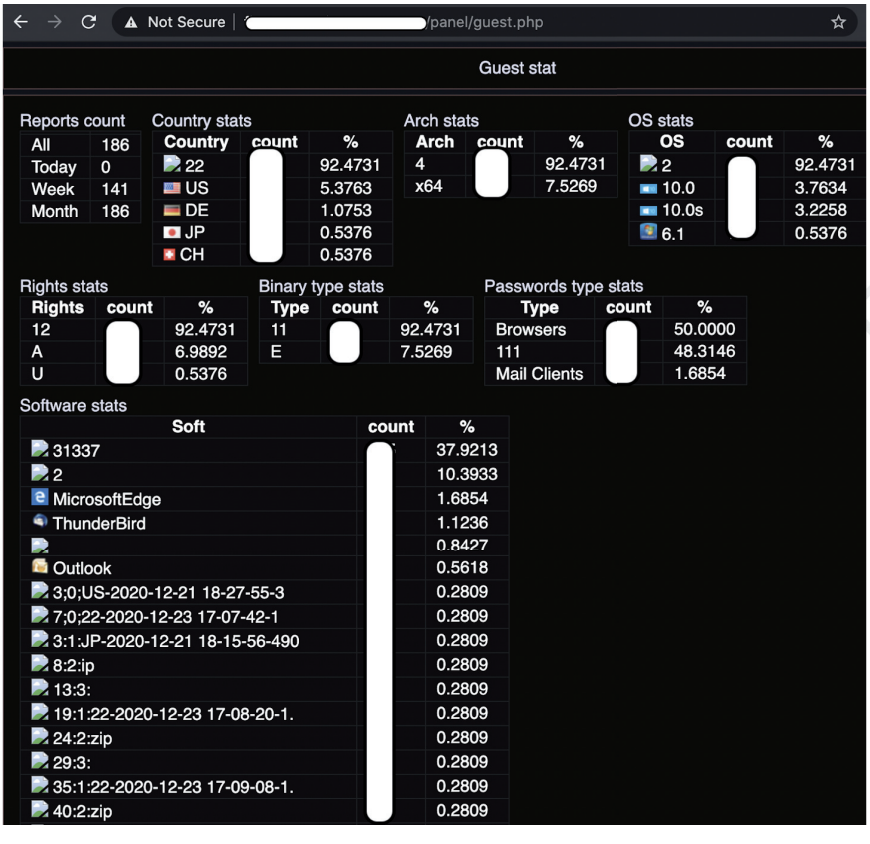

Figure 4: Infection stats file can be viewed by guests.

From an intelligence point of view this information is crucial because it reveals how many bots are connecting back to the deployed C&C panel and includes information such as types of browsers compromised, binary types, country stats, infected operating systems, and more. This information helps threat researchers and threat responders determine the risk level associated with this C&C panel and actions that need to be taken to minimize the impact.

Information disclosure via default PHP pages

One of the other information security issues that exists in the deployment of C&C panels is the presence of default PHP pages in the same web directory as that in which C&C web components are hosted. This is not a security issue of the C&C panel itself, but a side effect of the deployment in which additional software packages such as PHP, MySQL, etc. are required to run the C&C panel. Figure 5 shows an example of an exposed PHP web page discovered on the AZORult C&C deployment.

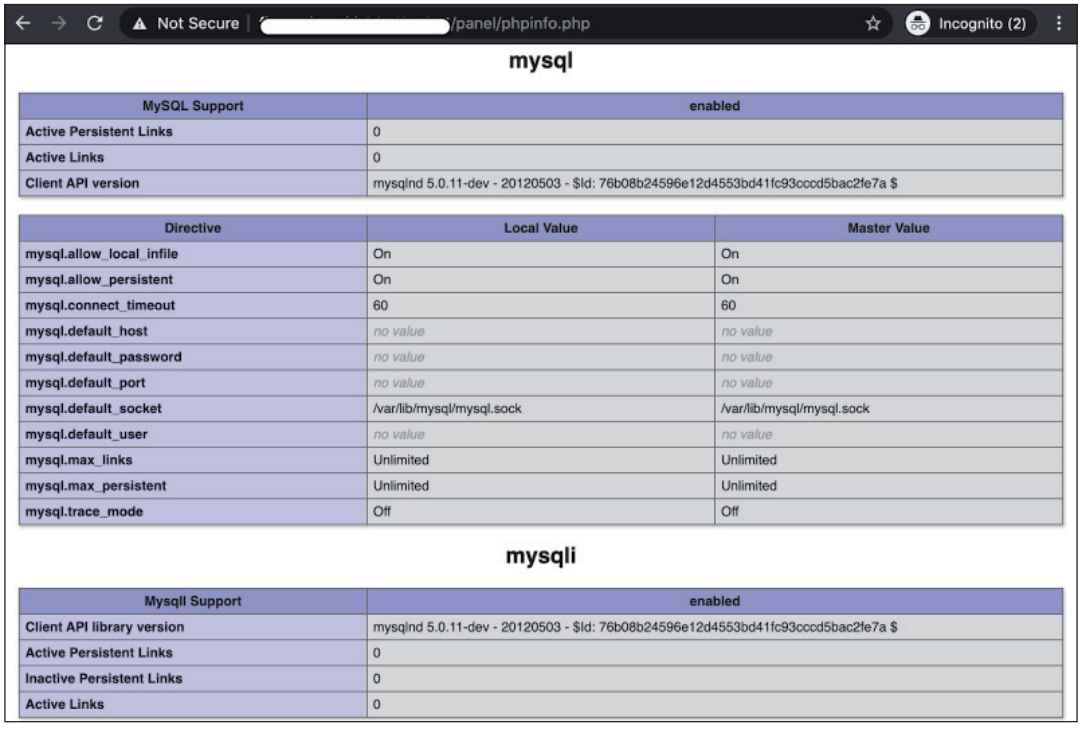

Figure 5: Exposed PHPInfo web pages hosted in the C&C panel web directory.

The default PHP info web page reveals information about the underlying architecture that is running AZORult C&C panels. The information includes but is not limited to:

- The configured PHP version.
- Installed operating system and its version.
- Significant details of the PHP configuration.
- Information about the internal IP addresses.
- Details about the server environment variables.
- Installed and loaded PHP extensions and their configurations.

The above information helps gain an understanding of the architecture and installed software that is used to deploy C&C panels.

GENERATING THREAT INTELLIGENCE

Using an offensive approach to threat research can help to build threat intelligence as discussed below:

- Vulnerabilities in C&C panels reveal the weaknesses that exist in the server-side software used by the botnet (malware) operators to command and control the malicious code running on compromised systems.

- Information from C&C panels can help to build indicators of compromise (IoCs). For example, the malicious code residing on the compromised systems connects back to C&C panels. Knowing the server-side information about the C&C panel, we can scan and dissect the network traffic to detect potential infections. Another example is that the leaking of information about compromised systems on the Internet helps us to identify how many active infections are running, including the type of operating systems, target browsers, etc.
- Understanding the design of C&C internals helps us to gain intelligence on how the information (credentials, client information, etc.) stolen from the end-user systems is stored and managed by the botnet operators. In fact, most important is where exactly the stolen information is stored in the database on the C&C infrastructure.
- Intelligence gained from the C&C panels can be used to feed back into security solutions in order to detect and prevent malware infections on the fly in an automated manner.
- The intelligence can also be used to build 360 degree threat profiles, conducting threat analytics and empirical analysis for correlating other sets of threats in order to understand the threat landscape better.
- Intelligence can also be used to build security features to conduct data science (DS) and machine learning (ML) experiments for detecting potential anomalies related to specific threats.

Overall, in order to combat advanced threats, an offence-to-defence (OTD) approach to research is the need of the hour.

CONCLUSION

In this paper we discussed the C&C panel design of AZORult malware including a number of information disclosure vulnerabilities that result from insecure authentication mechanisms in place. The information leakage via the C&C panel reveals a lot of significant intelligence about the AZORult threat. Potential flaws in C&C infrastructure owned by the malware operators present useful threat intelligence to defend against ongoing active threats in the wild.

REFERENCES

[1] AZORult Malware. U.S. Department of Health & Human Services. https://www.hhs.gov/sites/default/files/azorult-malware.pdf.

[2] New AZORult campaign abuses popular VPN service to steal cryptocurrency. Kaspersky. https://www.kaspersky.com/about/press-releases/2020_new-azorult-campaign-abuses-popular-vpn-service-to-steal-cryptocurrency.

[3] AZORULT Malware Information. Trend Micro. https://success.trendmicro.com/solution/000146108-azorult-malware-information-kAJ4P000000kEK2WAM.

[4] Yan, T.; Jin, X.; Qu, B.; He, Z. New Wine in Old Bottle: New Azorult Variant Found in FindMyName Campaign using Fallout Exploit Kit. Unit 42 Palo Alto Networks. https://unit42.paloaltonetworks.com/unit42-new-wine-old-bottle-new-azorult-variant-found-findmyname-campaign-using-fallout-exploit-kit/.

[5] Sood, A. K.; Enbody, R. J.; Bansal, R. Dissecting SpyEye – Understanding the design of third generation botnets. Computer Networks, volume 57, issue 2, pp. 436-450. https://www.sciencedirect.com/science/article/abs/pii/S1389128612002666.

[6] Sood, A. K.; Zeadally, S.; Enbody, R. J. An Empirical Study of HTTP-based Financial Botnets. IEEE Transactions on Dependable and Secure Computing, volume 13 issue 2. https://ieeexplore.ieee.org/document/6991594.

Head of Testing: Peter Karsai

Security Test Engineers: Gyula Hachbold, Adrian Luca, Csaba Mészáros, Tony Oliveira, Ionuţ Răileanu

Sales Executive: Allison Sketchley

Editorial Assistant: Helen Martin

Tel: +44 20 3920 6348 Email: editorial@virusbulletin.com

Web: https://www.virusbulletin.com/