Nakivo Backup & Replication - Multiple vulnerabilities

CVE-2020-15850, CVE-2020-15851

Type

Lack of access control, Local privilege escalation

Severity

High

Affected products

Nakivo Backup & Replication versions below 10.3.0.n54125

Credits

These issues were found by Łukasz Czarniecki.

CVE Reference

CVE-2020-15850, CVE-2020-15851

Read more

Timeline

14/04/2020	F-Secure makes initial contact with the vendor
23/04/2020	F-Secure delivers the advisory to Nakivo
10/06/2020	Nakivo advises that they propose to address the issues in an upcoming release (10.2)
10/06/2020	F-Secure requests an update on the intended release date for version 10.2
16/06/2020	Nakivo advises that version 10.2 is scheduled for release in October 2020

Intro

NAKIVO Backup & Replication software provides image-based, application-aware, incremental backup and replication features. Commonly used to backup physical and virtual machines to a NAS for example. According to the vendor website and marketing materials, they boast that big brand organisations such as Verifone, Coca-Cola, Honda, Radisson, DHL et al trust NAKIVO and use their software. F-Secure Labs have identified two security issues with the NAKIVO Backup & Replication software; "Local privilege escalation in Nakivo Director on Linux (CVE-2020-15850)" & "Lack of access control in Nakivo Transporter (CVE-2020-15851)". These vulnerabilities can be leveraged to escalate privileges as well as gain unauthenticated remote access to backups.

The NAKIVO Backup & Replication software consists of three main components:

- Director - a central management service providing a web interface (that by default) listens on TCP port 4443 (HTTPS).
- Transporter - a network service responsible for performing all of the backup, data protection and recovery tasks and listens on TCP port 9446 (TLS).
- Backup repository - data folder on local or remote storage (NAS).

All components can be installed on a single machine or can be distributed across multiple machines and geographical locations. Two security vulnerabilities were identified in the Director and Transporter components.

NAKIVO Backup & Replication can be installed on Windows and Linux, or deployed as a pre-configured Virtual Appliance. The core of the product is written in (cross-platform) Java. Free and trial versions of the software can be obtained from the vendor website https://www.nakivo.com/resources/download/trial-download/.

The issues discussed in this advisory relate to the Linux version

21/09/2020	F-Secure release the advisory
04/05/2021	Issues fixed in v10.3

Lack of access control in NAKIVO Transporter (CVE-2020-15851)

For details of the Transporter service see the vendor websites: https://www.nakivo.com/blog/nakivo-backup-replication-components-transporter/ and https://helpcenter.nakivo.com/display/NH/Transporter.

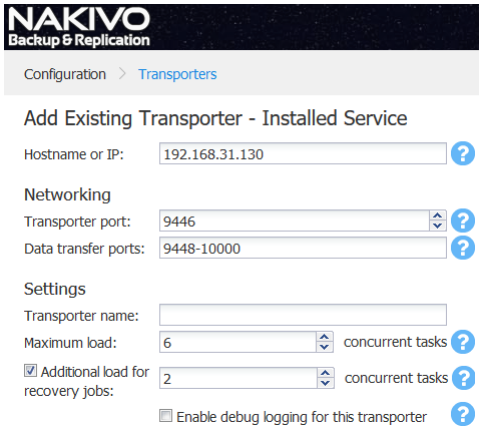
In a default installation NAKIVO Backup & Replication automatically backs up its own configuration, including all jobs, inventory, information about connected transporters, repositories, etc to a local unencrypted backup repository. It can be found in the following location "/opt/nakivo/repository".

The NAKIVO Transporter service does not implement any access or authentication controls. It is therefore possible to access unencrypted repositories without providing any credentials. Using the trial or free versions of the software readily available from the vendor website, it is a trivial task to exploit the issue.

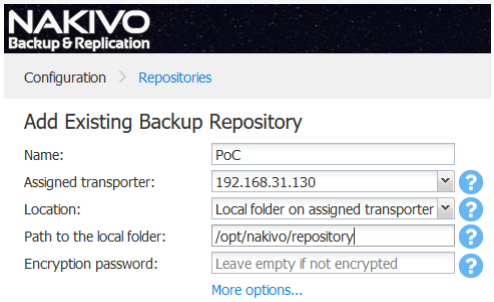
Exploitation steps

1. Install and configure a local Nakivo Backup & Recovery Director instance.

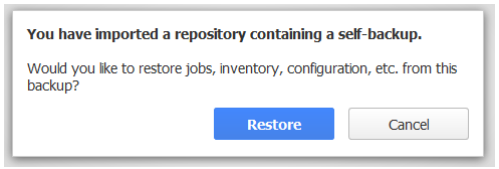
2. Connect to the exposed Transporter service via the "Add Existing Transporter" step (a hostname and port is required).



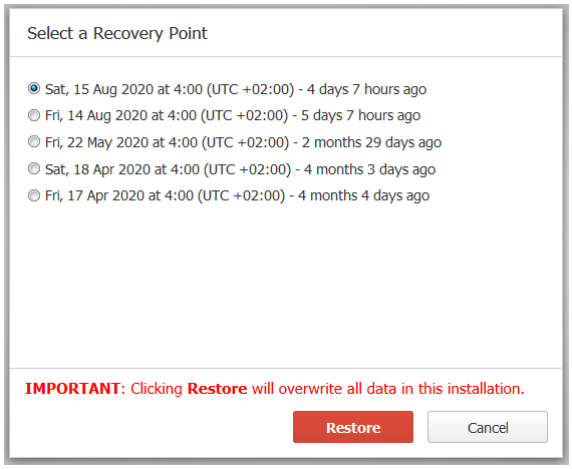
3. Connect to a remote backup repository via the "Add existing backup repository" option by choosing the transporter added in the previous step and providing the repository path (the default path is "/opt/nakivo/repository").



4. If the repository contains a Director configuration backup it will be automatically imported.



5. Access to all unencrypted backup data existing in the configured repository path on the Transporter will also be possible.



6. Imported configuration data may contain passwords to other encrypted repositories configured on the Director.

Detection

Default installations use a self signed NAKIVO TLS/SSL certificate. The Common Name is "NAKIVO Backup & Replication Transporter".

```
% ncat --ssl -v 192.168.86.237 9446
Ncat: Version 7.80 ( https://nmap.org/ncat )
Ncat: Subject: C=US, CN=NAKIVO Backup & Replication Transporter
Ncat: Issuer: C=US, CN=NAKIVO Backup & Replication Transporter
Ncat: SHA-1 fingerprint: 2552 FF72 9529 5F26 C901 42F4 56EE D580
2767 C0AA
Ncat: Certificate verification failed (self signed certificate).
Ncat: SSL connection to 192.168.86.237:9446.
Ncat: SHA-1 fingerprint: 2552 FF72 9529 5F26 C901 42F4 56EE D580
2767 C0AA
^C
```

certificate common names approx. 1k hosts found on the internet exposing the NAKIVO Transporter service and of those, 278 hosts were also exposing the Director service. Shodan queries identified little to no exposed systems:

https://www.shodan.io/search?query=ssl%3A%22NAKIVO%22+port%3A%229446%22

https://www.shodan.io/search?query=ssl%3A%22NAKIVO+Backup+%26+Replication+Transporter%22

Many systems appear to be home users who have presumably accidentally exposed their NAS to the internet. However, if the vendor marketing and boasts are to be believed; the solution is also popular with enterprises also.

A PoC scanner is below and can be used to identify potentially vulnerable transporter services. The scanner sends a 'PING' to the Transporter service:

```
import ssl, socket, sys, getopt, re
import M2Crypto, OpenSSL
context = ssl.SSLContext(ssl.PROTOCOL_TLSv1_2)
context.verify_mode = ssl.CERT_NONE
def main(argv):
 inputfile = ''
 port = 9446
 try:
  opts, args = getopt.getopt(argv,"hi:",["ifile="])
 except getopt.GetoptError:
  print 'scanner.py -i <inputfile>'
  sys.exit(2)
 for opt, arg in opts:
  if opt == '-h':
   print 'scanner.py -i <inputfile>'
   sys.exit()
  elif opt in ("-i", "--ifile"):
   inputfile = arg
  f=open(inputfile, "r")
  hosts = f.read().split('\n')
  for host in hosts:
   if host != "":
    s = context.wrap_socket(socket.socket(socket.AF_INET),
server_hostname=host)
    try:
     cert = ssl.get_server_certificate((host,port))
     x509 = M2Crypto.X509.load_cert_string(cert)
     CN = x509.get_subject().as_text()
     if re.search("CN=NAKIVO",CN):
      s.connect((host, port))
      ping =
'280a2430383061303363332d366465662d346433382d383564362d3163
336234373035653733341000'.decode('hex')
      s.send(ping)
      data=s.recv(1024)
      pattern = "\x10\x03\x72"
      regex = re.compile(pattern)
      for match_obj in regex.finditer(data):
       offset = match_obj.start()
       ver_len = ord(data[offset+3:offset+4])
       print '[x]
host:',host,'port:',port,'version:',data[offset+4:offset+4+ver_len]
    except socket.error:
     print "[!] failed to connect to:",host,"port:",port
if __name__ == "__main__":
 main(sys.argv[1:])
```

(https://github.com/nccgroup/blackboxprotobuf) Python library was used to analyse the Transporter communications.

The Java source file com/nakivo/nbr/controller/remote/protocol/PrimaryProtocolExtender.java contains the command code mappings.

The following message issues a PING command. The command code for Ping is 0.


```
{
 "1": "080a03c3-6def-4d38-85d6-1c3b4705e734",
 "2": 0
}
```


The response is processed and the version string is extracted by the scanner.


```
{
 "15": "071C0986-1FA3-EDCC-66CC-9533D4BBDE80",
 "14": "9.2.1.r40930",
 "19": "56 4d 99 8f 67 66 56 e5-26 6f 99 68 e8 c1 33 35",
 "18": [
  2,
  3,
  <CUT>
  25,
  18
 ],
 "1": "27d9024a-632d-4f1b-9024-1c58a76500dd",
 "2": 3
}
```


Impact

- Read / modify unencrypted backup repositories.
- Create new backup repositories in any path writable to the Nakivo Transporter process.
- Import the Nakivo Director configuration backup saved to unencrypted repositories.
- Nakivo Director configuration backup contains encrypted credentials for hosts which are being backed up (SSH, AWS keys, VSphere, RDP, etc.), hashes of the web interface passwords and passwords to encrypted repositories. The encryption/decryption key is included in the backup.

Note

This vulnerability does not impact the Transporter service installed on physical hosts as a backup agent. In this configuration the Transporter service requests need to be signed with a pre-shared transporter key.

Workaround

Do not expose the transporter service to public networks. Restrict access from controlled and trusted management networks. Encrypt all backup repositories including the default one.

Consider the security specific implementation and configuration advice

Remediation

Apply the patch or fix from the vendor when it is available.

Local privilege escalation in Nakivo Director on Linux (CVE-2020-15850)

For details of the NAKIVO Director see the vendor website https://helpcenter.nakivo.com/display/NH/Director.

The NAKIVO Director service runs with root privileges and does not restrict access to configuration files.

Exploitation steps

1. Copy the configuration database.
2. Extract valid logins.
3. Connect to the controller web interface: https://localhost:4443
4. Initiate a password recovery for the extracted login.
5. Read the recovery key from configuration file readable for all local users and login.
6. Create a new backup job, in "Options" select "Run local pre job script", give path to a script controlled by the local user.
7. Run the job. User script will be executed with root privileges.

```
$ cp /opt/nakivo/director/userdata/db/product01.h2.db /tmp
$ java -classpath /opt/nakivo/director/libs/h2-1.4.196.jar \
org.h2.tools.Shell -url jdbc:h2:/tmp/product01 \
-sql "SELECT login FROM users WHERE login !='guest' and login
!='8A9E9ED63750271B0137502757F80001';"
$ cat /opt/nakivo/director/forgot_password.txt
```

https://www.shodan.io/search?query=ssl%3A%22NAKIVO%22+port%3A%224443%22

Impact

A local Linux user can gain access to Nakivo Director web interface and elevate his privileges to root.

Workaround

Consider the security specific implementation and configuration advice presented in the vendors guidance https://helpcenter.nakivo.com/display/KB/Security+Considerations.

Remove unnecessary permissions from the Nakivo Director directory in order to protect the configuration data.

```
chmod o-rx /opt/nakivo/director
```

Version 10.3 Update: This version has the CVE-2020-15850 fixed and has added the Transporter password option. In order to protect against the CVE-2020-15851 configure the strong transporter password as described in Nakivo documentation: https://helpcenter.nakivo.com/display/NH/Installing+on+Linux#InstallingonLinux-TransporterInstallation

With Great Research Comes Great Responsibility.

Resources

Research

Expertise

Tools

Advisories

Find Labs

Contact us

GitHub

WithSecure™ Company

Contact WithSecure™

Careers at WithSecure™

WithSecure™ Newsletter

Vulnerability Disclosure Policy

advisories