Hacked

date 24.10.2021

solved post time of CTF

category Forensic score none

Description

A log analysis, with time delay.

Attached files

solve.java

```java
public static void main(String[] args) throws Exception {
    File file = new File("access.log");
    FileInputStream fis = new FileInputStream(file);
    String contents = Files.readString(file.toPath());
    Pattern pattern = Pattern.compile("^192.168.100.1 - - \\[01\\/Oct\\/2021:([0-9]{2}:[0-9]{2}:[0-9]{2})\\] \"GET \\/users=ASC,\\(select \\(case when \\(substring\\(password,([0-9]+),([0-9]+)\\) = ([a-zA-Z0-9_])\\) then sleep\\(3\\) else sleep\\(1\\)\\)\\) HTTP\\/1\\.0\" 200 1200", Pattern.MULTILINE);
    Matcher matcher = pattern.matcher(contents);
    DateFormat formatter = new SimpleDateFormat("HH:mm:ss");

    StringBuilder res = new StringBuilder();
    long last = 0;
    String before = null;
    while(matcher.find()) {
        long cur = formatter.parse(matcher.group(1)).getTime();
        if(last == 0)
            last = cur;
        if(cur - last >= 3000)
            res.append(before);
        last = cur;
        before = matcher.group(4);
    }
    System.out.println(res);
}
```

join.java

```java
public static void main(String[] args) throws Exception {
    int w = 290;
    int h = 30;
    int s = 8;
    int t = 9;
    int wf = w;
    int hf = h * t;
    for(int j = 0; j < s; j++) {
        BufferedImage result = new BufferedImage(wf, hf,
BufferedImage.TYPE_INT_ARGB);
        Graphics2D gr = (Graphics2D) result.getGraphics();
        for(int i = 0; i < t; i++) {
            System.out.println("2/" + (i == 0 ? "" : i) + (j + 1) + ".png");
            BufferedImage im = ImageIO.read(new File("2/" + (i == 0 ? "" : i) + (j
+ 1) + ".png"));
            gr.drawImage(im, 0, i * h, null);
        }
        ImageIO.write(result, "PNG", new File(j + ".png"));
    }
}
```

Summary

Anaylis the log, then parse the timing, then decrypt the content inside of the zip file.

Flag

```
hology4{c0ngr4tzzz_y0u_got_m3}
```

Detailed solution

1. The exploit that was described in the log file was time based. The key was on the character and the date.
2. After parsed the log and got the password `SupEr_s3CreT_p4ssw0rd_f0R_sup3r_seCr3t_LaUnCh_c0De`, unzip the file
3. The contents of zip file were png images, files' name ranges from {n}1-{n}8, but skipping {n}9 and {n}10
4. After digging up, those images were splitted, so I wrote a joiner.
5. After the image was joined, resulted in QR codes, which contains sequence of base64 string.
6. Join the string, and decode.

Another solutions

Not that I can think of.