SECURE CODE WARRIOR®

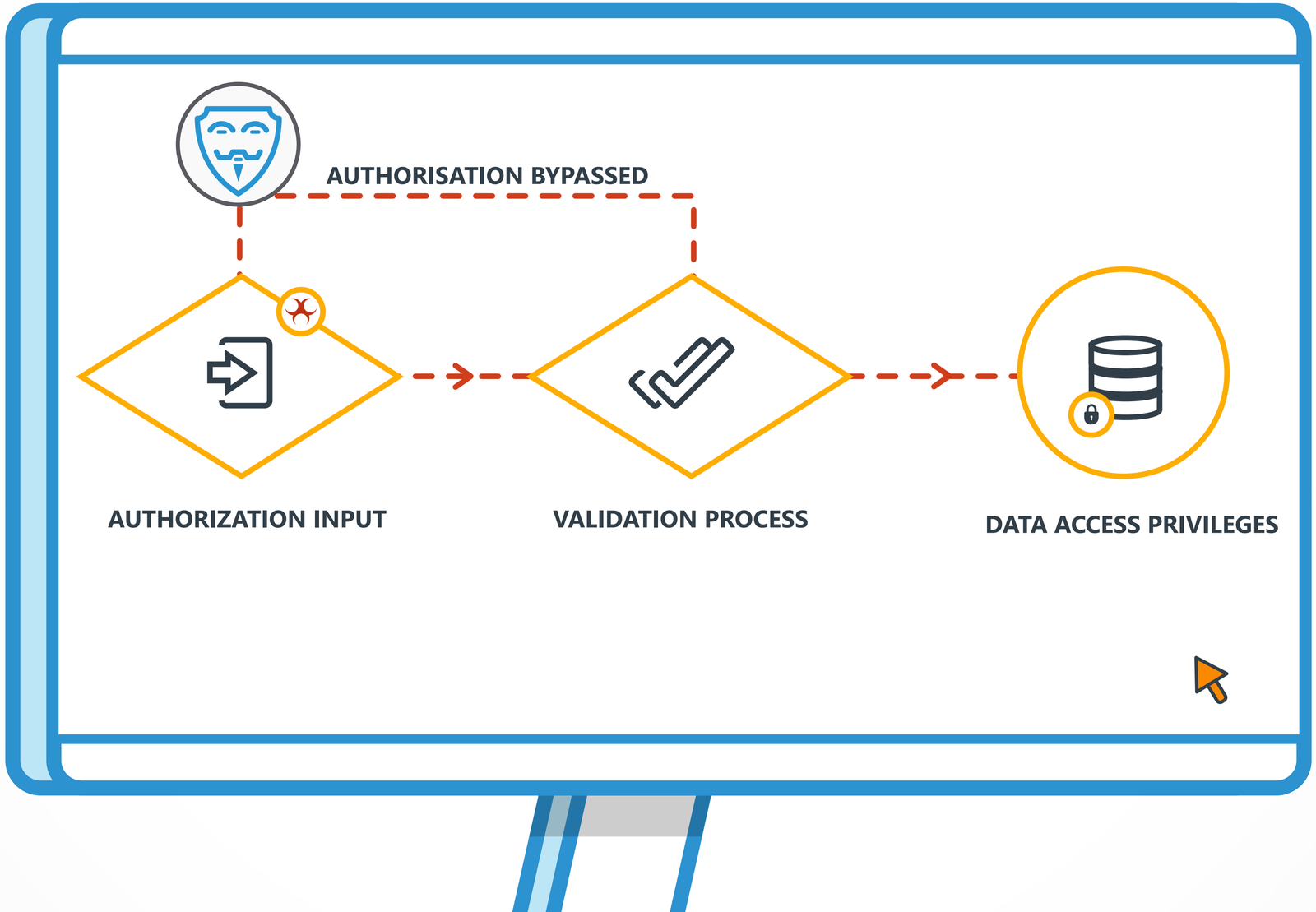
USING INPUT FROM UNTRUSTED SOURCES

We'll go through

some causes and preventions of vulnerabilities in this category

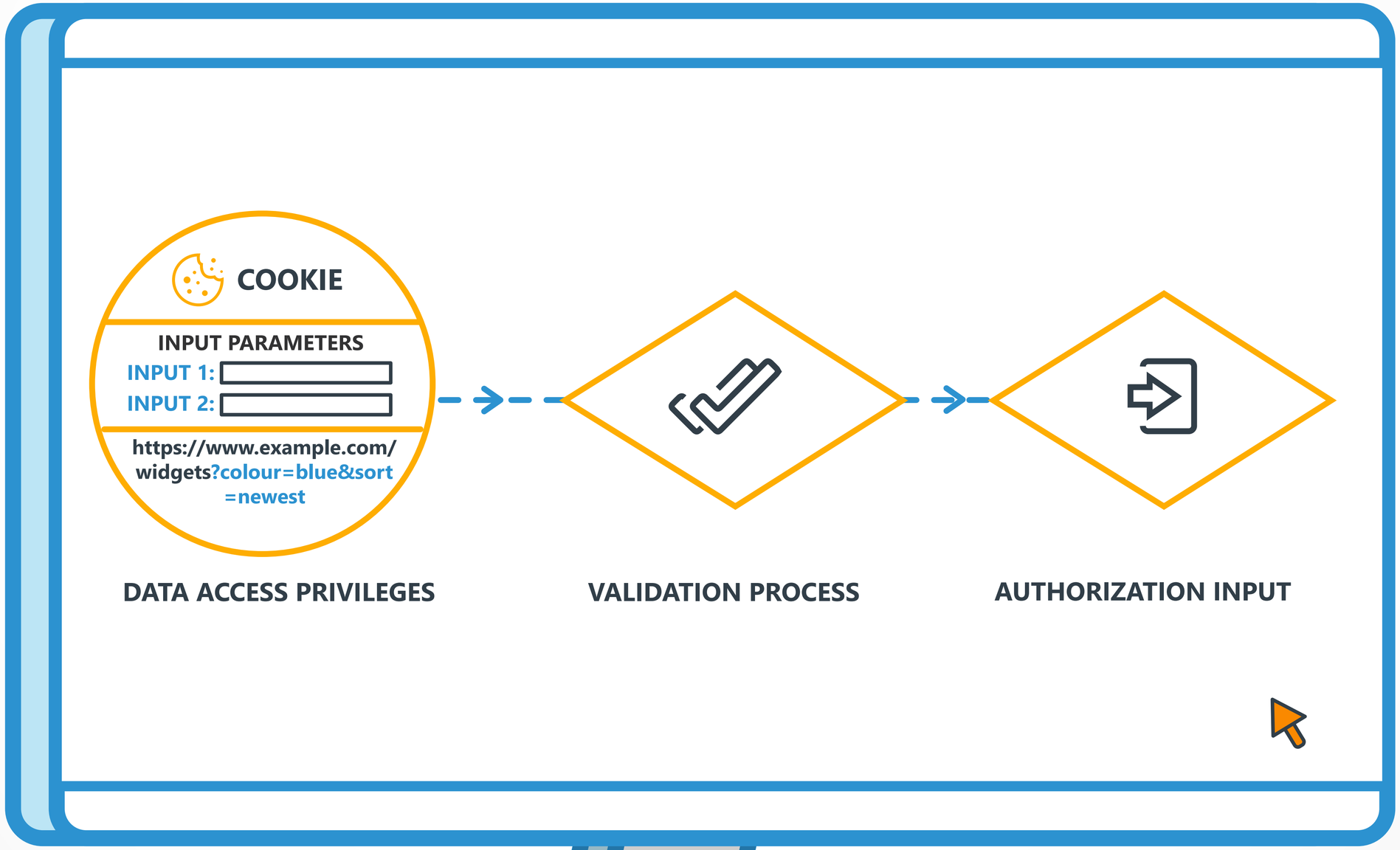
This vulnerability can happen in applications that use input values to complete the authorization processes that determine what data users can access and manipulate.

Sometimes developers don’t realize that attackers can modify input values such as cookies, environment variables, and hidden form fields.



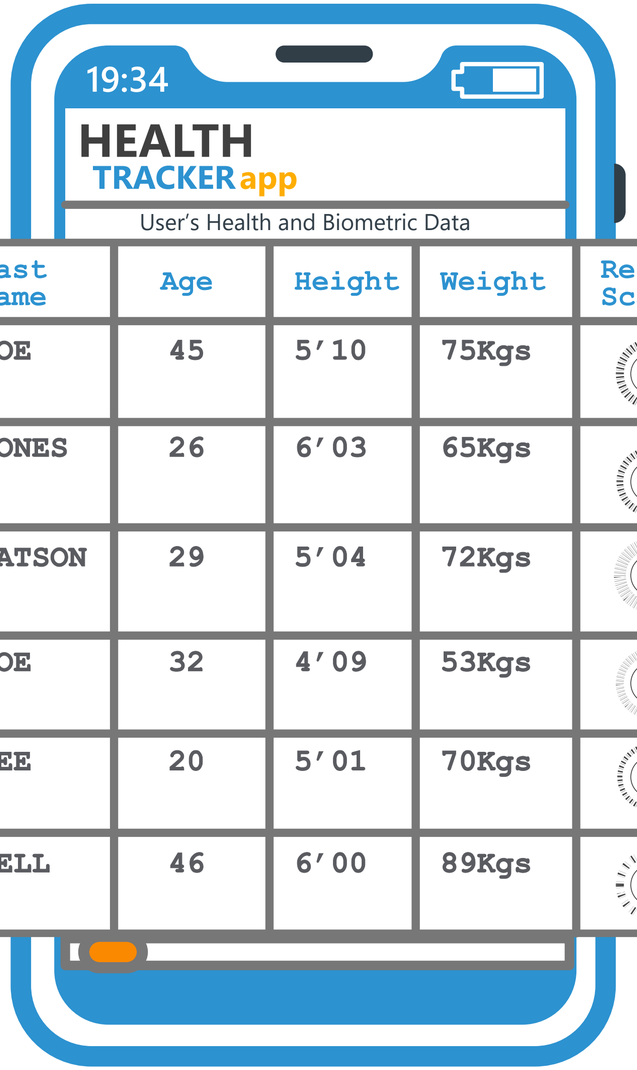
WHERE DOES THIS VULNERABILITY OCCUR?

Just to name a few, examples of input sources that can be tampered with include: roles in cookies, input parameters, query results, sections of a URL and more.



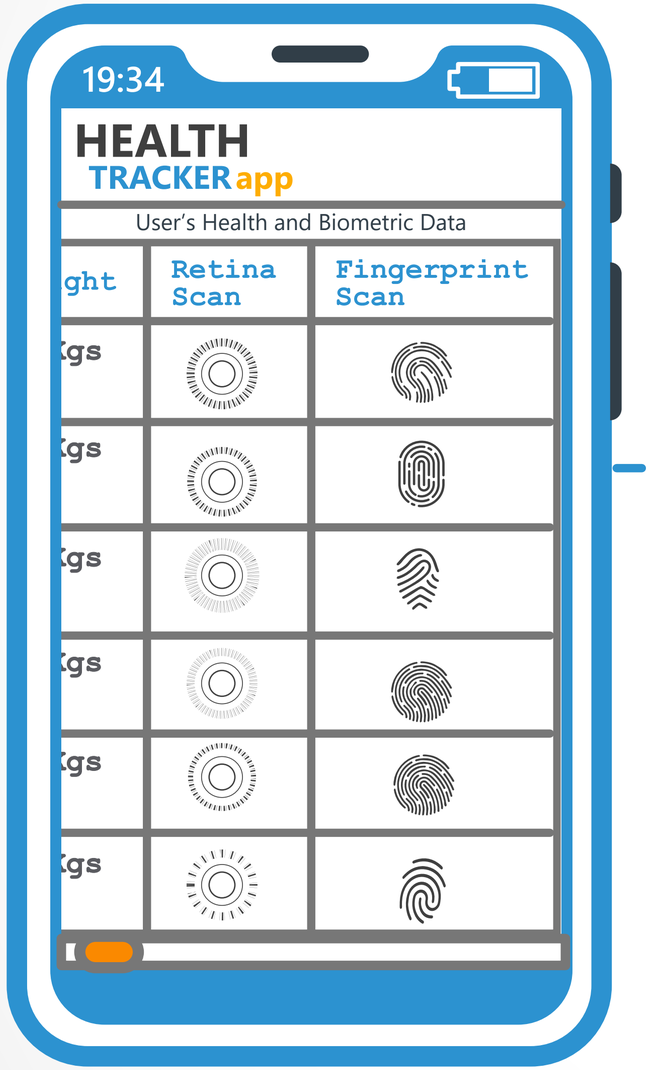
LET’S LOOK AT AN EXAMPLE

This health tracker app stores biometric data about its users.



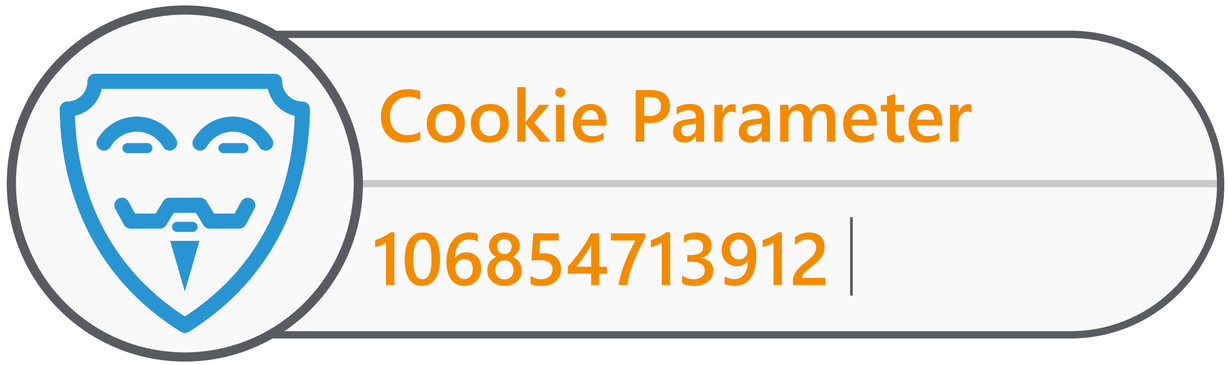
S.No	First Name	Last Name	Age	Height	Weight	Retina Scan	Fingerprint Scan
01	JOHN	DOE	45	5'10	75Kgs		
02	BETTY	JONES	26	6'03	65Kgs		
03	STEVE	WATSON	29	5'04	72Kgs		
04	JANE	DOE	32	4'09	53Kgs		
05	FRANK	LEE	20	5'01	70Kgs		
06	JAMES	BELL	46	6'00	89Kgs		

In order to allow users to access confidential biometric records, the app authorizes each user to specific roles. Each role has varying access privileges on the data.



The application knows whether the ADMIN role is assigned by checking whether a cookie value has been set.

Here, an attacker modifies the cookie parameter with a non-zero value.



As a result, the attacker bypasses the authorization check and the application allows them access to confidential data.

To avoid Using Input From Untrusted Sources, developers should

- Review any potential areas where an untrusted input could potentially enter the application
- Use a trusted framework in the application's architecture that prevents this weakness from occurring
- If possible, try to avoid relying on any type of user submitted input in the authentication process

Congratulations, you have now completed this module!

SECURE CODE

WARRIOR®

www.securecodewarrior.com