**COSC540 Computer Networks and Information Security**

Name of the University

Name of the Student

## Link: <https://www.infosecurity-magazine.com/news/okta-investigates-possible-lapsus/>

## CIA Classification:

As this attack by Lapsus$, breached the authorized level of access on their platform and breaks integrity of the access levels as well as confidentiality of the end users who are using this tool as their authentication mechanism.

## Severity:

Okta is one of the leading cloud-based user authentication and management tools widely used by different enterprise users. Through the attack, the hackers were able to get super user privilege that enabled them to modify user credentials or even reset the same.

## Reported time and place:

In the month of March,22, the hacker group Lapsus$ posted on their telegram channel few screenshots that shows their access to their website and different other application as superusers or admin users. Even though initially Okta’s chief security officer denied the credibility of the attack claim but later on they agreed that at least 375 organizations who were their customers were affected due to this breach.

## Description of the attack:

By attacking the Okta, the Lapsus hackers got the access with the role of superuser rights, as well as they also had access to Okta’s Jira, Confluence, Zoom, AWS, Splunk, Google Workspace, Salesforce as well as other internal accounts.

One of the posted images by Lapsus$ also dates from 21 January 2022, which indicates that Lapsus attackers were active on Oktas platform for at least two months before.

Hackers from the Lapsus$ gained remote access to a work station that belonged to one of their employees from Sitel, subcontracted company in order to provide customer service and support for Okta customers.

Further intel points to the fact that, initially the contractor’s account that was compromised which in tun enabled ransomware actors in order to infiltrate Okta’s network finally and then target their customers.

## Recommendation for prevention/recovery:

From the example of Okta attacked by Lapsus$ attackers, thus it is always recommended as in Okta’s case it was due to the lack of “use of least privilege access” principle. If the least privilege principle is in place, then a network user will only be allowed in order to perform minimum set of actions with minimal access and privilege that are required for their role in the organization. If the same was in place, the hackers would have limited in what they could access through the compromised support engineer’s account. The contracted user/employee also have the privilege to reset the Password and MFA.

## Link: <https://www.securitymagazine.com/articles/97458-github-notifies-organizations-of-third-party-oauth-token-theft>

## CIA Classification:

As due to this attack the attackers were able to get access to the different code repositories on github that belonged to specific companies/individuals with the compromised “oauth” tokens. Therefore, it breached the code confidentiality of the codes that are developed by the organizations. Therefore, this attack breached the confidentiality factor of the CIA triad.

## Severity:

The Oauth tokens are secret tokens that enables the users in order to get access to their respective repositories. If such a token is compromised that are used by individuals/ organization then a hacker or malicious actor will be able to steal corporate IP/ modify source code in the repositories. This changes in source codes might initiate a supply chain attack from that modified code and could spread malware or collect personal identifiable data from unsuspecting end customers of the organization.

## Reported time and place:

The breach was reported by github on 15th April 2022, 3 days after the attack was discovered github itself when the attackers access its npm infrastructure.

## Description of the attack:

OAuth tokens are presently used as standard method in order to authenticate on different platforms as well as automating cloud services like on code repositories, DevOps pipelines. According to GitHub Security, the tokens were not obtained through a compromise of GitHub as they do not store user tokens in their original and usable formats.

The GitHub Security team, identified unauthorized accesses to its npm production infrastructure on 12th of April when attacker used a compromised AWS API key for the same.

It seems the attacker got the API key only after downloading multiple private npm repositories through the use of stolen OAuth tokens.

## Recommendation:

In order avoid the aftermath of this kind of security breaches, below are the recommendations;

Users should always monitor their private repositories for the secrets/credentials used for authentication stored in them. In order to do so they can use tools like GitHub secret scanning and trufflehog or further enable MFA.

Periodically review OAuth applications which the users have authorized for their personal account or organizational account as well as remove those which are no longer required.

Always review account access activity logs such as used access tokens, different OAuth apps that were used as well as SSH keys for any unauthorized activity or changes that might indicate any that it is coming from attackers.

# Bibliography

Lopez, J. and Rubio, J.E., 2018. Access control for cyber-physical systems interconnected to the cloud. *Computer Networks*, *134*, pp.46-54.

Sheikh, A.F., 2020. Access Control and Privilege Management. In *CompTIA Security+ Certification Study Guide* (pp. 215-236). Apress, Berkeley, CA.

Varadharajan, V., Tupakula, U. and Karmakar, K., 2018. *Lightweight authentication mechanism and OAuth protocol for IoT devices*. Draft, Technical Report TR2: ISIF ASIA Funded Project.

Visočnik, V., 2018. *Comparison of JWT and OAuth 2.0 for authorisation and authentication in rest services* (Doctoral dissertation, [V. Visočnik]).