SECURE CODE

WARRIOR

IMPROPER ASSET MANAGEMENT

We'll go through

some causes and preventions of vulnerabilities in this category

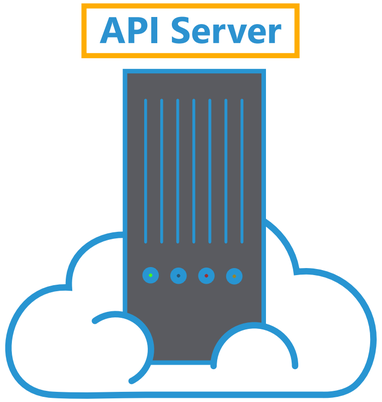
WHAT DO WE MEAN BY IMPROPER ASSET MANAGEMENT?

Improper Asset Management is a vulnerability in which attackers are able to take advantage of outdated, incomplete or undocumented API behaviour.

HOW CAN IMPROPER ASSET MANAGEMENT VULNERABILITIES OCCUR?

Improper Asset Management vulnerabilities happen when an attacker is able to leverage undocumented or insecure functionality

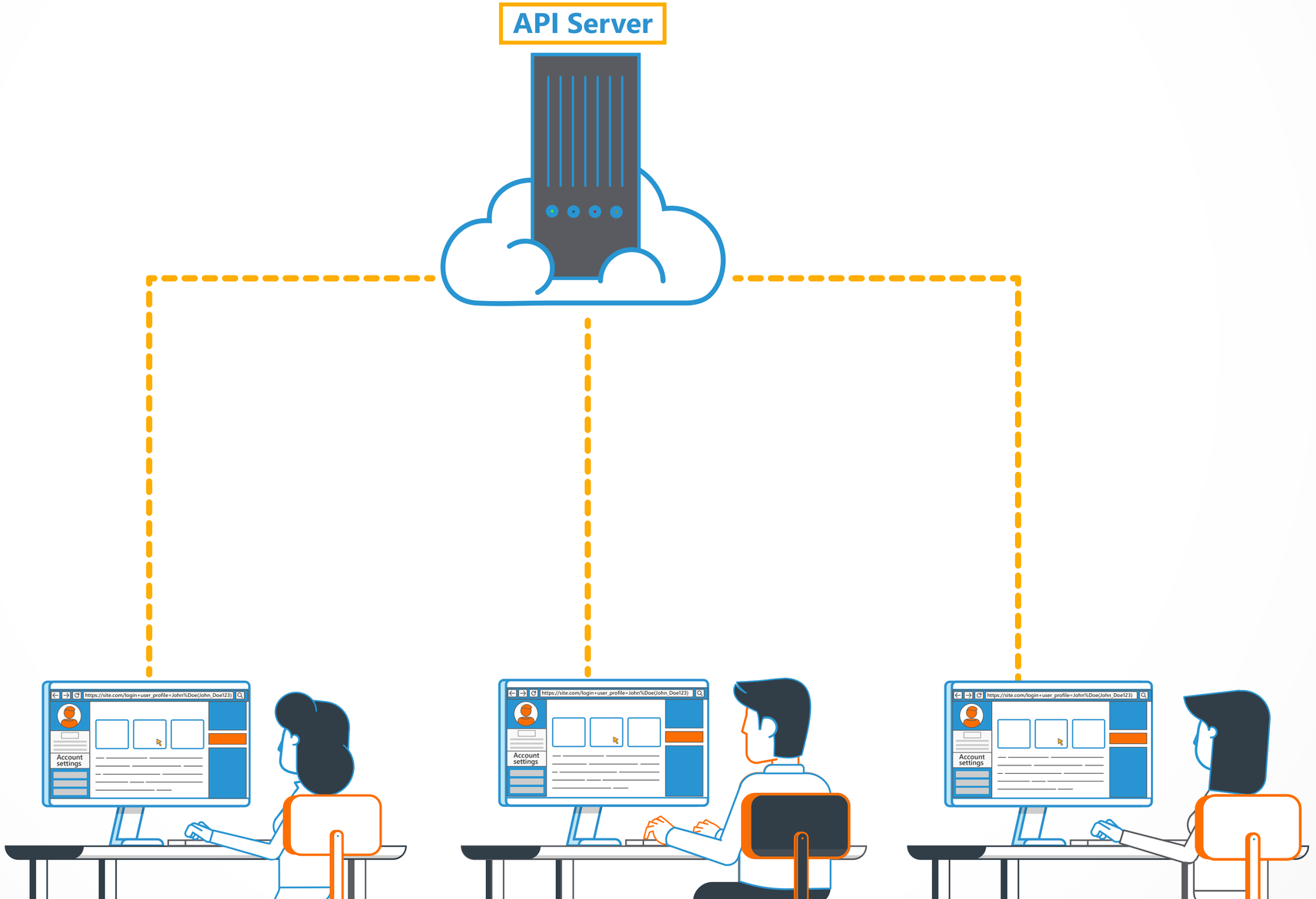
from outdated or development versions of API endpoints.



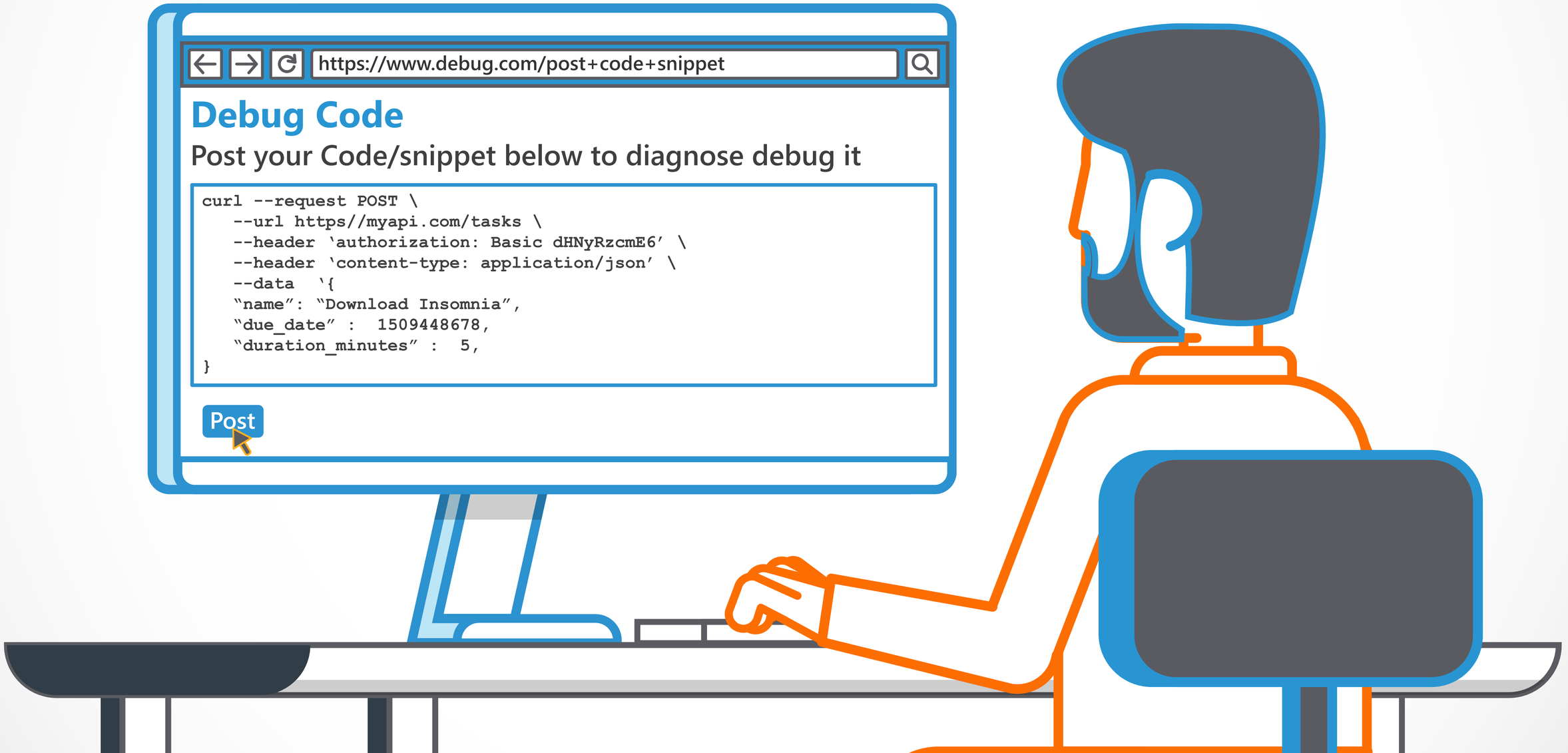
Usually when these versions remain connected to the data stores used by the current API.

LET’S LOOK AT AN EXAMPLE

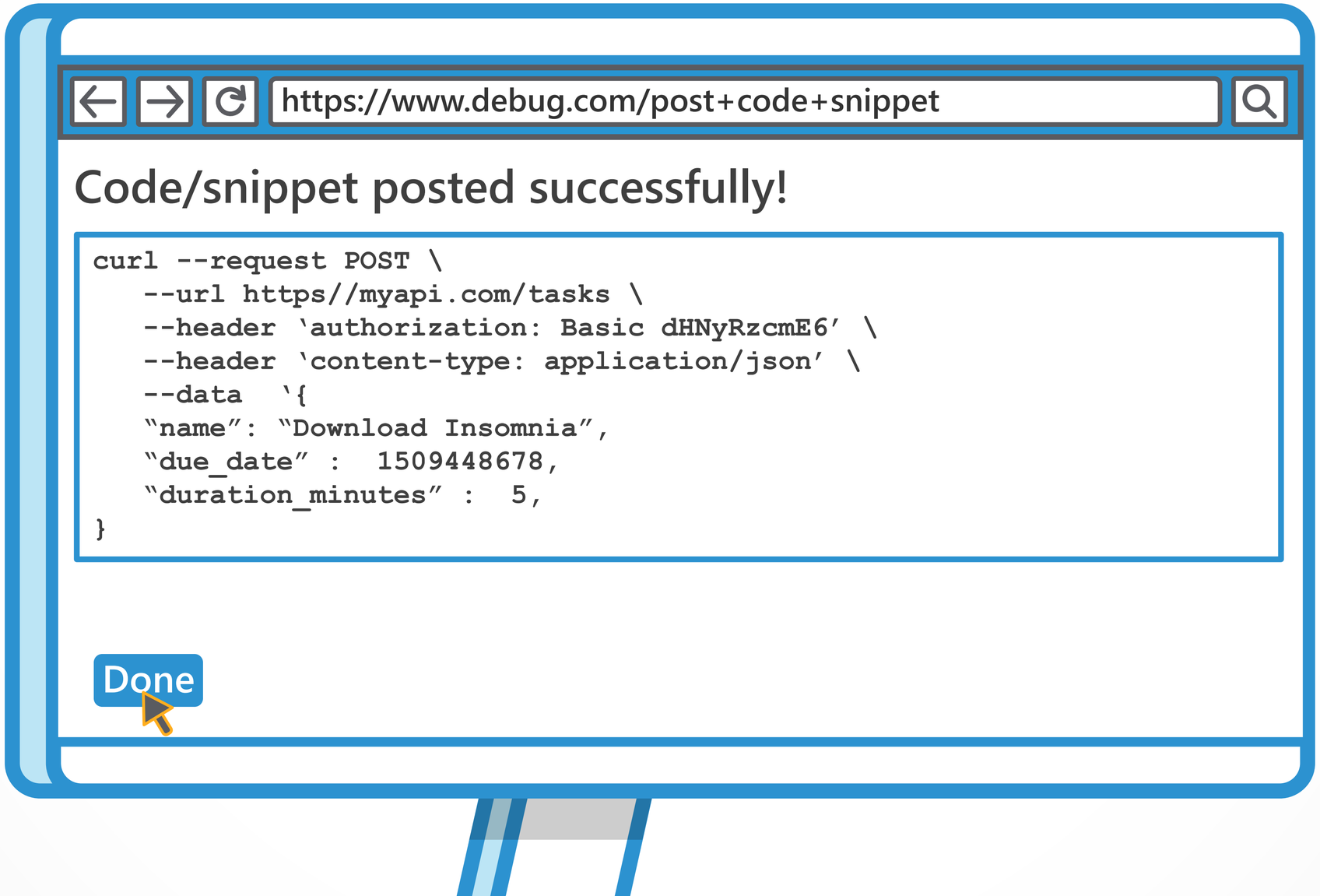
A code compilation service provides an API which is rate-limited per user to ensure that all customers are able to access the functionality according to their access tier.



One of the development team posts a snippet of code to a popular debugging site, hoping to solve a tricky problem they're having.



Unfortunately, their post leaks the address of the staging server used for testing new features against the production database.



Noticing this, an attacker finds that they are able to run tasks freely using the staging server.

To prevent Improper Asset Management function vulnerabilities, developers should:

- Ensure that Production data and datastores are kept separate from Development and Staging environments.
- Implement additional security controls such as firewalls, to prevent unintended access to sensitive or weakened environments.
- Fully document the endpoints and error states that exist within the API.
- Ensure that legacy API versions are validated by any safety checks implemented in the latest version.

Congratulations, you have now completed this module!

SECURE CODE
WARRIOR

www.securecodewarrior.com