Name: **Virtual Machine centos_7**

Researcher: **Anatolii Do**

Date: **1.11.2023**

1. Brief description (object of research, infrastructure, tasks):

   The object of the study is the web portal of St. Petersburg State University ITMO.

   Objective: to conduct a forensic study of a malware incident.

   Operating System: Linux 4.9.0-9-amd64: Debian SMP 4.9.168-1IP Host IP Address: 193.41.142.9
   IP Address(es) of attackers: 89.232.113.1, 85.140.1.102

   Server version: nginx/1.10.3 (support ended on 12.04.2017)

   Attack date: 22.05.2019

2. Information Found During Investigation

   2.1. Process Information
   Active processes:
   PID 603 bash
   PID 612 ps

   2.2. Traffic dump

   Detected Http request file robots.txt;

   Following this, probably with the help of a fuzzer, there were attempts at Path traversal with .. /.. /.. /etc/passwd, including encoded variants;

   Google search queries have also been spotted: "OWASP ZAP" is a web vulnerability scanner used especially often by amateur pentesters. Presumably, exactly this program was used for scanning. It was launched without any options and with default settings.

   There have also been attempts to perform SQL injection and arbitrary code execution (XSS) in the login field, apparently by the same utility. Brute force attempts are also noticeable.

2.3. Memory dump...

2.4. Information about malicious files:
drop.sh – a shell script that loads another two: bot and diamorphine.ko
bot –
diamorphine.ko is an LKM rootkit for Linux 2.6.x / 3.x / 4.x kernels.

2.5. Network Connection Information
...

2.6. Dependencies
...

2.7. Log files

In the SSH connection logs, a connection to the docker_admin was recorded on May 23, 2019 from the IP address 83.232.113.1 at 1:56:06.

Analysis of the contents of var/log/auth.log.1 showed that the user's password docker_admin was brute-forced from 1 a.m.

At 1:50:10 a.m., the password was received from the IP address 83.232.113.1, port 50511 via SSH2.

Next, attackers tried to run crontab -e but was prompted with deny access: docker_admin is not in sudoers. Six minutes later, another successful attempt was made and a cron instruction was added, which after three minutes loaded the drop.sh into legitimate /bin/avahi.

In crontab itself, an unerased instruction was found: @reboot /bin/avahi/

The contents of /bin/avahi/ (aka drop.sh) are attached to the report (see Screenshot 1)

From the contents, it is clear that the bot and diomorphine.ko, downloaded from http://37.46.128.71, were placed in /tmp/N3M3SIS_folder.

diamorphine.ko is a rootkit that gives root rights to the user – in fact, by launching it, all the actions of the attacker are finished: he has gained full control over the machine. Screenshots of the VirusTotal results of the above files are attached to the report. (see Screenshot 2, Screenshot 3, Screenshot 4)

Then, the attacker ran ps aux to display a list of processes, and then launched nohup.out

At 2:21 a.m., the bash history file was deleted, and 7 minutes later, the 1855 2.8 process was killed.

Screenshots.

enclosed with archive

2.9. Other data (database dumps, configuration files, etc.)
...

3. Summary

The attacker scanned the system using the OWASP ZAP utility on open TCP port 80, which led to a view of the /etc/passwd/ file and a list of available users on the server. This was followed by a brute force of the password of one of them: docker_admin. The user did docker_admin have access to write files to /bin/avahi, where the shell script drop.sh was loaded, loading two more additional ones: bot and diamorphine.ko, the first one is a malware (bot) sleeping in the system, and the second one is needed to hide their work.

Best practices for safety and security measures:

1) Upgrade your version of nginx (two years out of date);
2) Strictly filter user input;
3) Use absolute paths for files;
4) Restrict users' rights to files and directories that correspond to their expected set of operations;
5) Make sure that user passwords are difficult to brute-force using dictionaries rockyou.txt and others. (see NIST password policy) and train personal staff on information security;
6) It is worth considering transferring authentication to SSH keys instead of passwords, so that any brute force becomes useless;
7) Also, loading and configuring WAF in your systems is a good solution, especially automatic ones;